Meints, K. & Woodford, A. (2008). Lincoln Infant Lab Package 1.0: A new programme package for IPL, Preferential Listening, Habituation and Eyetracking. [WWW document: Computer software & manual].
URL: http://www.lincoln.ac.uk/psychology/babylab.htm

Lincoln Infant Lab Package 1.0:

A new programme package for IPL, Preferential Listening, Habituation and Eyetracking.

Kerstin Meints & Alan Woodford

Software package Version 1.0

Concept and design: Kerstin Meints & Alan Woodford, University of Lincoln
All programmes written by Alan Woodford, University of Lincoln

General information & Acknowledgements

The Lincoln Infant Lab Package 1.0 consists of five software packages that are used to create, present, score and analyse Intermodal Preferential Looking studies, habituation studies and Preferential Listening studies.

The package was developed for use with infant and child studies which are carried out in Kerstin Meints' Lincoln Infant Lab at the University of Lincoln, UK. The Lincoln Infant Lab Package programmes are currently in use in several universities (e.g. Lincoln, Oxford, Cardiff).

The development of the programmes was supervised by Dr. Kerstin Meints and written by Alan Woodford. We would like to thank the Oxford Babylab team for their "wishlist" and useful feedback. We would also like to extend our thanks to Les Cohen and his team whose "Habit" has inspired us to integrate habituation also into PC-based programmes.

For more information about the Lincoln Infant Lab Package 1.0 programmes, please click the link below to visit our Software page, accessible via the Lincoln Babylab pages: http://www.lincoln.ac.uk/psychology/babylab.htm

Rules for usage

To obtain any of the programmes from the Lincoln Infant Lab Package you can email us under: babylab@lincoln.ac.uk.

Usage is free for all research purposes and individual use. All users must read and follow the ground rules for usage, standards for informed consent, and code of ethics provided at the website above. Please read the relevant sections of the handbook when using the programme for the first time.

All publications using Lincoln Infant Lab Package 1.0 software package should cite the following reference:

Meints, K. & Woodford, A. (2008). Lincoln Infant Lab Package 1.0: A new programme package for IPL, Preferential Listening, Habituation and Eyetracking. [WWW document: Computer software & manual]. URL: http://www.lincoln.ac.uk/psychology/babylab.htm

DISCLAIMER

Limits of warranty, disclaimer of warranty and conditions of usage

The authors have used their best efforts in developing, preparing and testing the programmes and manuals. We make no warranty of any kind, express or implied, with regard to the programmes and manual. The authors and University of Lincoln shall not be liable in any event for incidental or consequential damages in connection with, or arising out of, the furnishing, performance, or use of these programmes or the accompanying manual.

Note:

Although we have spent considerable time developing these programmes and the manual, we can, of course, not guarantee that the programmes or the text are free of errors. If you detect an error, or an omission which you think is important, then we would appreciate if you could **contact us:** babylab@lincoln.ac.uk.

Content

1. Introduction

The Lincoln Infant Lab Package 1.0 was created mainly because, after many years experimenting using Preferential Looking and Listening, we wished to automate many of the time-extensive processes involved in these kinds of infant experimental studies, and we also wished to create a clear and user-friendly means of setting up, testing, scoring and then analysing the data.

Thus, the Lincoln Infant Lab Package 1.0 software consists of five programmes and a manual. The programmes enable the user to create, present, code and analyse Intermodal Preferential Looking studies, Habituation studies, Preferential Listening and Eye-tracking studies.

In addition, with our package, the different methods can for the first time be used by themselves or in combination in one testing session.

This package was developed for use with infant and child studies carried out in Kerstin Meints' Infant Lab at the University of Lincoln, UK. The Lincoln Infant Lab Package 1.0 is currently in use in several other universities (e.g. Oxford, Cardiff).

The development of these programmes was initiated and supervised by Dr. Kerstin Meints and the programmes were written and tested by Alan Woodford. We would like to thank especially also the Oxford Babylab team from 2006/7 for adding their "wishlist" to ours and for useful input and feedback.

2. Short overview of key features

2.1 Experimental methods covered

The current finally enables us to programme, present, code and analyse our studies more efficiently and more conveniently not only for Intermodal Preferential Looking (IPL) and Preferential Listening studies, but also for Habituation and Eye-tracking studies, and the package also enables us now to combine these methods.

2.2 The individual programmes

The experimental set-up programme "Create" allows for easy and flexible use of experimental parameters and stimulus sequences and provides quick and easy setting-up of studies using click and browse. Trial presentation is flexible in choice of stimuli (e.g. video files, still images and soundfiles), in duration, on- and offset, repetition, etc.

The presentation programme "Present" presents the visual and audio stimuli to the participant. All presentation parameters are entered via "Create", leaving only participant details to be entered in "Present". The programme "PresentEye" is adapted for eye-tracking.

The scoring programme ("Score") operates on a video of the participant made during stimuli presentation and allows users to use computer keys to score direction and duration of looking in frame-by-frame mode or in advanced, personalised speeds. The programme can move forward and backward through the video and allows for highly accurate scoring to single video frame resolution.

Finally, data analysis can be carried out with "DataAnalyser", a programme which has been created to interface with the scoring programme output, and which sorts the data into longest look, total looking, first look, latency, timecourse, etc. (please see manual for details and full scope).

2.3 Standard requirements for implementation

PC, projections screen or second monitor for presenting stimuli, at least one video camera for monitoring participants.

For eye-tracking, programmes are available for use with Tobii 1750 or Tobii X120 models.

3. Creating Experiments

There are two ways to create an experiment:
a) using CREATE
b) using a text editor.

The stimulus presentation programme (PRESENT.EXE) uses Babylab Experiment files (.BEX files) to describe experiments. These are ASCII script files consisting of lists of command tags and parameters. With a knowledge of the commands (described in section 3.2), the files can be written using a text editor. However, without this knowledge, the easiest way of creating these files is to use the programme CREATE.EXE (described in section 3.1).

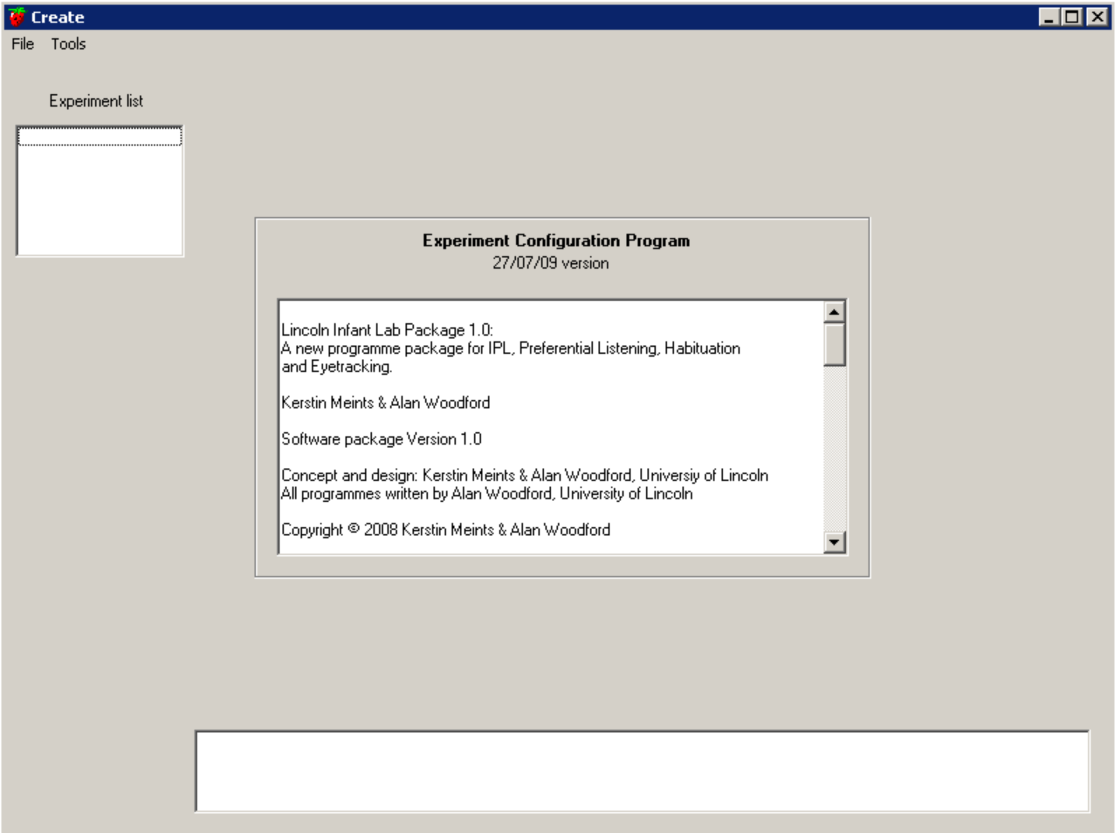
To get started, we recommend the use of CREATE.EXE first.

3.1 Tool 1: *CREATE* - Experiment Configuration Programme (CREATE.EXE)

Experiments are structured in the following way:

Each experiment consists of one or more blocks.
Each block contains a number of trials.
Each trial contains a number of stimuli.
This structure is reflected in the CREATE programme which consists of a number of frames and subframes where settings are displayed and can be changed.

3.1.1 Main programme window



Experiment Configuration Program
27/07/09 version

Lincoln Infant Lab Package 1.0:
A new programme package for IPL, Preferential Listening, Habituation and Eyetracking.

Kerstin Meints & Alan Woodford

Software package Version 1.0

Concept and design: Kerstin Meints & Alan Woodford, Universiy of Lincoln
All programmes written by Alan Woodford, University of Lincoln

Copyright © 2008 Kerstin Meints & Alan Woodford

3.1.1.1 File menu

Within the main programme window, new or existing experiment files can be loaded from the File menu. As they are loaded, they will appear in the Experiment List, on the left of the programme window.
New - creates a new 'empty' experiment. You will be asked for a filename for the new experiment. This will appear in the experiment list and be set as the current experiment.
Open - loads an already existing .BEX experiment file. The filename will appear in the experiment list and be set as the current experiment.
Save - saves the current experiment file.
Save as - saves a copy of the current experiment with a new filename. The name in the experiment list will be changed accordingly.
Close - closes the current experiment. If changes have been made and not yet saved, you will asked if you want to save them.
Exit - Exits the programme. If changes have been made to any loaded experiments and not yet saved, you will be asked if you want to save them.

3.1.1.2 Tools menu

Within the main programme window, the tools menu contains a couple of utilities that allow you to check the experiment.
Check experiment - Checks the current experiment for errors, such as values out of range, conflicting values, etc. Results of this check appear in the frame at the bottom of the programme window.
Stimuli check - Checks that the programme can find all the stimuli specified in the experiment.
Stimuli check is not possible if the experiment is being created on a PC other than that on which it will be run, unless the experiment and stimuli are kept together in one of the recommended directory structures (see 7.4). Results of this check appear in the frame at the bottom of the programme window.

3.1.1.3 Experiment list

Within the main programme window, several experiments can be loaded into the computer's memory at one time, allowing you to copy/paste settings between experiments if required. Experiments that have been loaded into the computer are displayed in the Experiment List. When you select an experiment from the list, it becomes the 'current experiment' and all subsequent operations will apply to that experiment. The name of the current experiment also appears in the title bar.

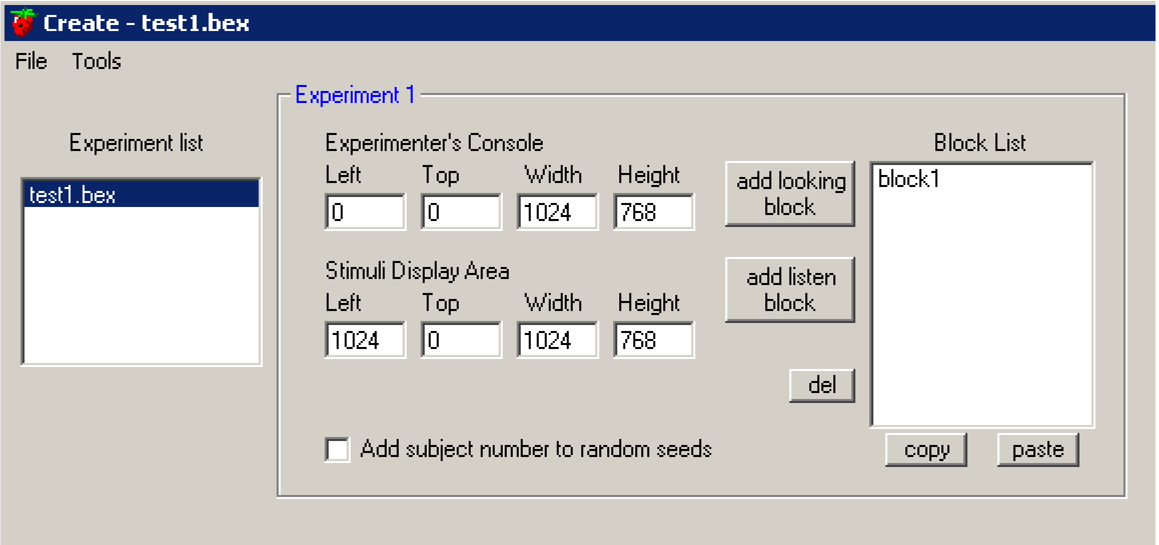
3.1.1.4 Settings Frames

Settings for the current experiment are entered and displayed in a number of frames which appear to the right of the experiment list. The experiment frame shows experiment settings plus a list of block frames. A block frame shows block settings plus a list of trial frames, and a trial frame shows trial settings and may contain a list of visual stimulus frames. The various types of frame are described below.

