

HEALthy Brain Child Development (HBCD) Study : Task Creation, Coding, and Versions Manual

Compiled by Olufemi Shakuur Nyabingi

Updates by Jessica Norris

With the collaborative contributions of the Child Development Lab, HBCD
EEG Working Group and collaborators.

Last Updated: 4.1.2023

TABLE OF CONTENTS

INTRODUCTION

Preface

Task Selection

PART ONE: PARADIGM CREATION

Resting State

Face

Mismatch negativity

Visually Evoked Potential

PART TWO: E-PRIME CODING MANUAL

System Specs and General

Features

Resting State

Face

Mismatch Negativity

Visually Evoked Potential

APPENDIX

Version Release History

Additional Resources

Indigenous Faces Photo

Protocol

Face Races

Informal Face Race Validation

Benasich MMN Modification

MMN Trial List

Cedrus Polarity Blog Post

E-Objects and Corresponding Event

Flags EEG Parameters Graphic

Introduction

Preface

This manual seeks to compile information regarding the creation of the four EEG tasks developed for HBCD. The experimental designs and stimuli were contributed by the EEG working group and collaborators and the tasks files constructed in E-Prime at the Child Development Lab at the University of Maryland (UMD). Each of the tasks have undergone many iterations and varying versions (*See Appendix: Version Release History*). This document aims to document how the tasks are currently constructed and how past versions, constraints, and considerations have impacted how and why the tasks are designed as they are. The goal is 1) in the highly likely event that the tasks need editing or revisiting beyond their current state, this documentation should aid in the navigation of the tasks for future members of the project and 2) for folks who may be interested in the *nitty gritty* details of the task designs that are otherwise considered a bit excessive to include in other study documentation to have access to this information.

Neither this document nor the construction of the tasks in E-Prime is perfect; it is merely an attempt to document and explain portions of the task creation and process. In depth details regarding the administration of these tasks within the study as they relate to the visit protocol and participant interaction will not be included in this document, and interested readers should reference the linked [Official EEG Acquisition Manual](#). This document will not provide a detailed tutorial on the basics of E-Prime, and a basic knowledge of E-Prime will be assumed for *Part Two: E-Prime*. Resources for familiarizing oneself with the basics of E-Prime can be found in the additional resources portion of the appendix (*See Appendix: Additional Resources*). This document will instead serve to illustrate how the experimental paradigms above were realized within E-Prime, demonstrating the mechanisms and structure that allowed the tasks to run as intended.

Further official study information and materials can be found here: <https://hbcdstudy.org/>

Task Selection

The goal of EEG in HBCD is to mark windows of development within the child's first three years of life and capture maturation in attention, language, memory, visual and auditory perception, inhibitory control, and affective processing. The four EEG tasks were selected with this goal in mind, and their descriptions are as follows:

Resting state EEG provides information about neural oscillations and dynamics between oscillatory rhythms across development¹. Ontogenetic changes in oscillatory activity reflect underlying developing large scale neural networks associated with early critical self-regulatory, cognitive, and affective processes² and developmental outcomes³.

Auditory mismatch negativity (MMN) Task: speech stimuli have been shown to be an important index of future language abilities⁴. The structure of the task facilitates examining auditory evoked potentials, habituation/dishabituation to auditory stimuli, as well as perceptual narrowing in language processing.

Visual Evoked Potential Task (VEP) is a measurement of the primary visual cortex' response to visual stimuli. VEP amplitude and latency decreases with age during the first three years of life and has been associated with concurrent and later developmental measures. In addition, the morphology of the VEP likely reflects varying degrees of synaptic efficiency and as such, can be used as a readout of general cortical function⁵.

Face Task: Face processing abilities as well as the neural structures supporting face and object processing become increasingly specialized during the first years of life^G . ERPs will be recorded while infants view faces and objects using an oddball task designed to index different stages of processing including attention, perception, categorization, individuation and memory⁷.

¹ Deco et al., 2011, 2013

² Whedon et al., 2020

³ Gabard-Durnam et al, 2019; Jones et al., 2020

⁴Lachmann et al., 2005; Schulte-Korne et al., 1998 ⁵see ...LeBlanc et al 2015

^G Scherf & Scott, 2012; Markant & Scott, 2018

⁷ Barry-Anwar, Riggins & Scott, 2020

Part One : Paradigm Creation

Resting State

Stimulus acquisition

The stimulus is a 3 minute silent video (File format .AVI). There are two versions of the Resting State video, one for Visit 3 and one for Visits 4 and 6. The Visit 3 video is a compilation of different clips taken from Baby Einstein Baby Mozart videos. They display colorful toys and abstract images⁸. The Visit 4 and 6 video contains alternating 30 second clips of construction scenes and marble runs.