Navigation between frames is by double-clicking list items or by using 'Back', 'Previous' and 'Next' buttons.

3.1.1.5 Experiment frame

The experiment frame shows settings relevant to the whole experiment, plus a list of blocks. See Figure 2 below.



Experiment list: test1.bex

Experimenter's Console

Left	Top	Width	Height
0	0	1024	768

Stimuli Display Area

Left	Top	Width	Height
1024	0	1024	768

Block List: block1

☐ Add subject number to random seeds

Figure 2: Main programme window, showing an experiment frame

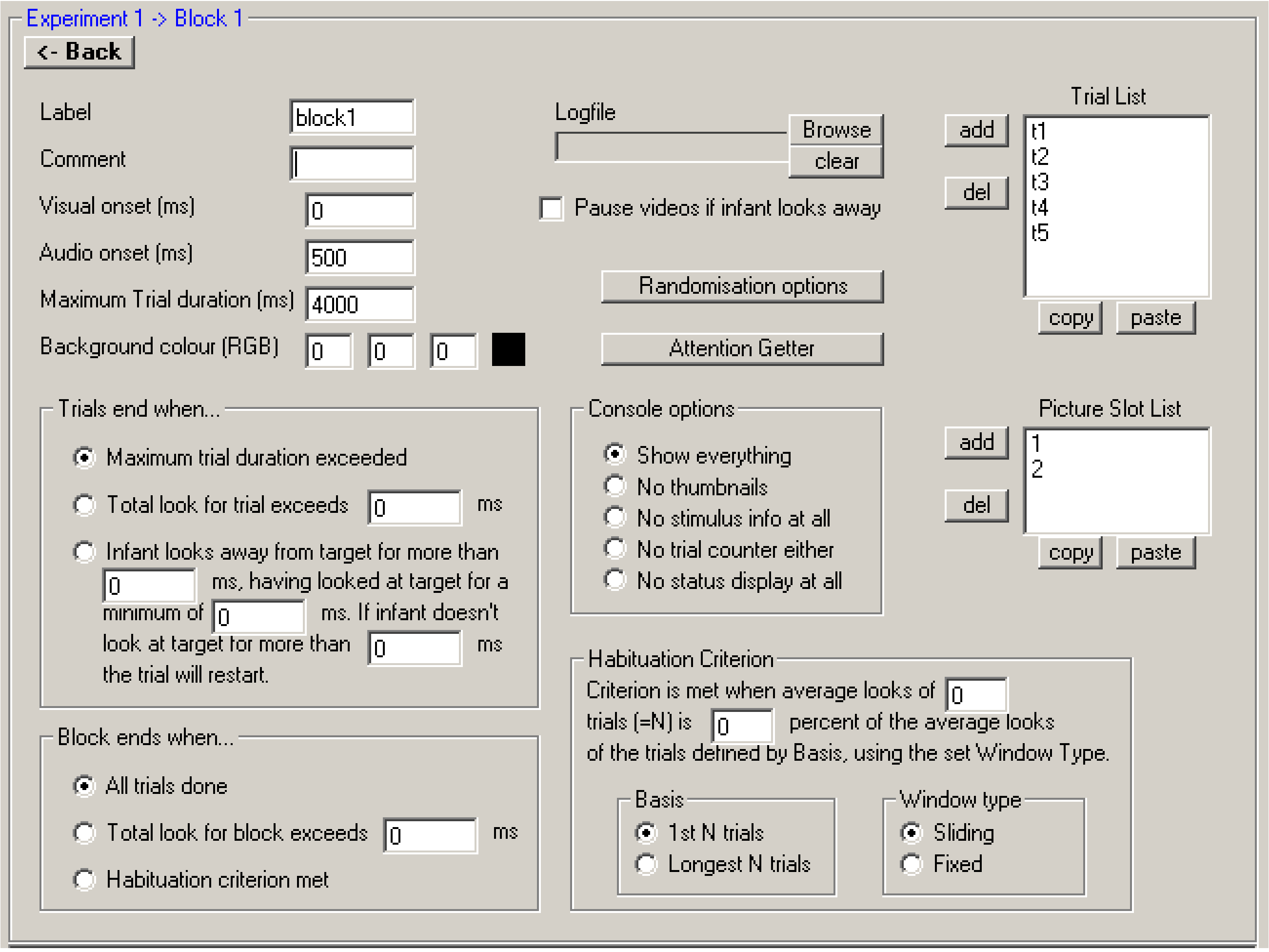
Experimenter's console is the area in which the experiment's status box is displayed while the experiment is running.
Stimuli display area is the area of screen visible to the infant. These settings are used for automatic placement of picture-slots, and also for the experiment programme's understanding of whether a visual stimulus is left, centre or right (it divides the width of this area into thirds and looks to see in which third the centre of the stimulus lies), so some care should be taken when defining this area.

Note that overlap of the experimenter's console and stimuli display area is permitted.

New empty looking or listening blocks can be added to the experiment using the ***add***-*b*uttons. The new block will be inserted above a selected item in the block list. You can delete selected blocks in the list using the ***del*** button. The ***copy*** button will copy selected blocks from the block list. The ***paste*** button will insert copied blocks into the block list above a selected item. You can copy blocks from one loaded experiment and paste them into another. When you copy a block it copies everything in the block including all its trials and picture-slots.
To view/edit the information for a block, double-click a block item in the block list. This will display the block frame.

3.1.1.6 Block frame

The Block frame shows settings relevant to a block, plus lists of that block's picture-slots and trials.



Experiment 1 -> Block 1

Label: block1

Comment:

Visual onset (ms): 0

Audio onset (ms): 500

Maximum Trial duration (ms): 4000

Background colour (RGB): 0 0 0

Logfile

☐ Pause videos if infant looks away

Trial List

t1
t2
t3
t4
t5

Trials end when...

- ● Maximum trial duration exceeded
- ○ Total look for trial exceeds 0 ms
- ○ Infant looks away from target for more than 0 ms, having looked at target for a minimum of 0 ms. If infant doesn't look at target for more than 0 ms the trial will restart.

Console options

- ● Show everything
- ○ No thumbnails
- ○ No stimulus info at all
- ○ No trial counter either
- ○ No status display at all

Picture Slot List

1
2

Block ends when...

- ● All trials done
- ○ Total look for block exceeds 0 ms
- ○ Habituation criterion met

Habituation Criterion

Criterion is met when average looks of 0 trials (=N) is 0 percent of the average looks of the trials defined by Basis, using the set Window Type.

Basis

- ● 1st N trials
- ○ Longest N trials

Window type

- ● Sliding
- ○ Fixed

Figure 3: an example of a block frame

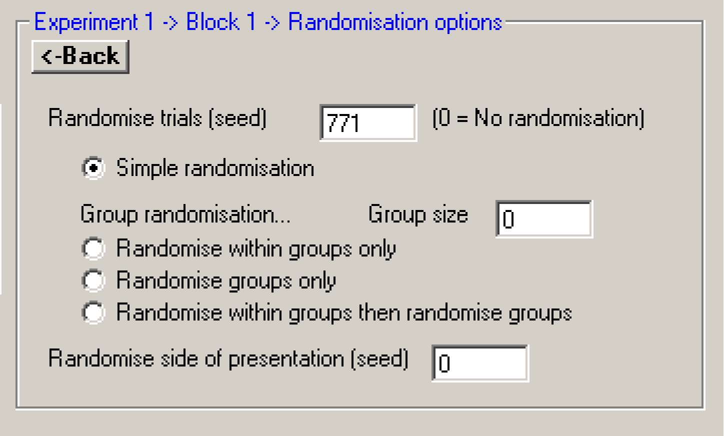
New empty trials can be added to the block using the ***add*** button next to the Trial List. The new trial will be inserted above a selected item in the trial list. You can delete selected trials in the list using the ***del*** button. The ***copy*** button will copy selected trials from the list. The ***paste*** button will insert copied trials into the list above a selected item. You can copy trials between different blocks and experiments. To view/edit the information for a trial, double-click a trial item in the trial list. This will display the trial frame.

Similarly, in looking blocks, new empty picture-slots can be added to the block using the ***add*** button next to the Picture-slot List. The new picture-slot will be inserted above a selected item in the picture-slot list. You can delete selected picture-slots in the list using the ***del*** button. The ***copy*** button will copy selected picture-slots from the list. The ***paste*** button will insert copied picture-slots into the list above a selected item. You can copy picture-slots between different blocks and experiments. Note that when you use ***add***, ***del*** and ***paste*** commands the picture-slots will be renumbered... this may affect the placement of visual stimuli. To view/edit the information for a picture-slot, double-click a picture-slot item in the picture-slot list. This will display the picture-slot frame.

Use the ***back*** button to return to the experiment frame.

3.1.1.7 Randomisation options frame

Shows block settings for the randomisation of trials, and randomisation of the side of presentation of visual stimuli.



Randomise trials (seed) 771 (0 = No randomisation)

● Simple randomisation

Group randomisation... Group size 0

○ Randomise within groups only

○ Randomise groups only

○ Randomise within groups then randomise groups

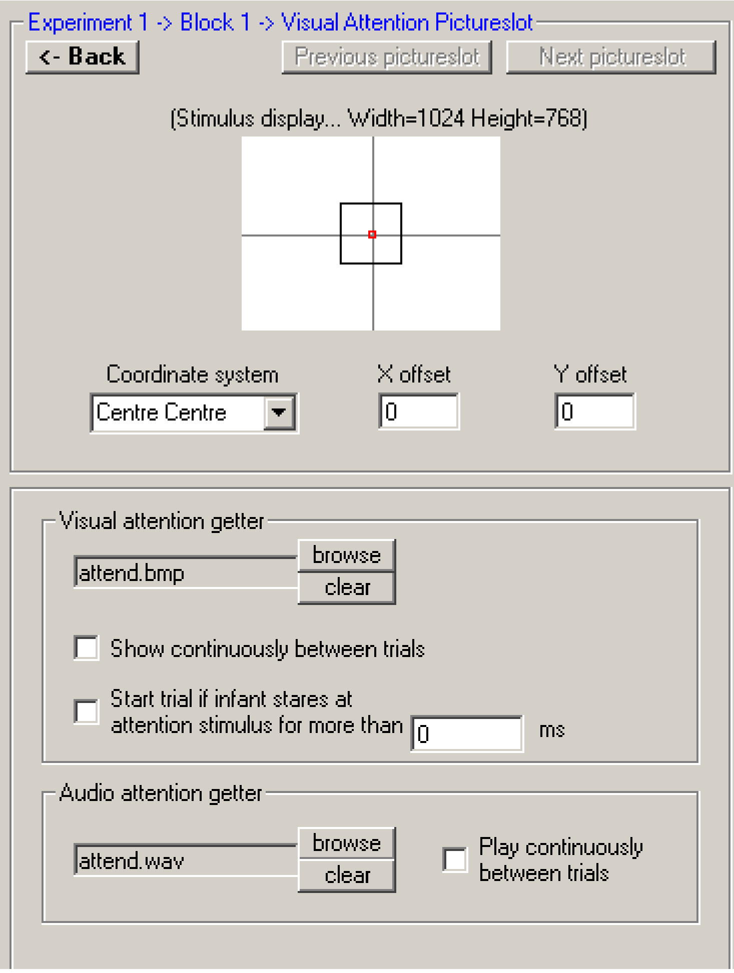
Randomise side of presentation (seed) 0

Figure 4 an example of a Randomisation options frame

Use the ***back*** button to return to the block frame.

3.1.1.8 Attention Getter frame

Shows visual and audio attention-getting stimuli settings for a block.



(Stimulus display... Width=1024 Height=768)

Coordinate system: Centre Centre

X offset: 0

Y offset: 0

Visual attention getter

attend.bmp

□ Show continuously between trials

□ Start trial if infant stares at attention stimulus for more than 0 ms

Audio attention getter

attend.wav

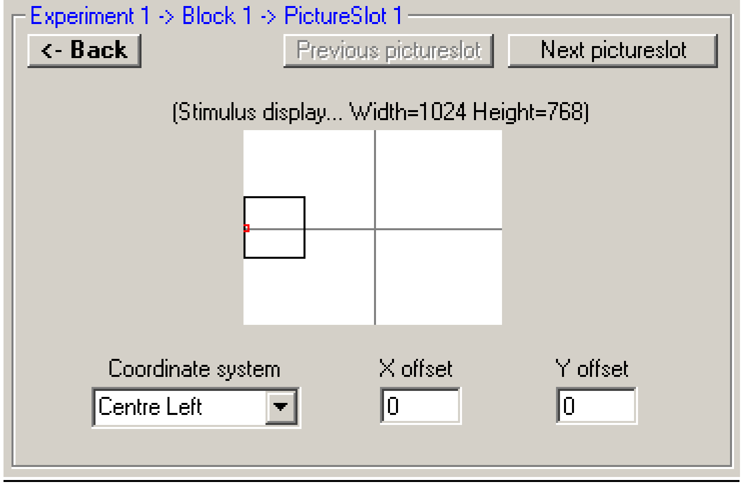
□ Play continuously between trials

Figure 5. Attention-getter frame

Use the ***back*** button to return to the block frame.

3.1.1.9 Picture-slot frame

Shows picture-slot information.