Experimental Paradigm

The video plays un-interrupted for 3 minutes. The task concludes after the video finishes.

Image: selected scenes from the Visit 3 Resting State video

Resting State



3 min. continuous video

Timing

Stimulus Duration: 3mins

Inter-stimulus Interval: N/A

Total Trial Length: 3mins

Task Background/History

A variety of videos were put forth as potential stimuli for the resting state task. The primary criteria were to include visual content that was sufficiently interesting to an infant that they may be able to sit calmly and mostly still during the duration of the task, but not overly stimulating as to prevent evoking too much unintended neurological response. Thus, the videos considered did not have sound, and a preference emerged amongst the working group against videos with explicit social-content and facial features (i.e: animated faces). A second resting state video was created for Visits 4 and 6 when it was determined that the Visit 3 video would not be sufficiently interesting to engage older infants and toddlers.

⁸ The Visit 3 resting state video was clipped from a larger video, provided by the team at Northwestern University (NU). The Visit 4 and 6 resting state video was created by the UMD team using internet-sourced video clips.

Face

Stimulus acquisition

Face images:

This photo set was specifically developed through recruiting participants and conducting a photo/video shoot in order to obtain uniformly framed and lit photos⁹. The original photoset was then expanded for HBCD to include photos to further service the need of the study demographic, such as incorporating photos of Indigenous individuals. This expansion was acquired through a second photoshoot conducted over Zoom (*see Appendix: Indigenous Faces Photo Protocol*).¹⁰ There are a total of 36 unique images in the set, with women all displaying neutral expressions, included from each of the following self identifying demographics: Indigenous, Black, White, Asian, Hispanic/Latino, South Asian (*see Appendix: Face Races*).

All of the face images were converted from color into grayscale value and uniformly equated for luminance using the SHINE toolbox in MATLAB. The faces were then cropped to fill uniform oval shapes to ensure the presentation of each image occupied the same space on the presentation monitor¹¹. To create the inverted face condition, a copy of these face images were simply rotated 180 degrees. Below shows a sample of the upright faces with an example of an inverted presentation.



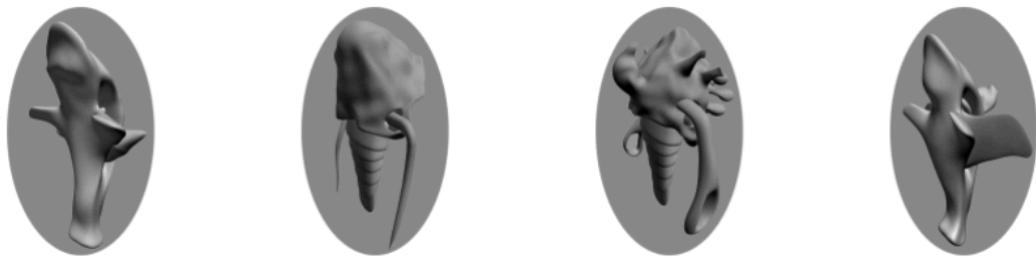
⁹ The original set of face images were provided by the team at University of Florida (UFL)

¹⁰ Indigenous identifying individuals were recruited and coordinated by the team at Oklahoma State University (OK). The participant video/photos and final stimuli quality control were coordinated between UMD and UFL.

¹¹ SHINING and oval cropping provided by team at University of Florida

Object Images:

Images of computer generated non-human “objects” were also used as contrasting stimuli for the face task. Similar to the face images, the object images were cropped to fill uniform oval shapes to ensure the presentation of each image occupied the same space on the screen¹². There are 30 unique objects.



Presentation size of each oval on the participant monitor, based on measuring the dimensions of the oval with a measuring tape pressed against the monitor, are below.

	Height (cm)	Width (cm)
Face	20.4	15.5
Object	20.7	16.0

Backgrounds:

The background images that are presented behind the stimulus object (face or object) were created by taking images of scenery that had relatively dense visual texture, converting them to grayscale, then concentrically blurring the image to create a gradated focal blur¹³. The innermost circle/oval of the image has the clearest image, and the outer ring is the most blurred. This was done to help draw and maintain the visual attention to the center of the screen¹⁴. There are five images in total, three are shown below.

¹² Provided by team at UFL

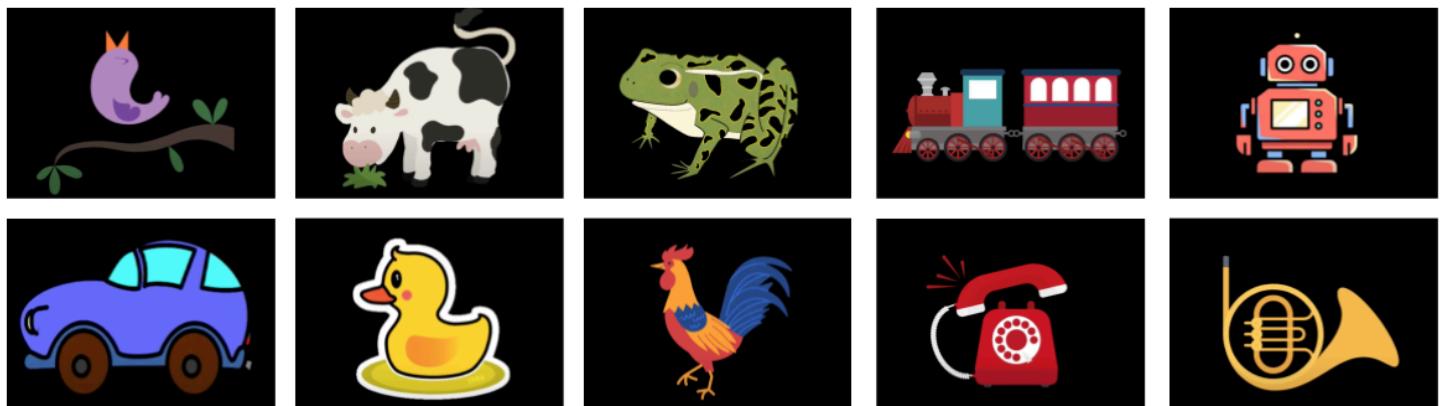
¹³ Provided by team at UFL

¹⁴ Similar to what was done here, Conte et. al. 2020



Attention-Getters:

The attention getters are 2 second video clips (file format .mp4) that are simple but bright animated pictures of a cartoon object over a black background with a sound playing in the background (i.e: frog image with frog ribbit sound)¹⁵. The object in the video slowly zooms in and out or contains some time of visual movement. There are 10 in total, the images from the videos are displayed below.



Experimental Paradigm

The task consists of 2 blocks:

- 1) 50 trials of upright faces & 50 trials of inverted faces
- 2) 50 trials of upright faces & 50 trials of objects

The four stimuli conditions for this task are listed below. The numbers correspond to how the conditions are coded in the EEG files.