Experiment 1 -> Block 1 -> PictureSlot 1

(Stimulus display... Width=1024 Height=768)

Coordinate system: Centre Left

X offset: 0

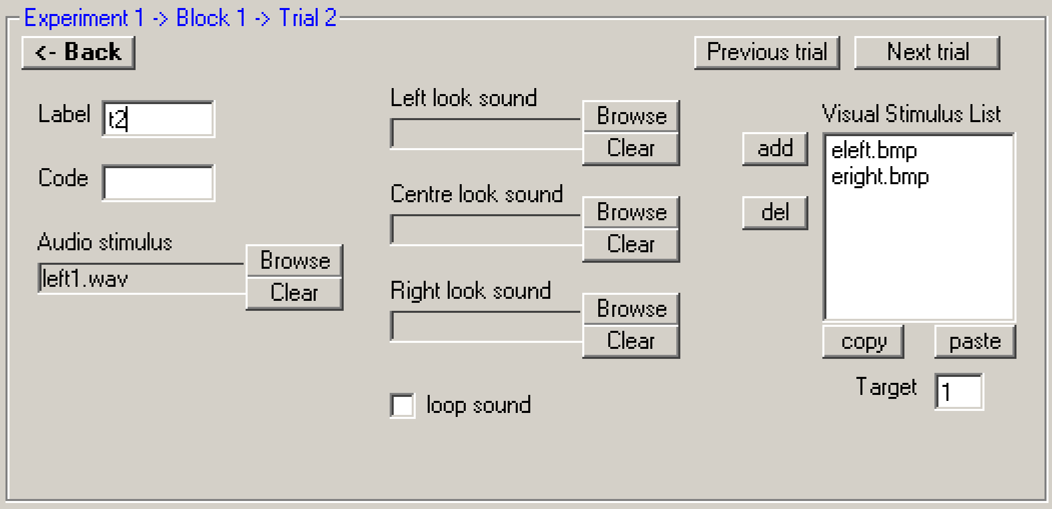
Y offset: 0

Figure 6. Picture slot frame

The ***coordinate system*** listbox allows coarse placement of the picture-slot within the stimuli display area. Alternativly it can be used to specify the how the visual stimulus will be displayed relative to the picture-slot coordinate, and the ***xoffset*** and ***yoffset*** boxes can set the actual coordinate (see also 7.3 “screen areas and picture slots”). A graphical display of the stimuli display area at the top of the frame shows the position of the picture-slot within it. The small red square represents the picture-slot's coordinate, the larger black square represents how a visual stimulus will be displayed relative to that coordinate. Use the ***back*** button to return to the block frame. The ***previous picture slot*** and ***next picture slot*** buttons show other picture-slots from the same block's picture-slot list without the need to return to the block frame first.

3.1.1.10 Trial frame

Shows settings relevant to a trial, plus a list of that trial's visual stimuli.



Experiment 1 -> Block 1 -> Trial 2

Label: t2

Code:

Audio stimulus: left1.wav

Left look sound:

Centre look sound:

Right look sound:

□ loop sound

Visual Stimulus List:
eleft.bmp
eright.bmp

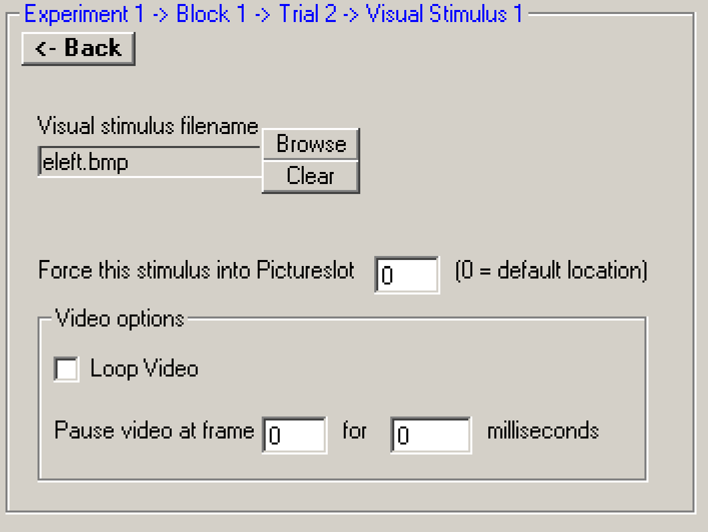
Target: 1

Figure 7. Trial frame

Left look, ***Centre look*** or ***Right look*** sounds, if specified, will play when the infant looks in a particular direction. Note though that if any of those sounds have been specified, the normal ***audio stimulus*** will be

ignored by the experiment programme.
Use the ***Target*** box to specify which item in the visual stimuli list is to be designated as the target. 1 is the first item, 2 is the second, and so on.
New empty visual stimuli can be added to the trial using the ***add*** button next to the visual stimuli List. The new visual stimulus will be inserted above a selected item in the list. You can delete selected visual stimuli in the list using the ***del*** button. The ***copy*** button will copy selected visual stimuli from the list. The ***paste*** button will insert copied visual stimuli into the list above a selected item. You can copy visual stimuli between different trials, blocks and experiments. To view/edit the information for a visual stimulus, double-click a visual stimulus item in the visual stimuli list. This will display the visual stimulus frame.
Use the ***back*** button to return to the block frame. The ***previous trial*** and ***next trial*** buttons show other trials from the same block's trial list without the need to return to the block frame first.

3.1.1.11 Visual stimulus frame



Experiment 1 -> Block 1 -> Trial 2 -> Visual Stimulus 1

Visual stimulus filename: eleft.bmp

Force this stimulus into Pictureslot 0 (0 = default location)

Video options

□ Loop Video

Pause video at frame 0 for 0 milliseconds

Figure 8. Visual stimulus frame

Normally visual stimuli would be placed in picture-slots in the same order as they appear in the picture-slot and visual stimuli lists (unless random side of presentation has been specified for the block). The first visual stimulus would be placed in picture-slot 1, the second in picture-slot 2 and so on. ***Force this stimulus into picture slot*** box allows the visual stimulus to be forced into a picture-slot other than this default. For example you may have three picture-slots, but only one visual stimulus for a particular trial. You could then set the value to 2, to force the stimulus into picture-slot 2 rather than its default position in picture-slot 1.

Note this forcing of picture-slots will override random side of presentation, if set, so its possible if you have three picture-slots, say, to set it up to have one stimulus forced into a central picture-slot while another two are randomly placed in picture-slots to either side. Beware of conflicts when using this setting.

Use the ***back*** button to return to the trial frame.

3.2 Using Text editor to create experiment configuration files

Experiments are controlled by commands set up in a Babylab Experiment File (BEX file) which is used as an input file for the 'PRESENT' programme.

You can use the 'CREATE' programme to generate a BEX file, without requiring knowledge of the file's internal format. Alternatively you can create it from scratch in a text editor, in which case you will need the information below. Another possibility is to create a template for the experiment in 'CREATE', then edit the file in a text editor.

Below is an example of a BEX file for a simple fixed duration preferential-looking experiment with 2 blocks and 2 trials in each block (admittedly, not a very realistic experiment, but gives an idea of what a BEX file looks like inside).

```
<EXPERIMENT>
 <CONSOLE>                 0       0       1024   768
 <EXPTSCREEN>              1024    0       1024   768

 <BLOCK>                   "block1"
  <COMMENT>                "comment"
  <VISUAL_ONSET>            0
  <AUDIO_ONSET>             600
  <TRIAL_DURATION>          4000
  <LOG_FILE>               "test1.log"
  <ATTEND_PIC>             "blob.bmp"
  <ATTEND_SLOT> 0           600     300
  <ATTEND_SOUND>           "chimes.wav"
  <TRIAL_END_OPTION>        0
  <BLOCK_END_OPTION>        0
  <PICSLOT>                 0       650     300
  <PICSLOT>                 0       850     300
  <TRIAL>                  "trial1"
   <VSTIM>                 "dog2.bmp"
   <VSTIM>                 "dog1.bmp"
   <TARGET>                 1
   <ASTIM>                 "lion.wav"
   <CODE>                  "train"
  <TRIAL>                  "trial2"
   <VSTIM>                 "dog1.bmp"
   <VSTIM>                 "dog2.bmp"
   <TARGET>                 2
   <ASTIM>                 "lion.wav"
   <CODE>                  "train"

 <BLOCK>                   "block2"
  <COMMENT>                "comment"
  <VISUAL_ONSET>            0
  <AUDIO_ONSET>             600
  <TRIAL_DURATION>          4000
  <LOG_FILE>               "test1.log"
  <RANDOM_SEED>            77103
  <ATTEND_PIC>              "blob.bmp"
  <ATTEND_SLOT> 0            600     300
  <ATTEND_SOUND>            "chimes.wav"
  <TRIAL_END_OPTION>         0
  <BLOCK_END_OPTION>         0
  <PICSLOT>                  0       650     300
  <PICSLOT>                  0       850     300
  <TRIAL>                   "trial1"
   <VSTIM>                  "dog1.bmp"
   <VSTIM>                  "dog2.bmp"
   <TARGET>                  1
   <ASTIM>                  "lion.wav"
   <CODE>                   "train"
  <TRIAL>                   "trial2"
   <VSTIM>                  "dog2.bmp"
   <VSTIM>                  "dog1.bmp"
   <TARGET>                  2
   <ASTIM>                  "lion.wav"
   <CODE>                   "train"
```

The file is in ASCII format, with each field being delimited by tabs or spaces. Each line consists of a command tag followed by parameters. There are some 50 commands, but not all are required for every experiment, and commands with default values can be left out. Because spaces are used as delimiters, any parameter that contains spaces (say, in a filename) should be enclosed in quotes (" "). To be on the safe side though, we recommend not using spaces in filenames at all if possible.
Below is a summary of the command tags.

3.2.1 Experiment Tags

These tags affect the whole experiment

<EXPERIMENT>
Indicates the start of a new experiment description.

<CONSOLE> ***left top width height***
Defines the screen area allocated to the user's console (where an experiment's status is displayed while the experiment is running).
Example: <CONSOLE> 0 0 1024 768

<EXPTSCREEN> ***left top width height***
Defines the screen area allocated to the experiment stimuli.
Example <EXPTSCREEN> 1024 0 1024 768

<SUBJECT_SEED>
If this tag is present, the participant number will be added to random seeds, otherwise not.

3.2.2 Block Tags

Block tags are used to define a block. Some of the tags affect trials and will apply to all trials within the block.

<BLOCK> ***label***

Defines the start of a new looking-block description. The label parameter is used to identify blocks in the 'CREATE' programme but not currently used by the experiment programme itself.
Example: <BLOCK> "Block 1"

<LISTEN_BLOCK> ***label***

Defines the start of a new preferential-listening block description. The label parameter is used to identify blocks in the 'CREATE' programme but not currently used by the experiment programme itself.
Example: <LISTEN_BLOCK> "Block 1"

<COMMENT> ***comment***

A comment field for the block.

<VISUAL_ONSET> ***time***

Sets the time in milliseconds (from start of trial) at which visual stimuli appear.
Example: <VISUAL_ONSET> 0

<AUDIO_ONSET> ***time***

Sets time in milliseconds (from start of trial) at which audio stimulus occurs.
Example: <AUDIO_ONSET> 500

<TRIAL_DURATION> ***duration***

Maximum trial duration. (note: trials could end earlier in infant-controlled blocks).

<LOG_FILE> ***filename***

Specifies the name of the output logfile. The logfile was used in older versions of the programme to keep track of trial stimuli for analysis. It has now been replaced by .TXT output files for use in the 'DATAANALYSER' programme, but can be used as an additional backup of this data if required. If the file already exists, data is appended to the file, rather than overwriting the file.
Example: <LOG_FILE> "test1.log"

<ATTEND_PIC> ***filename***

Specifies the image or video file to be used for visual attention stimulus.
Example: <ATTEND_PIC> "attend1.mpg"

<ATTEND_SLOT> ***mode xoffset yoffset***

Defines the screen coordinates of the visual attention stimulus within the
stimulus display area of the screen.
Example: <ATTEND_SLOT> 4 0 0

<ATTEND_SOUND> ***filename***

Specifies the sound file to be used for audio attention stimulus.
Example: <ATTEND_SOUND> "ding.wav"

<ATTEND_PIC_CONTINUOUS>

If this tag is present the visual attention stimulus will be displayed continuously between trials, rather than only displayed when the visual-attention key is pressed.

<ATTEND_SOUND_CONTINUOUS>
If this tag is present the audio attention stimulus will be played continuously between trials, rather than only played when the audio-attention key is pressed.