1 UprightINV 2 Inverted 3 Object 4 UprightOBJ

¹⁵Created by team at UMD

Timing

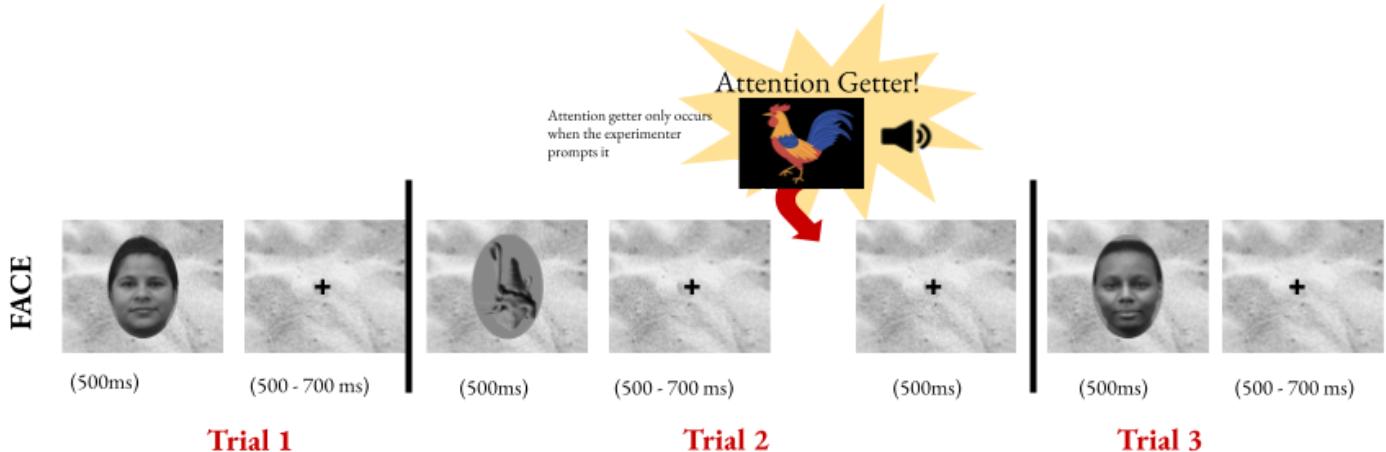
Stimulus Duration: 500ms

InterStimulus Interval: 600-700ms*

Total Trial Length: 1100 -1200ms*

*plus whatever additional time the experimenter takes to press the spacebar to progress between trials (i.e: natural button pressing time, using an attention getter, redirecting infant's attention).

A schematic of the task progression is below.



Task Background/History

Selecting the images set:

Other face image sets were considered before finally selecting the photo set created by the team at UFL. A variety of elements - including but not limited to, racial/demographic diversity of included participants, simplicity of appearance (i.e.: whether they had make-up, jewelry, hair pulled back vs out etc), image quality, and size of photoset - were considered. After exploring varying options, the UFL photoset best served the needs of the task. An informal survey was sent to members of the working group in order to get a preliminary validation of the photos and their legibility (see *Appendix: Informal Face Race Validation*).

Conditions:

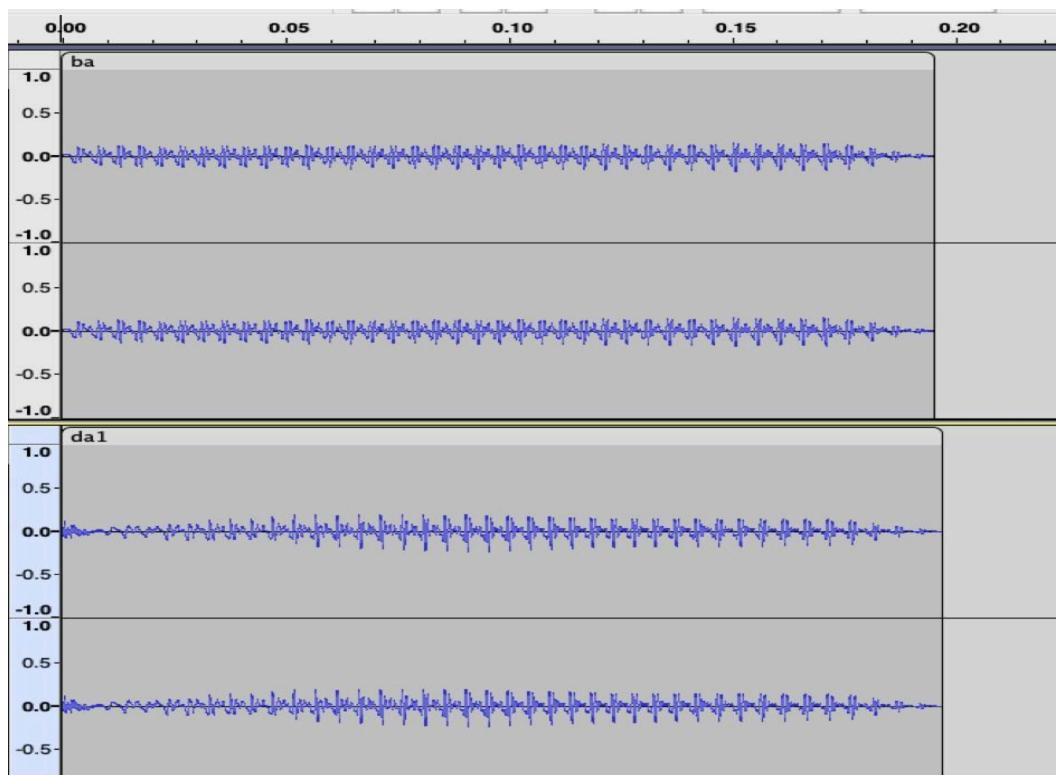
Earlier versions of the task included only 3 conditions, but the upright condition was split into two groups in order to specify which upright images had been presented in a block with inverted faces versus which upright images had been presented alongside objects. The possibility of interspersing the 3 conditions together within one block was briefly discussed, but opting for 2 discrete blocks was decided.