<ATTEND_LOOK> ***time***
This can be used to stop the attention phase and start the trial automatically when the infant looks at the centre for more than the time set by the 'time' parameter.
Example <ATTEND_LOOK> 1000

<RANDOM_SEED> ***seed***
Random seed to be used for simple randomisation of trials. The parameter 'seed' can be any positive integer. The default value 0 is used if no randomisation is required.
Example: <RANDOM_SEED> 77102

<RANDOM_GROUP> ***seed groupsize groupoption***
Use this tag instead of <RANDOM_SEED> to randomise trials in groups. The 'seed' parameter is the same as for <RANDOM_SEED> tag. The parameter 'groupsize' sets the size of each group of trials. The 'groupoption' parameter is a number between 0 and 3 where...
0 = Simple randomisation. All trials within the block are randomised together, groups are ignored.
1 = Trials are randomised within their groups, but the groups aren't randomised.
2 = Groups are randomised but trials within groups are not randomised.
3 = Trials are randomised within their groups, then the groups are randomised
Example: <RANDOM_GROUP> 77102 2 2

<TRIAL_END_OPTION> ***option***
Determines how a trial will end. The 'option' parameter is a number between 0 and 2 where...
0 = Trial ends at the time set with <TRIAL_DURATION> tag
1 = Trial ends if infant's accumulated looking time at target exceeds time set with <TRIAL_TOTAL_LOOK> tag.
2 = Trial ends if infant looks away from target for more than time set with <TRIAL_LOOK_AWAY> tag

<BLOCK_END_OPTION> ***option***
Determines how a block will end. The 'option' parameter is a number between 0 and 2 where...
0 = Block ends after all trials are completed.
1 = Block ends after total looking time exceeds time set with <BLOCK_TOTAL_LOOK> tag.
2 = Block ends when habituation criterion met, as set by <CRITERION> tag

<TRIAL_TOTAL_LOOK> ***time***
Sets total-looking-time threshold at which trials will end if <TRIAL_END_OPTION> parameter has been set to 1.
Example: <TRIAL_TOTAL_LOOK> 5000

<TRIAL_MIN_LOOK> ***time***
Sets the minimum time infant can look at target for it to count as a valid trial. Used in conjunction with <TRIAL_END_OPTION> parameter set to 2.
Example: <TRIAL_MIN_LOOK> 2000

<TRIAL_LOOK_AWAY> ***time***
Sets the minimum time infant can look away from the target to stop a trial. Used in conjunction with <TRIAL_END_OPTION> parameter set to 2.
Example: <TRIAL_LOOK_AWAY> 500

<RESTART_TRIAL> ***time***
Sets the maximum time infant cannot look at a trial before it restarts. Used in conjunction with <TRIAL_END_OPTION> parameter set to 2.
Example: <RESTART_TRIAL> 4000

<BLOCK_TOTAL_LOOK> ***time***
Sets total-looking-time threshold at which block will end if <BLOCK_END_OPTION> parameter set to 1.
Example: <BLOCK_TOTAL_LOOK> 20000

<CRITERION> ***basis wintype winsize percent***
This tag sets the parameters used to calculate the habituation criterion used to end blocks when <BLOCK_END_OPTION> parameter has been set to 2.
Parameters:
basis - Basis used for comparing trial averages where... 0 = 1st ***winsize*** trials 1 = longest ***winsize*** trials
wintype - Window type used for averaging where... 0 = sliding 1 = fixed
winsize - Window size, ie number of trials used for averaging.
percent - Percentage used for criterion calculation.
Criterion is met when the average look of ***winsize*** trials is ***percent*** % of the average look of the trials defined by ***basis*** ,and where averages are based on ***wintype*** window.
Example: <CRITERION> 0 0 4 50

<LOOK_AWAY_PAUSE_VIDEO>
If this tag is present and visual stimulus is a video, then the video will pause if the infant looks away from it.

<RANDOM_SIDE_SEED> ***seed***
Sets random seed to be used for randomisation of side of presentation of visual stimuli. The parameter 'seed' is any positive integer. The default value 0 is used if no randomisation required.
Example: <RANDOM_SIDE_SEED> 44

<BACKGROUND_COLOUR> ***red green blue***
Sets the screen background colour. The parameters are values between 0 and 255, describing an RGB colour.
Example: <BACKGROUND_COLOUR> 128 128 128 (=medium grey)

<CONSOLE_OPTION> ***option***
Specifies the type of status information displayed on the user console while an experiment is running. The parameter is a number between 0 and 4 where...
0 = Show thumbnails, stimulus information, trial counter and experiment status.
1 = Show stimulus information, trial counter and experiment status (but no thumbnails).
2 = Show trial counter and experiment status.
3 = Show experiment status only (to prevent user bias in infant controlled blocks).
4 = Don't display any status console at all.
Example: <CONSOLE_OPTION> 3

<PICSLOT> ***mode xoffset yoffset***
Defines a picture-slot's coordinates for visual stimuli within the stimulus display area of the screen.
Example: <PICSLOT> 4 0 0

<LAMP_FLICKER> ***time***
Used by preferential listening blocks, to set the rate at which external lamps interfaced to I/O port will flicker. The lamp will turn on for the duration set by the 'time' parameter, then turn off for the same duration, and so on.
Example: <LAMP_FLICKER> 500

<FLICKER_OFF> ***time***
Used by preferential listening blocks, to turn off external lamp flickering at a set 'time' after the sound stimulus for each trial has started. If the tag is not included, the flickering will continue until the trial ends.
Example: <FLICKER_OFF> 1000

<STIM_TOTAL_LOOK> ***time***
Used by preferential listening blocks. Sets the maximum time each sound stimulus is played within the block. If the time is exceeded for a particular stimulus, any further trials using that stimulus will not be run.
Example <STIM_TOTAL_LOOK> 30000

3.2.3 Trial Tags

Trial tags affect individual trials.

<TRIAL> ***label***
Defines the start of a new trial description. The label parameter is used to identify trials in the “Create” programme but not currently used by the experiment programme itself.
Example: <TRIAL> "Trial 1"

<VSTIM> ***filename***
Specifies the image or video file to be used for visual stimulus.
Example: <VSTIM> "walk1.mpg"

<TARGET> ***value***
The number of the visual stimulus to be designated as the target.
(1 = the first visual stimulus specified using <VSTIM> tag, 2 = the second, and so on).
Example: <TARGET> 1

<CODE> ***code***
Same as the code field from the old .trl files.
Example: <CODE> "train"

<ASTIM> ***filename side***
Specifies the sound file to be used for normal audio stimulus.
If 'side' is specified as "left" or "right" the sound will be played from the left or right speaker, rather than both speakers (default).
Example: <ASTIM> "lion.wav"
Example: <ASTIM> "lion.wav" "left"

One request was for an option where the stimuli sounds change depending on where the infant is looking. The next three tags deal with that. If any of these has been specified, the programme will ignore the <ASTIM> tag.

<LOOK_LEFT_SOUND> ***filename***
Specifies the sound file to be played if the infant looks left.
Example: <LOOK_LEFT_SOUND> "left.wav"

<LOOK_CENTRE_SOUND> ***filename***
Specifies the sound file to be played if the infant looks to the centre.
Example: <LOOK_LEFT_SOUND> "centre.wav"

<LOOK_RIGHT_SOUND> ***filename***
Specifies the sound file to be played if the infant looks right.
Example: <LOOK_LEFT_SOUND> "right.wav"

<LOOP_SOUND>
If this tag is present sound stimuli will keep looping until the end of the trial.

3.2.4 Visual stimulus tags

Visual stimulus tags provide options for individual visual stimuli.

<PSLOT> ***slot***
Visual stimulus files specified with the <VSTIM> tag will normally be placed in the same order as the picture slots specified with the <PICSLOT> tag. **<PSLOT>** can be used to force the file into a specific picture slot. Default value 0 indicates the visual stimulus will be placed in its default position.
Example: <PSLOT> 2

<LOOP_VIDEO>
If this tag is present and the visual stimulus is a video, then it will keep looping until the end of the trial.

<PAUSE_VIDEO> ***frame duration***
Allows you to pause a video for a particular duration when it gets to a set frame.
Example: <PAUSE_VIDEO> 50 1000

3.3 Sample IPL block

Basic IPL blocks can be set up in a .BEX file, as in the example below.

```
<EXPERIMENT>
 <CONSOLE>                  0        0        1024   384
 <EXPTSCREEN>               1024     0        1024   768
 <SUBJECT_SEED>

 <BLOCK>
  <COMMENT>                "training_block"
  <VISUAL_ONSET>            0
  <AUDIO_ONSET>             600
  <TRIAL_DURATION>          4000
  <TRIAL_END_OPTION>        0
  <BLOCK_END_OPTION>        0
  <RANDOM_SEED>            44
  <PICSLOT>                 3        0        0
  <PICSLOT>                 5        0        0
  <ATTEND_PIC>             "blob.bmp"
  <ATTEND_SLOT>             4        0        0
  <ATTEND_SOUND>           "chimes.wav"

  <TRIAL>
   <VSTIM>                 "butfly.GIF"
   <VSTIM>                 "whale.GIF"
   <TARGET>                 1
   <ASTIM>                 "butfly.wav"
   <CODE>                  "trial1"

  <TRIAL>
   <VSTIM>                 "whale.GIF"
   <VSTIM>                 "butfly.GIF"
   <TARGET>                 1
   <ASTIM>                 "whale.wav"
   <CODE>                  "trial2"

  <TRIAL>
   <VSTIM>                 "butfly.GIF"
   <VSTIM>                 "whale.GIF"
   <TARGET>                 2
   <ASTIM>                 "whale.wav"
   <CODE>                  "code"

  <TRIAL>
   <VSTIM>                 "whale.GIF"
   <VSTIM>                 "butfly.GIF"
   <TARGET>                 2
   <ASTIM>                 "butfly.WAV"
   <CODE>                  "code"
```

Further explanations

<VISUAL_ONSET> and <AUDIO_ONSET> tags are used to set when trial stimuli begin, in this case the visual stimulus at the start of the trial and the sound stimulus 600ms later.

The <TRIAL_END_OPTION> parameter, 0, tells the programme to end each trial at the time set by the <TRIAL_DURATION> tag, in this case 4000ms.

The <BLOCK_END_OPTION> parameter is set to end the block after all the trials are completed.

The <RANDOM_SEED> tag causes the trials within the block to be randomised using a random seed of 44.

The presence of the <SUBJECT_SEED> tag in the experiment settings means that the participant number will be added to this random seed so that each participant will have a differently randomised set of trials.

The <PICSLOT> tags set up 2 locations for the visual stimuli, to the far left and far right of the stimulus display area.

The <ATTEND_PIC> and <ATTEND_SLOT> tags set up a visual attention-getter stimulus at the centre of the stimulus display area.

The <ATTEND_SOUND> tag sets up an audio attention-getter stimulus. The attention getter stimuli are invoked during the experiment by pressing the appropriate command key.

Each trial is set up with 2 visual stimuli and a sound, as defined by the <VSTIM> and <ASTIM> tags.

The <TARGET> tag tells the programme which of the visual stimuli is the target.

The <CODE> tag gives a code describing the trial condition for later analysis. This is any code that the experimenter can give for describing the type of trial, if required. DataAnalyser programme displays the code in the output for each trial, but doesn't do anything further with it. It may be useful in subsequent analysis for sorting out certain types of trial.

3.4 Sample Habituation Block

Habituation blocks can be set up in a .BEX file, as in the example below.

```
<EXPERIMENT>
 <CONSOLE>                        0      0      1024   768
 <EXPTSCREEN>                     1024   0      1024   768

 <BLOCK>                         "habit"
  <COMMENT>                      "habit"
  <CONSOLE_OPTION>                3
  <ATTEND_SLOT>                   4      0      0
  <ATTEND_PIC>                    "attend.avi"
  <ATTEND_PIC_CONTINUOUS>
  <VISUAL_ONSET>                  0
  <AUDIO_ONSET>                   0
  <TRIAL_DURATION>                10000
  <TRIAL_END_OPTION>              2
  <TRIAL_LOOK_AWAY>               500
  <TRIAL_MIN_LOOK>                1000
  <RESTART_TRIAL>                 4000
  <BLOCK_END_OPTION>              2
  <CRITERION>                     0      0      3      50
  <PICSLOT>                       4      0      0

 <TRIAL>                         "hab1"
  <VSTIM>                        whale.gif
  <TARGET>                      1
  <CODE>                       "hab"

 <TRIAL>                       "hab2"
  <VSTIM>                      whale.gif
  <TARGET>                      1
  <CODE>                       "hab"

 <TRIAL>                       "hab3"
  <VSTIM>                      whale.gif
  <TARGET>                      1
  <CODE>                       "hab"
```

Further explanations
For habituation blocks the <COMMENT> tag's parameter should start with the word 'habit' - this is later picked up by the DataAnalyser programme so that it knows it should analyse this block for habituation.
The <CONSOLE_OPTION> tag is set to 3 to hide trial information from the experimenter's console. This is to prevent bias while entering participant looks from the keyboard.
The <ATTEND_SLOT>, <ATTEND_PIC> and <ATTEND_PIC_CONTINUOUS> tags set up a video file as a visual attention stimulus, running continuously between trials at the centre of the screen.
The <TRIAL_DURATION> tag sets the maximum trial length to 10 seconds.
<TRIAL_END_OPTION> and <TRIAL_LOOK_AWAY> tags tell the programme to end a trial if the participant looks away from the target stimulus for more than 500 ms.
<TRIAL_MIN_LOOK> tells the programme that the participant must have looked at the target stimulus for a minimum of 1000ms for it to count as a valid trial. A trial will restart if the participant doesn't look the target stimulus for more than 4000ms, as set by the <RESTART_TRIAL> tag.
The <BLOCK_END> option is set to 2 which causes the block to end when the habituation criterion, defined by the <CRITERION> tag, is met - in this case when a sliding window average look of 3 trials is 50 % of the average look of the 1st 3 trials.
The <PICSLOT> tag places trial stimuli at the centre of the screen.
There should be a sufficient number of trials set up to allow the participant to habituate to the stimuli before the trials run out (i.e. more than in the example above).

3.5 Sample Preferential Listening block

Preferential-listening blocks can be set up in a .BEX file, as in the example below...

```
<EXPERIMENT>
 <CONSOLE>                      0      0      1024   384
 <EXPTSCREEN>                   1024   384    1024   384

 <LISTEN_BLOCK>                "block1"
  <COMMENT>                    "listen_train"
  <RANDOM_SEED>                 12345
  <ATTEND_PIC>                 "lamp.vlf"
  <ATTEND_SLOT>                 4      0      0
  <ATTEND_LOOK>                 1000
  <LAMP_FLICKER>                200
  <FLICKER_OFF>                 1000
  <TRIAL_LOOK_AWAY>            2000
  <STIM_TOTAL_LOOK>            6000

  <TRIAL>                    The trial can also be given a label, but its not compulsory.
   <ASTIM>                   "carrot.wav" left
  <TRIAL>
   <ASTIM>                   "carrot.wav" right
  <TRIAL>
   <ASTIM>                   "truck.wav" left
  <TRIAL>
   <ASTIM>                   "truck.wav" right
```

Further explanations

The <LISTEN_BLOCK> tells the programme to run this block as a preferential-listening block.

The <COMMENT> tag's parameter should start with the word 'listen' - this is later picked up by the 'DATAANALYSER' programme so that it knows it should analyse the block for preferential-listening.

The <RANDOM_SEED> tag will cause the trials within the block to be randomised.

Before the start of each trial a flashing attention stimulus is displayed on the screen. This is a .VLF file set up with the <ATTEND_PIC> and <ATTEND_SLOT> tags.

The <ATTEND_LOOK> tag tells the programme to stop the attention stimulus after the baby has stared at it for 1000ms, then start the trial (ie the programme will start the trial after the centre-look key has been held down for 1000ms).

The trial starts with the lamp on the target side flickering. The rate of flicker is set by the <LAMP_FLICKER> tag. In the above example the lamp will turn on for 200ms then off for 200ms, and so on. When the baby looks towards the flickering lamp, the target sound will start (e.g. if the left lamp is flickering, the sound will start from the left speaker when the left-look key is pressed). The trial stops either when baby looks away from the target side for more than the time set with the <TRIAL_LOOK_AWAY> tag (2000ms in this case), or when the target sound finishes, whichever is the sooner.

The <FLICKER_OFF> tag turns off the flickering 1000ms after sound has started. If this tag is not present the lamp will continue flickering until the trial ends.

Trials will continue either until they have all finished, or until the accumulated look to each particular stimulus, has exceeded the value set with the <STIM_TOTAL_LOOK> tag. If baby has listened to one stimulus for more than this amount, then the programme will stop playing trials containing that stimulus, but continue playing the others. That applies to training blocks.

For testing blocks, set the <STIM_TOTAL_LOOK> parameter to some large number, say 999999, so that the block ends when all trials have been completed rather than after an accumulated looking time.

Trials use the <ASTIM> tag to tell the programme which sound stimulus to play, and from which speaker to play the sound, left or right. If this last parameter is omitted the programme will play the sound at the centre (i.e. from both speakers).

The following tags can be used within preferential listening blocks, others will be ignored.
<COMMENT>, <LOG_FILE>, <ATTEND_PIC>, <ATTEND_SLOT>, <ATTEND_SOUND>, <ATTEND_SOUND_CONTINUOUS>, <ATTEND_FLASH_SIZE>, <ATTEND_LOOK>, <RANDOM_SEED>, <RANDOM_GROUP>, <TRIAL_LOOK_AWAY>, <BACKGROUD_COLOUR>, <CONSOLE_OPTION>, <TRIAL>, <ASTIM>, <LOOP_SOUND>, <CODE>, <LAMP_FLICKER>, <FLICKER_OFF>, <STIM_TOTAL_LOOK>.

The programme will automatically add the looking data for the block to the .ODF output file, so that this file along with the .TXT output file can be fed directly into the 'DATAANALYSER' programme.

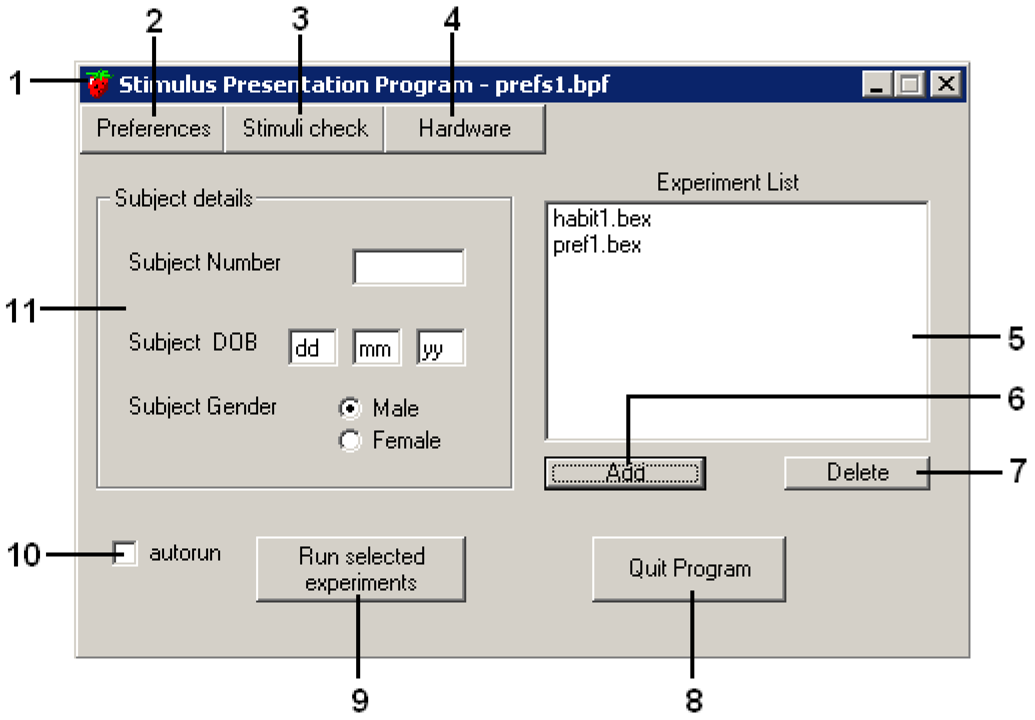
4. Presenting Experiments

Present.exe presents experimental stimuli, as specified by the experimenter in Babylab EXperiment files (.BEX files). You can use Present for IPL, Habituation, and Preferential Listening experiments. An adapted version of Present, namely PresentEye, can be used for IPL experiments with eye-tracking.

4.1 Tool 2: *PRESENT:* Stimulus presentation programme (PRESENT.EXE)

When you start the stimulus presentation programme, you will first be shown a splash screen displaying version, license and acknowledgement information.

There are two buttons on this screen labelled 'Use default preferences' and 'Load preferences file'.
A preferences file holds a user's preferences for such things as which keys to use for experimental control commands. After clicking the appropriate key you will be shown the panel below:



Stimulus Presentation Program - prefs1.bpf

Subject details

Subject Number

Subject DOB dd mm yy

Subject Gender ● Male ○ Female

Experiment List

habit1.bex
pref1.bex

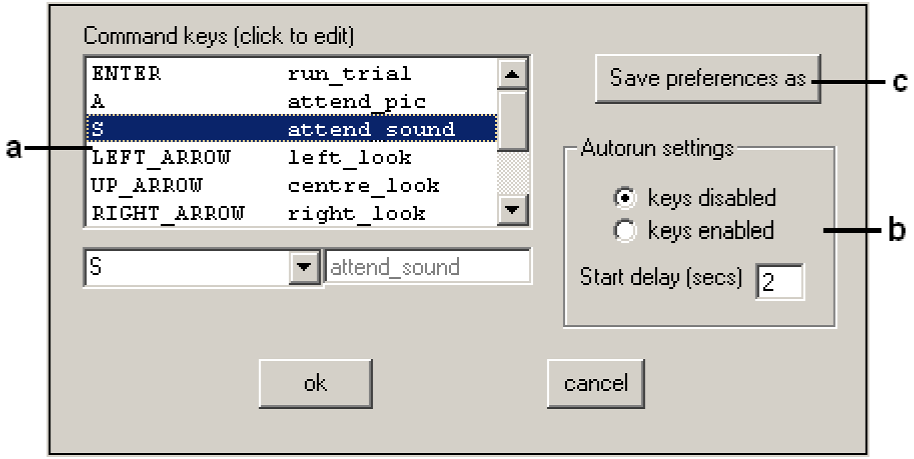
□ autorun

1 The title bar

This shows the name of the preferences file you are using.

2 Preferences button

When you click this button, you will be shown the panel below:



Command keys (click to edit)

ENTER	run_trial
A	attend_pic
S	attend_sound
LEFT_ARROW	left_look
UP_ARROW	centre_look
RIGHT_ARROW	right_look

S attend_sound

Autorun settings

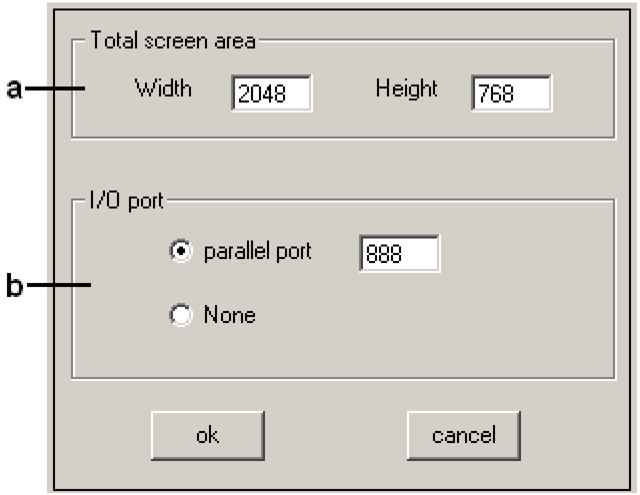
● keys disabled
○ keys enabled

Start delay (secs) 2

a) Allows you to choose which keys you prefer to use for the experiment control commands.
b) Sets whether keys are enabled or not when running an experiment in Autorun/demo mode. Keys should be enabled for testing an infant-controlled experiment. 'Start delay' allows a delay between clicking the run-experiment button and the demo actually starting.
c) Allows you to save preferences to a different file.

3 Stimulus Check button
If you click this button, the programme will test that it can find all the stimuli specified within experiment files selected in the experiment listbox.

4 Hardware button
For hardware settings. If you click this you will be shown the panel below:



Total screen area

Width 2048 Height 768

I/O port

● parallel port 888
○ None

a) Sets the total screen area in use for the experimental setup.
b) Sets the port to be used for controlling external equipment.

5 Experiment Listbox
This holds a list of experiment files.

6 Add button
Use this button to add a new experiment file to the list.

7 Delete button
Use this button to remove selected file(s) from the experiment list.

8 Quit programme button
Exits the programme.

9 Run selected experiments
Runs one or more experiments selected in the experiment listbox.

10 Autorun tickbox
Tick this box to run experiments in Autorun/demo mode. This is useful if you want to check experiments/stimuli combinations before running them with your participants. In this mode no output will be saved. If you have disabled keys in the Preferences panel, fixed duration experiments will run trials non-stop without the need to press the 'start trial' key between trials.

11 Enter participant details here
Participant details will be recorded in the .odf output files (see data analysis programme).

Once you are satisfied that you have entered all the necessary information for your experiment, you go to 9 Run selected experiments, and your experiment starts.

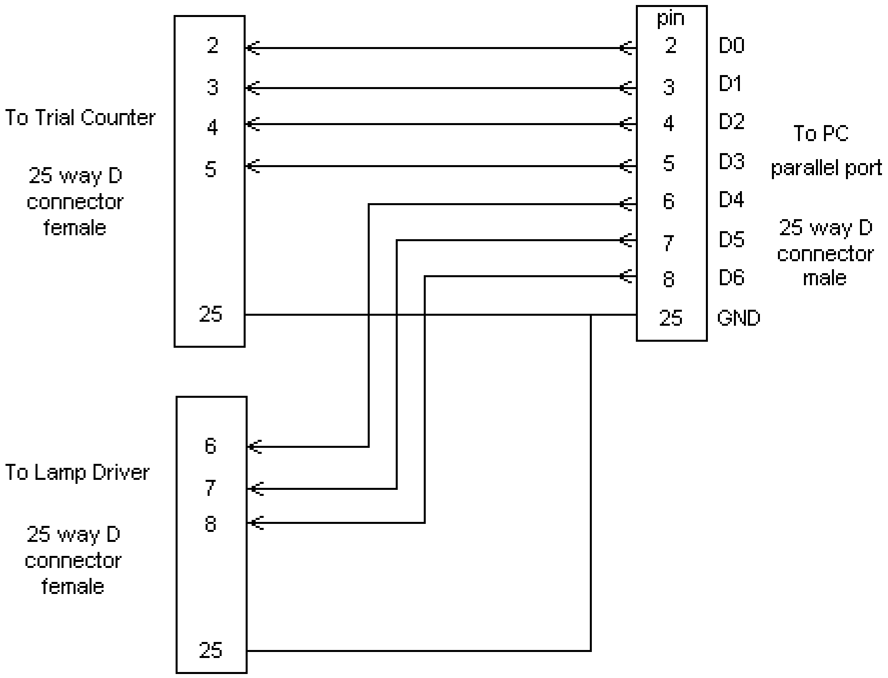
4.2 Control of additional external equipment: trial counter and lamp driver

The PRESENT programme provides signals from the output pins of the PC's parallel-port to control two items of external equipment, a trial-counter and a lamp-driver.
Trial-counters are used because we wish to see, during scoring, which trials the child is viewing (however, without seeing the trial stimuli themselves, so that we can be truly blind to the stimuli while scoring looking behaviour). In the current set-up at Lincoln, the trial numbers displayed by the trial-counter are superimposed on the video by mixing them in from an additional camera. Another possibility would be to place the trial counter in the field of view of the camera filming the child. Alternatively one could use a video-mixer to record both the child and the stimuli to see which trials children are doing. Then, after synchronising the first trial during scoring, mask that part of the screen showing the stimuli.

Lamp-drivers are used for Preferential Listening experiments only. Thus, they are **not necessary** for IPL, habituation or eye-tracking experiments. The lamp driver supplies the power to drive the lamps in the Preferential Listening booth. It is a sort of amplifier, enabling the low power digital signals from the computer to switch higher power to the lamps.

A description of the signals is provided below should anyone wish to build their own equipment.

Splitter-box for Babylab, so that PC's parallel port can drive both the Trial Counter, and the Lamp Driver for Listening experiments.



4.2.1 Trial counter control (only if you want to use a trial counter)

D0-D3 are used to control a trial-counter that is visible in the field of view of the camera videoing the participant during the experiment. This equipment consists of a box showing a 2-digit 7-segment LED display. The box also incorporates 2 LEDs that show the state of the video and audio stimuli. PRESENT programme provides the following signals to control this equipment:

At the start of an experiment all 4 lines (D0-D3) are initially low. D1 is set high briefly (approx 2msec) then reset low. This resets the trial counter.
Before each trial D0 is set high briefly then reset low. This happens after the trial's files have been loaded successfully and before waiting for the experimenter's key press to start the trial. This signal increments the trial counter.
During each trial D2 is set high when the auditory stimulus starts. This turns on the audio stimulus state LED.
During each trial D3 is set high when the visual stimulus is displayed. This turns on the visual stimulus state LED.
At the end of each trial D2 and D3 are reset low again. The stimulus state LEDs go off.
At the end of an experiment D1 is set high briefly then reset low.
Finally all 4 lines are reset low.

4.2.2 Lamp Driver control (only if you want to use Preferential Listening)

D4-D6 lines are used to control the lamp-driver hardware used by Preferential Listening experiments (See Preferential Listening Block description for more information on the use of the lamps). Each line controls one of the three lamps...

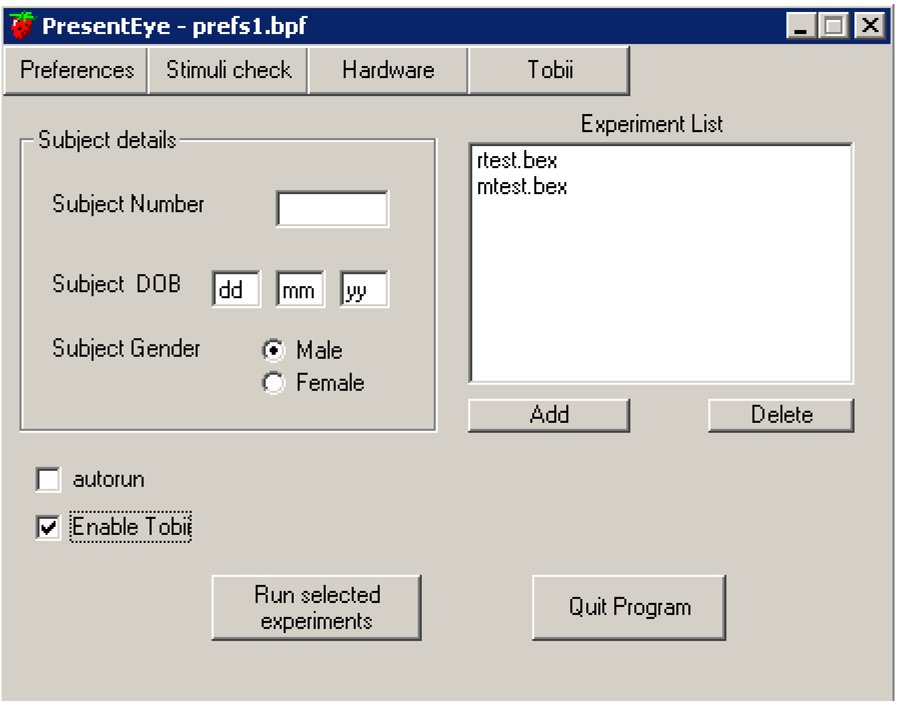
D4 controls the Left lamp
D5 controls the Right lamp
D6 controls the Centre lamp
As the lamp-driver uses inverted logic inputs, the programme initially sets all 3 lines (D4-D6) high so that no lamps are illuminated. Lamps are turned on by resetting the appropriate line low, and off again by setting the line high.

4.3 Eye-tracking

Eye-tracking is to be programmed and run using CREATE and a different PRESENT, namely **PresentEye.** The Tobii button is used to set up the eyetracker (calibration etc) before testing. After the set up is done, the experiments are started in the normal way by clicking the 'Run selected experiments' button.

Testing blocks can be set up in a .BEX file, as in the examples above.

PresentEye.EXE programme is a modified version of the PRESENT.EXE programme that collects Tobii eyetracker samples while running IPL experiments. This programme is still under development and should only be used for IPL experiments with a Tobii eyetracker attached (1750 or X120).
Start up PresentEye to get the main control panel.

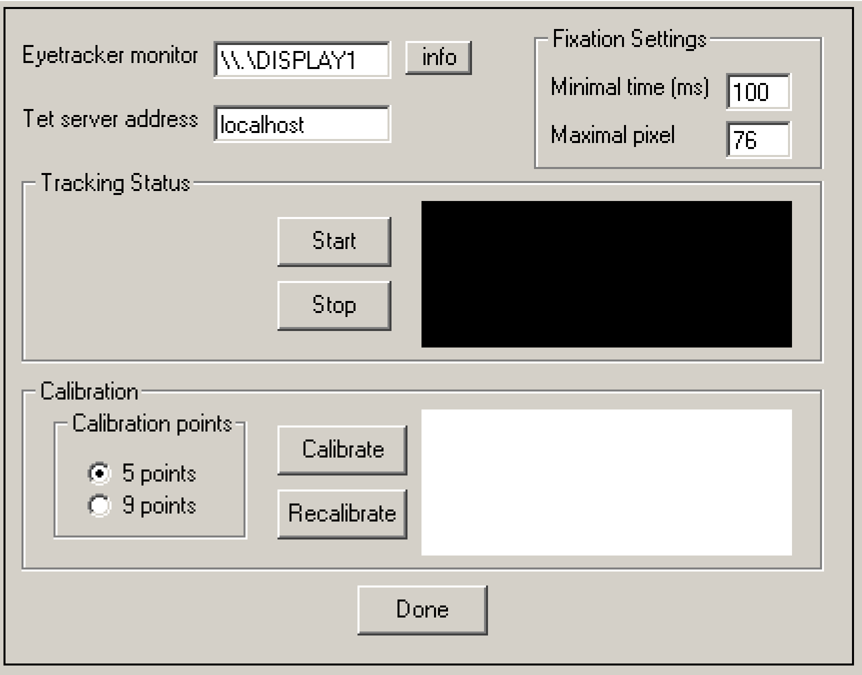


Subject DOB dd mm yy
Subject Gender ● Male ○ Female

Experiment List
rtest.bex
mtest.bex

□ autorun
☑ Enable Tobii

Before running an experiment, set up the programme to work with the Tobii eyetracker by clicking the ***Tobii*** button. This displays the Tobii control panel.



Eyetracker monitor: \\.\DISPLAY1

Tet server address: localhost

Minimal time (ms): 100

Maximal pixel: 76

● 5 points
○ 9 points

The text boxes at the top of the panel tell the programme which monitor is associated with the eyetracker, and the IP address or name of the host where the Tobii TETserver runs.

The Fixation Settings define eye fixations, used to calculate looks for the output .ODF file.

With the participant in position and the eyetracker connected and powered up, use the ***Start*** button to check that the eyetracker can detect the participant’s eyes. If succesful, representations of the eyes appear in the black Tracking-Status box to the right. Use the ***Stop*** button to stop the check.

If the eyes are being succesfully tracked, use the ***Calibrate*** button to perform eyetracker calibration - there is an option for 5-point or 9-point calibration. The calibration screen will be presented to the participant. This consists of blue spots appearing on the screen in sequence. When the calibration process is complete, the results will be displayed in the white calibration box on the control panel. Each calibration point will have red/green lines of various lengths - short lines indicate accurate calibration. If some points have poor or missed calibration, you can use the ***Recalibrate*** button to redo those points. Note: a separate calibration has to be done for each participant prior to testing.

With a successful calibration, click the ***Done*** button to return to the main control panel, from where the experiment can now be started.

During the experiment, the programme will store eye samples as they come in from the eyetracker. These raw samples (time, x, y) are saved in the experiment's .EVT output file. At the end of the experiment the programme also calculates fixation data from the raw samples and uses those to add left, middle, and right looking data to the .ODF output file. This allows the .ODF file to be inserted directly into the DataAnalyser programme without the need to score a video of the participant.

5. Coding looking behaviour

With the next tool, you can code the recorded person’s looking behaviour.

5.1 Tool 3: Multiple-Video Scoring Programme (SCORE.EXE)

This programme enables you to code the looking behaviour into looks to left, right and middle (straight forward) from one or more video files of the participant.

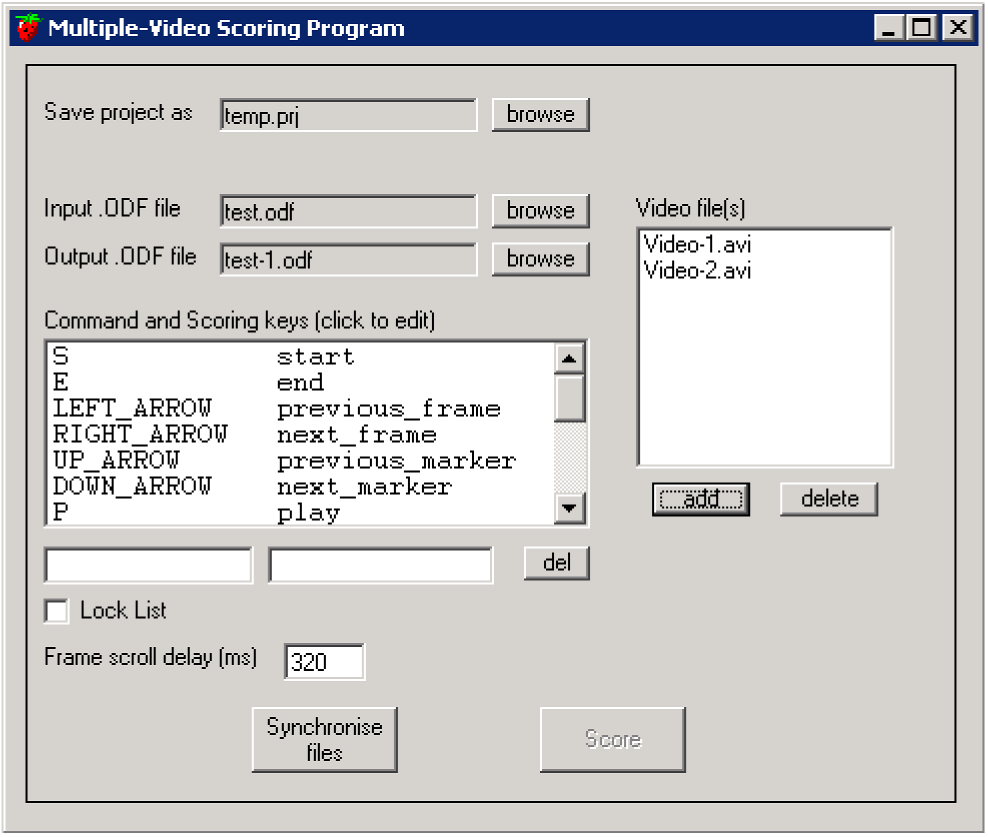
5.1.1 Opening display

When you start the programme you will be presented with a display showing 2 buttons:
"Create a new project" and "Load an existing project".

A project contains information for a scoring session such as the video and .odf files to be used, keys that have been assigned for controlling the scoring and other settings. This is stored in a project file with a .prj extension.

To create a project from scratch, click the "Create a new project" button. You will then be shown the project settings display.

5.1.2 Project-settings display



Multiple-Video Scoring Program

Save project as: temp.prj

Input .ODF file: test.odf

Output .ODF file: test-1.odf

Video file(s):
Video-1.avi
Video-2.avi

Command and Scoring keys (click to edit)

```
S              start
E              end
LEFT_ARROW     previous_frame
RIGHT_ARROW    next_frame
UP_ARROW       previous_marker
DOWN_ARROW     next_marker
P              play
```

□ Lock List

Frame scroll delay (ms): 320

Enter a name for the project file using the browse button.

Use the browse buttons to add the names of the input .odf file and output .odf file. For the input .odf file you would normally enter the name of the .odf file generated by the experiment programme. This will be copied into the output .odf file which will become the working file. The next time you run a session with this project this working file will automatically appear in the input .odf file box.

Add one or more video files to the listbox using the Add button. You can also remove video files from the listbox by selecting a file and clicking the Delete button.

5.1.3 Key commands

Key commands are used to control the video when you are scoring and you can set which key you want to use for each command. You can change any of these keys by clicking the appropriate item in the list box. The item will appear in the edit boxes below. The key that you wish to use for that command can be selected by pressing the appropriate key in the left hand edit box. To disable a command key, select the item then press the ESC key in the left hand edit box. <BLANK> will be shown in the listbox for that command.

5.1.4 Scoring keys

Scoring keys can be assigned similarly. To create a new scoring key, click the blank line below the command keys in the listbox, then press the key you want to use in left hand edit box and enter a label in the right hand edit box. Labels should not contain spaces.
The label is what will appear in the output .odf file when you press that key during scoring. For example you might select the key "1(keypad)" as the left-look scoring key and give it the label "left". When you have selected your command keys you can tick the Lock List tick box which will disable the listbox to prevent accidental changes.
A scoring key (not command keys) can be deleted from the list by selecting it and then clicking the del button.

5.1.5 The 'Frame scroll delay' textbox

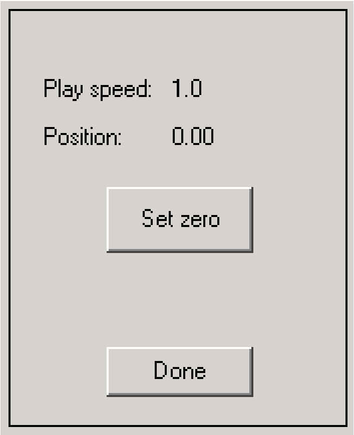
The 'Frame scroll delay' textbox is used in conjunction with the Next Frame and Previous Frame commands. When you press the Next Frame key, the video will jump to the next frame, then wait there for the time set in this textbox, then if the key is still depressed, it will scroll through frames at the rate set by the speed control keys.

5.1.6 Video / Trial synchronisation

Before you can score the video(s) you need to synchronise them with the start of the first trial. Click the "Synchronise files" button to do this.

5.1.7 Synchronisation display

The control panel below will be shown on the left of the screen.



Play speed: 1.0

Position: 0.00

Below that a list of command keys is displayed (see examples below).

The videos are displayed to the right of the control panel. You can move them about on the screen by dragging their caption-bars.

5.1.7.1 Command keys available for Synchronisation

You can control the videos using these command keys, assigned in the Project-settings section.
They are:

Start - jumps to the very start of the selected video
End - jumps to the very end of the selected video
Previous frame - moves selected video back by 1 frame.
Next frame - moves selected video forward by 1 frame

Note - if you hold down the keys for the previous 2 commands, they will play the video at the set play speed, backwards or forwards, for as long as the key is pressed.

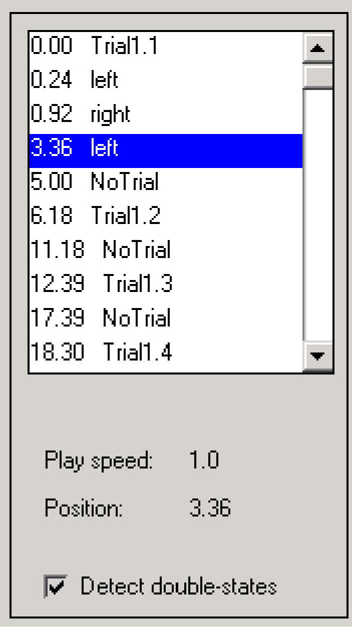
Zoom in - enlarges the selected video size on the screen.
Zoom out - reduces the selected video size on the screen
Slower - reduces the video play speed. Normal speed is 1.0
Faster - increases the video play speed

5.1.7.2 Synchronisation Procedure

To synchronise a video, first click the caption-bar above a video to select it.
The synchronisation commands only operate on the selected video. Then use the command keys to find the start of the first trial. When this is done, click the "Set zero" button to store the offset. Repeat this for each video. When all the synchronisation has been done, click the "Done" button to return to the Project-settings display.

5.1.8 Scoring display

When you click the "Score" button in the Project-settings display you will be shown the Scoring display. This will also create a copy of the input .odf file with a .bak extension, overwriting any previous .bak file with the same name, before copying its contents into the output .odf file.
A listbox on the left holds timing information from the .odf file.



```
0.00  Trial1.1
0.24  left
0.92  right
3.36  left
5.00  NoTrial
6.18  Trial1.2
11.18  NoTrial
12.39  Trial1.3
17.39  NoTrial
18.30  Trial1.4
```

Play speed: 1.0

Position: 3.36

☑ Detect double-states

Below that a list of command and scoring keys is displayed.
The videos are displayed to the right of the odf list. You can move them about on the screen by dragging their caption-bars. Most command keys operate on all videos together, but Zoom-in and Zoom-out commands operate on the selected video only. A video can be selected by clicking on its caption bar.

5.1.8.1 Command keys available for Scoring

You can control the video using the command keys, assigned in the project-settings section.

They are:

Previous frame - moves videos back by 1 frame
Next frame - moves videos forward by 1 frame

Note - if you hold down the keys for the previous 2 commands they will play the video at the set play speed, backwards or forwards, for as long as the key is pressed.

Previous marker - moves the videos to the next listbox position back from the videos current position.
Next marker - moves the videos to the next listbox position forward from the videos current position.
Play - plays the videos forward at the set play speed.
Stop - stops the video playing.
Zoom in - enlarges the selected video's size on the screen.
Zoom out - reduces the selected video's size on the screen
Slower - reduces the videos' play speed. Normal speed is 1.0
Faster - increases the videos' play speed

Scoring keys can be pressed at any time and the current frame time and key label will be inserted in the listbox. If the "detect double-states" tickbox has been ticked, a warning message will pop up if you enter the same score consecutively.

5.1.8.2 Listbox

If you click an item in the listbox, the video will jump to that position.

If you double-click an item in the listbox, or press the RETURN key while an item is selected, an edit box will appear and you can edit the label for that item.

If the first key you press in this edit box is a scoring key, the programme will automatically enter its scoring label. Press RETURN or click in the listbox to escape from the edit mode.

To delete an item from the listbox, select it then press the DEL key.

5.1.8.3 Quitting the programme

When you want to stop a scoring session, just close the programme with the **close button** at the top right.

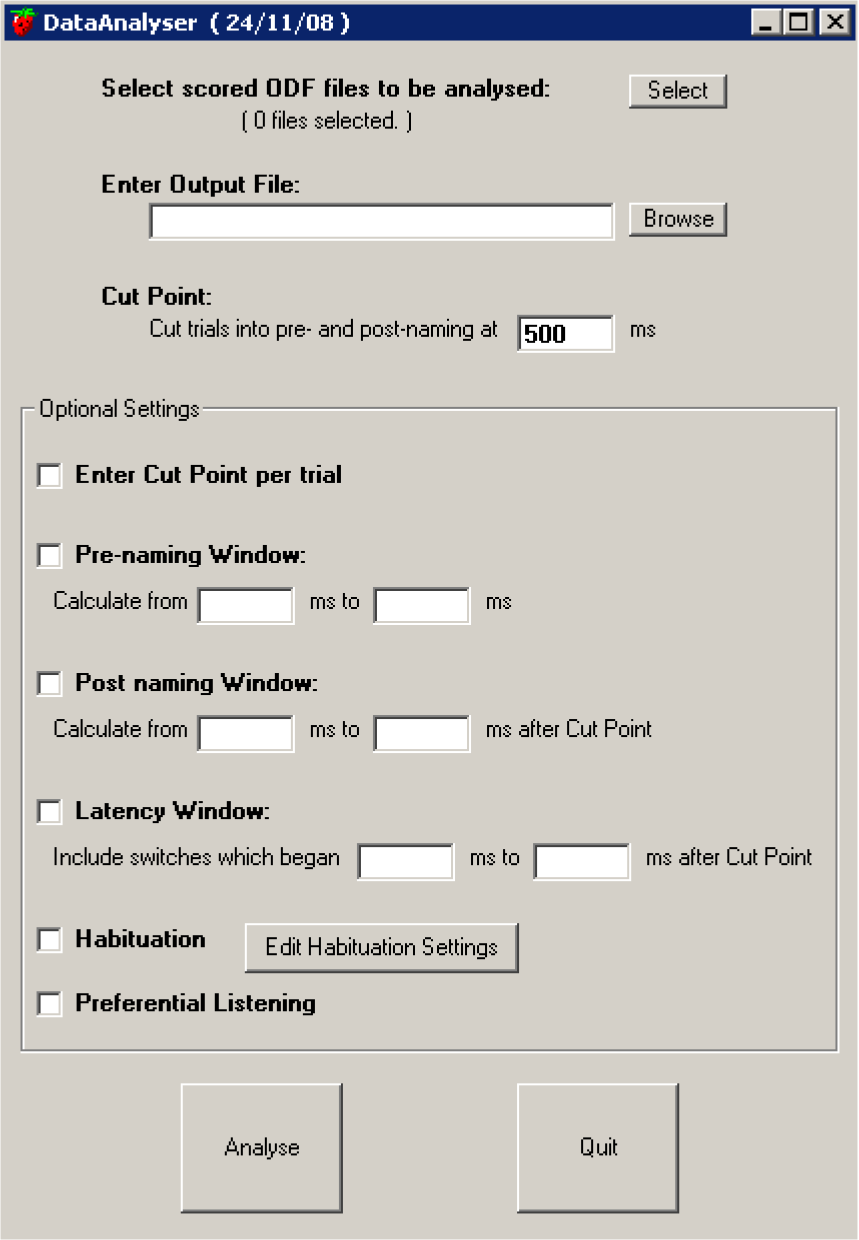
The programme will save the timing information from the listbox into the output .odf file, and also save the current settings in the project file, so that if you reload the project it will resume where you left off! ☺

6. Data Analysis

The next programme was created in order to avoid having to calculate results separately for separate measures (e.g. longest look, total looking, latency, etc.) from the coded looking data. Instead, the next programme uses the output from *score.exe* and calculates a range of measures for you.

6.1 Tool 4: DataAnalyser - Data analysis programme (DATAANALYSER.EXE)

This programme performs pre-analysis calculations on data from scored .odf files and their corresponding .txt files. See the 'Output Measures' section below for further information on the measures calculated. Conveniently, the output file is a comma-delimited text file that can be imported into statistics packages for further analysis.



Select scored ODF files to be analysed:
(0 files selected.)

Enter Output File:

Cut Point:
Cut trials into pre- and post-naming at 500 ms

Optional Settings

☐ **Enter Cut Point per trial**

☐ **Pre-naming Window:**
Calculate from ms to ms

☐ **Post naming Window:**
Calculate from ms to ms after Cut Point

☐ **Latency Window:**
Include switches which began ms to ms after Cut Point

☐ **Habituation**

☐ **Preferential Listening**

6.1.1 Front panel controls

Select scored ODF files to be analysed button
This button displays a panel where you can enter one or more .odf files for analysis. All the .odf files selected will be matched automatically with their corresponding input .txt files.

Enter Output File
Enter the name for the output file to receive the analysis results. If a file already exists, the programme will append the analysis results to that file, otherwise a new file will be created.

Note that if more than one input .odf file has been selected, the results from all of them will be appended to the specified output file.

Cut Point
Point in the trial at which it is predicted that target/distracter discrimination takes place. Used to divide measurements into pre-naming and post naming phases. The setting is entered in milliseconds from the start of the trial.

6.1.2 Optional Settings

Enter cut point per trial
This allows you to enter a different cut point for each trial. The programme prompts you for each trial's cut point as the analysis proceeds.

Pre-naming window
Use this to calculate a trial's pre-naming measures within the specified time period only. The window onset and offset parameters are in milliseconds from the start of the trial.

Post-naming window
Calculates a trial’s post-naming measures within the specified time period only. The window onset and offset parameters are in milliseconds from the cut-point.

Latency window
Sets the period within which latency values are to be included for analysis. The window onset and offset parameters are in milliseconds from the cut-point.

Habituation
Allows the programme to analyse habituation blocks. The programme recognises habituation blocks by the characters 'habit' at the start of the block comment field in the input .txt file. If the Habituation checkbox is left unchecked, habituation blocks will not be analysed.

Edit Habituation Settings
This button displays a panel where you can enter parameters controlling the analysis of habituation blocks.

Preferential Listening
Allows the programme to analyse preferential listening blocks. The programme recognises preferential listening blocks by the characters 'listen' at the start of the block comment field in the input .txt file. If the Preferential Listening checkbox is left unchecked then preferential listening blocks will not be analysed.

Analyse button
Starts the analysis process.

Quit button
Quits the programme.

6.1.3 Output Measures

6.1.3.1 IPL block measures

subjnum	Participant number
block	Block number
trialnum	Trial number
leftimag	Name of left image file
midimag	Name of middle image file
righimag	Name of right image file
word	Name of sound file
code1	Block comment
code2	Trial comment
targside	Target side (0=right, 1=left, 2=middle)
targimag	Name of target image file
cutpoint	Cut point

6.1.3.1.1 Total Looking Measures
Total amount of time spent looking at target or distracter

Pre-naming phase for targets and distracters

tottarpr	Total time looking at target
totdispr	Total time looking at distractor
totdifpr	Total target minus total distractor

Post naming phase for targets and distracters

tottarpo	Total time looking at target
totdispo	Total time looking at distractor
totdifpo	Total target minus distractor

Post-naming minus Pre-naming for targets and distracters

tottarch	Change in total target looking
totdisch	Change in total distractor looking
totdifch	Change in total difference

Pre-naming for left and right looks

totLpr	Total time looking Left
totRpr	Total time looking Right

Post-naming for left and right looks

totLpo	Total time looking Left
totRpo	Total time looking Right

6.1.3.1.2 Total Looking Proportion Measures Calculations

Proportion Measures
What do we calculate?
→ Proportion = T / (T + D)

Pre-naming
Total looking totarget pre
Total looking to distr pre
Proportion (t/(t+d) of target looks

Post naming
Total looking totarget post
Total looking todistr post
Proportion (t/(t+d)) of target looks

protarpr	Proportion of time spent looking at target, pre-naming (t/(t+d))
prodispr	Proportion of time spent looking at distracter , pre-naming (d/(d+t))

Post-naming

protarpo	Proportion of time spent looking at target, post naming (t/(t+d))
prodispo	Proportion of time spent looking at distracter, post-naming (d/(d+t))

Post-naming minus Pre-naming

protarch	Change in proportion of time spent looking at target (protarpo – protarpr)
prodisch	Change in proportion of time spent looking at distractor (prodispo – prodispr)

6.1.3.1.3 Longest Look Measures Calculations

Longest Look Difference Measure (T-D)
What we calculate:
→ Duration of the longest look to a target minus longest look to the distractor

Pre-naming

Llktarpr	Longest look to target, pre-naming
Llkdispr	Longest look to distractor, pre-naming
llkdifpr	**Difference between longest looks to target and distractor (T - D)**

Post-naming

llktarpo	Longest look to target, post-naming
llkdispo	Longest look to distractor, post-naming
llkdifpo	**Difference between longest looks to target and distractor (T - D)**

6.1.3.1.4 Further Measures

6.1.3.1.4.1 Post-naming minus Pre-naming

llktarch	Change in longest looks to target
llkdisch	Change in longest looks to distractor
llkdifch	Change in longest look difference (llkdifpo - llkdifpr)

6.1.3.1.4.2 First look measures

What we calculate:
Duration of first look to target or distractor

Pre-naming

flktarpr	Duration of first look to target
flkdispr	Duration of first look to distractor
flkdifpr	Difference between first looks to target and distractor (T - D)

Post-naming

flktarpo	Duration of first look to target
flkdispo	Duration of first look to distractor
flkdifpo	Difference between first looks to target and distractor (T - D)

Post-naming minus Pre-naming

flktarch	Change in first looks to target
flkdisch	Change in first looks to distractor
flkdifch	Change in first look difference (flkdifpo - flkdifpr)

6.1.3.1.4.3 Latency measures

'Latency' is the onset of a look relative to the cut-point

latlkdir	Direction of look at cut-point (1 = target, -1 = distractor, 0 = mid)
dir1sw	Direction of first look started after cut-point (1 = target, -1 = distractor) [check this]
dir2sw	Direction of second look started after cut-point (1 = target, -1 = distractor)
lat1sw	Latency of first look started after cut-point
lat2sw	Latency of second look started after cut-point
lattar	Latency of first look to target started after cut-point
latdis	Latency of first look to distractor started after cutpoint
latmid	Latency of first look to mid started after cutpoint

6.1.3.1.4.4 Number measures

Pre-naming

numtarpr	Number of looks to target
numdispr	Number of looks to distractor
numdifpr	Difference between number of looks to target and distractor (T - D)

Post-naming

numtarpo	Number of looks to target
numdispo	Number of looks to distractor
numdifpo	Difference between number of looks to target and distractor (T - D)

Post-naming minus Pre-naming

numtarch	Change in number of looks to target
numdisch	Change in number of looks to distractor
numdifch	Change in number of looks difference (numdifpo - numdifpr)

6.1.3.1.4.5 Pointing measures

ptanypr (point at any - pre)
pttarpr (point to target - pre)
ptdispr (point to distr - pre)
ptanypo (point at any - post)
pttarpo (point to target - post)
ptdispo (point to distr - post)

6.1.3.1.4.6 Time course measures

lkt0 - ... show the direction of looks throughout the trial in 40ms increments
(1 = target, -1 = distracter, 0 = mid)

6.1.3.2 Habituation block measures

subjnum	Participant number
block	Block number
trialnum	Trial number
leftimag	Name of left image file
midimag	Name of middle image file
righimag	Name of right image file
word	Name of sound file
code1	Block comment
code2	Trial comment
targside	Target side (0=right, 1=left, 2=middle)
targimag	Name of target image file
cutpoint	Cut point

Pre-naming
tottarpr Total time looking at target
avtarpr Average time looking at target in pre-defined window
protarpr Progressive percentage of time looking at target in predefined window compared to baseline window
critarprIndication of whether habituation criterion met or not (0=not met, 1=met)

Post-naming
tottarpo Total time looking at target
avtarpoAverage time looking at target in pre-defined window
protarpo Progressive percentage of time looking at target in predefined window compared to baseline window.
critarpo Indication of whether habituation criterion met or not (0=not met, 1=met)

6.1.3.3 Preferential Listening block measures

subjnum	Participant number
block	Block number
trialnum	Trial number
word	Name of sound file
code1	Block comment
code2	Trial comment
targside	Target side (0=right, 1=left, 2=middle)
cutpoint	Cut point

tottarpr	Total pre-naming time spent listening to sound
tottarpo	Total post-naming time listening to sound
acctmpr	Accumulated pre-naming time listening to sound
acctmpo	Accumulated post-naming time listening to sound
numtr	Number of trials taken to accumulate this amount of listening time.

7. Appendix

This appendix contains information on:

7.1 Visual List Files

Visual List Files allow you to run more than one visual stimulus per picture-slot per trial, by acting as a container for a sequence of visual stimuli. They can be used for trial or attention-getting stimuli.

7.1.1 Format

Visual List Files are ASCII text files with a .VLF extension. Each line in the file contains two fields, the name of a visual stimulus file, and a duration in milliseconds. The data are Tab or Space delimited, so any filenames containing space characters should be enclosed in quotes (" "). The filenames can be any of the recognised picture formats (.BMP, .GIF, .JPG, .AVI, .MPG). Other VLFs can also be used within a VLF but any duration values associated with them will be ignored - timing within the sub-VLFs will be used instead. A special keyword <BLANK> also allows you to insert periods with no visual stimulus visible in the picture-slot. Any line beginning with a colon character (:) is regarded as a comment and ignored by the programme.

<u>Example VLF:</u>

```
file1.bmp	1000
<BLANK>	1000
file2.jpg   3000
```

File1.bmp is displayed at the time defined by the <VISUAL_ONSET> tag in the .BEX file and will stay on for 1000ms, the picture-slot will then clear for 1000ms, then file2.bmp will be shown for 3000ms (or until the end of the trial, whichever
is sooner).

<u>Another example (with comment lines):</u>

```
  :Filenames containing space characters must be enclosed in quotes
"file one.avi"   1000
  :sub-vlf files contain their own duration information
file3.vlf
  :To ensure a file stays visible until the end of the
  :trial, just give it a very long duration.
file2.bmp	999999
```

7.1.2 Operation

VLFs are loaded at the start of a trial along with other stimuli. They are parsed by the programme and the component stimulus files loaded into memory. If a stimulus file is used more than once in a VLF, only one copy is loaded into memory to reduce loading time and memory space. The programme places a maximum limit of 50 visual stimulus objects in total per trial - that includes blanks and attention-getter stimuli. That also includes a maximum limit of 10 loaded video files per trial. The onset and offset times are stored in a 'schedule table' which is consulted by the programme during the trial timing loop, to tell it when stimuli should be displayed or hidden.

7.1.3 Animation

VLFs can be used as simple animation files. However if the frame rate is high or it uses a lot of images you'd probably be better off creating a video file. Bear in mind that loading a lot of seperate image files at the start of the trial could take longer than loading a video. VLFs can be looped in a trial by adding a <VIDEO_LOOP> tag to it in the .BEX file.

Note that if you loop a VLF, any videos within the VLF will also loop. Attention-getter stimuli will automatically loop, given half a chance. The VLF below would display a flickering image if used as an attention-getter stimulus.

```
 file1.bmp      100
<blank>         100
```

7.2 Sound List Files

Sound List Files are ASCII text files with a .LST extension, used to concatenate a list of sound files and silent intervals on-the-fly. They can save considerable disc space by combining a few elementary sounds together. The programme interprets this file when loading the trial stimuli and constructs a single WAV file in memory from the contents. It is important that all the components of the list are compatible...ie the bits-per-sample and sample-rate of the component WAV files should be the same.
The first line specifys an interval of silence (in msecs) between individual sounds. Subsequent lines contain the names of .WAV files or other .LST files that make up the required sound stimulus.

Below is an example .LST file to produce "Look! Bath!", with 500ms of silence between the two words:

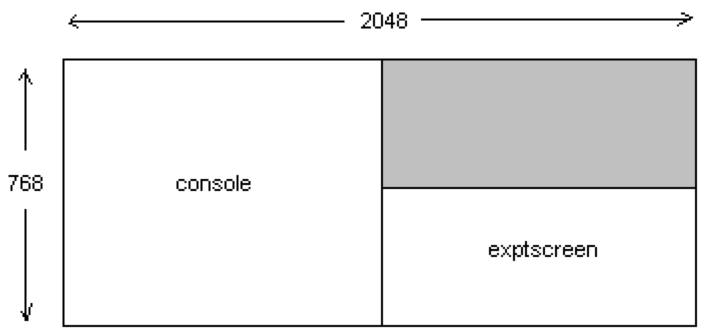
```
500
look.wav
bath.wav
```

7.3 Screen Areas and Picture Slots

7.3.1 Stimuli Display Area (experiment screen area)

The <EXPTSCREEN> tag (<EXPTSCREEN> *left top width height*) defines the area of screen where stimuli will be displayed. This area should be the area of the screen visible to the infant. For example, here at Lincoln a projector with a resolution of 1024 x 768 is connected to the PC's dual monitor card and used to display stimuli, but the top half of the screen is masked off so stimuli are only shown in the bottom half of the screen. The parameters for <EXPTSCREEN> would be set to define that visible part only. In our case:

```
<CONSOLE> 0  0  1024  768
<EXPTSCREEN> 1024  384  1024  384
```



The <EXPTSCREEN> settings are used for automatic placement of picture slots (see below), and determining left, middle and right locations (the programme divides this area into thirds).

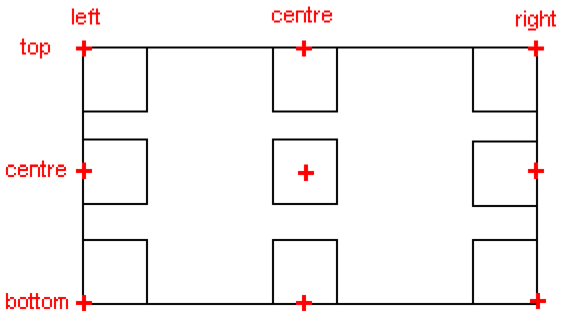
7.3.2 Picture Slots

With the picture slot tag (<PICSLOT> *mode xoffset yoffset*), the mode parameter determines 9 set positions within the area defined by <EXPTSCREEN>.

They are all combinations of horizontally left, centre, right and vertically top, centre, bottom.

0= top left
1=top centre
2=top right
3=centre left
4=centre centre
5=centre right
6=bottom left
7=bottom centre
8=bottom right

They can be used to place picture slots directly at these positions in the 'exptscreen' area. The red crosses below indicate the picture slot positions in the experiment screen area.

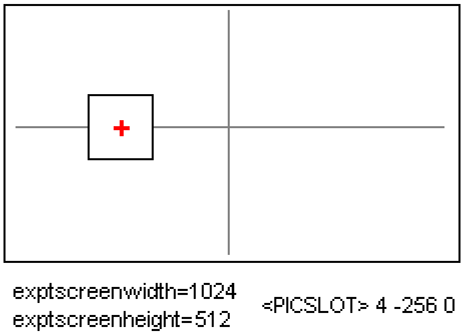


These positions also determine the coordinate system used. For example, selecting top-left will not only place the picture slot in the top-left corner, but also use top-left as the coordinate system for the stimulus that goes in that slot, i.e. the stimulus will extend to the right and down from that position.

Another example: Selecting centre-centre will place the picture slot at the centre of the experiment screen area and also use a coordinate system based on the centre of the stimulus (the stimulus will always be centred no matter what size).

Now, not everyone will want to use these set positions, in which case they can select the coordinate system they want with the mode parameter and adjust the actual position with the xoffset and yoffset parameters (default xoffset=0, yoffset=0 places picture slots in the positions above, depending on mode).

For example one could select centre-centre for the coordinate system (mode=4) and change the xoffset parameter to place the centre of the stimulus, say, midway between the centre and left hand side.



7.4 Files and Directories

In order to ensure portability of experiment files it is recommended that they be located in one of the directory structures below. When using either of these directory structures you only need to specify the name of stimulus files in a .BEX file and not their full pathnames.

1) All stimulus files are located in the same folder as the .BEX file that uses them. Output files from that experiment will also be saved to the same folder.

2) A 'root' directory (which can be located anywhere in file space) is created containing the following subdirectories:

CONFIG	for storing .BEX experiment configuration files.
IMAGES	for storing visual stimulus files
SOUNDS	for storing auditory stimulus files
LOGFILES	for storing .ODF, .TXT and .EVT output files