Mismatch Negativity (MMN)

Stimulus acquisition

Sound Files:

The stimuli for this task consists of two audio files of the naturally produced phonemes, /ba/ and /da/. They were recorded by a female native English speaker and equated for root mean square loudness¹⁶. The .wav files exact lengths are: ba file is: 196 ms long and da file is: 198 ms long. An image of their audio profiles (screenshot from Audacity) is included below.



Due to some technological limitations, constructing an audio task with timing precision became a difficult task (*see Task Background/History* below). As a result, very specific modifications to the sound files had to be made (edits completed in Audacity) to make them compatible with the E-Prime files. This included converting the file from mono to stereo, changing the project frequency of the sound to 48k HZ, and changing the type to 16-bit before re-exporting the sound files to .wav format (*see Appendix: Benasich MMN Modification* for instructions and more details) just remember they use 100ms sounds instead of 200ms.

¹⁶ supplied by the team at NU.

Video Distractor:

A video that is played as an option to give something for the infant to look at while they are listening to the MMN (file type is .MOV). There are 2 versions of the distractor video, one for Visit 3 and one for Visits 4 and 6. For the Visit 3 video, portions of the video clips come from the same video as the Resting State video with some additional content sourced from youtube videos to make the distractor video extra long. It is 15:41 mins long. The video for Visits 4 and 6 contains alternating 30 second clips of aquarium videos, safari videos, and clips from the animated "Hey Bear" videos. It is 12:06 mins long. Both videos are a bit longer than the MMN audio itself, so that the experimenter can have the video started before the MMN starts without having to worry too much about the video ending before the infant is done with the MMN.

Experimental Paradigm

The task consists of 667 trials:

569 Standard Condition - occurs the majority of the task (85.31% of the trials)

98 Deviant Condition- occurs infrequently (14.69% of the trials)

There are two conditions a trial could hold; either standard or deviant, depending on which sound is presented, and its frequency. There are two versions of the task that are counterbalanced for random selection. For one version, ba will be the standard sound, and da will be the deviant sound. For the other version da is instead the standard and ba the deviant. The task has a fixed trial list, meaning that everyone who completes this MMN will listen to the exact same order of standards and deviants. For example, trial 57 will always be a standard no matter the version type. However, which phoneme is assigned to be the frequent or infrequent sound will rotate. The first 9 trials of the MMN are standard sounds in order to help establish a pattern of expectancy for the start of the task.

See Appendix: MMN Trial List.

Timing Details

Stimulus Duration: 200ms

InterStimulus Interval: 820ms (Visit 3), 600 (Visits 4 and 6)

Total Trial Length: 1020ms (Visit 3), 800 (Visits 4 and 6)

A schematic of the trial progression for Visit 3 is below.



Task Background/History:

Technical limitations of audio timing accuracy:

After the UMD team had compared experiences across different labs, over years, across studies and across locations it became clear that timing precision of auditory stimuli, specifically with the E-Prime software configuration, was difficult to achieve and maintain. Stimulus presentation delays (how long it took for the sound to be presented to the participant) and drift (event flags inserted by E-Prime weren't matching the task timing) were common. The presentation accuracy depended on which version of E-Prime one was using, which version of the Windows one had on their E-Prime computer, and the sound chip capabilities of the computer. The E-Prime version and the PC specs were set by the study, so the design of the task had to fit within those parameters.

One attempted solution to this problem, affectionately referred to by the UMD team as "the Kludge", required stitching each of the individual sound files into one long .wav file such that playing one .wav through E-Prime would effectively play the entire MMN sequence (a version of this method where the long .wav and the distractor video were merged into one .MOV and played together, was also piloted for a considerable time period). With the timing already "frozen" in place by the long .wav the responsibility for E-Prime and the PC to accurately present each individual millisecond sound file, would be removed. While this addressed the issue of timing accuracy, it removed E-Prime's ability to distinguish which sound belonged to which trial, effectively hundreds of trials became 'just one long one' to E-Prime. As a result, a third device would have to externally add in trial markers. This was done either by hooking up the AV device by EGI (usually used occasionally to perform timing calibration and testing) permanently into the audio setup and amp in order to insert DIN2s. Eventually the StimTracker was configured to replace the role of the AV device and insert DIN2s.

After a long period of trial and error, a connection was made with another group who had very similar timing issues and who had finally managed to produce a MMN task in E-Prime 3 with timing accuracy (April Benasich's Lab and team at Rutgers University). It took their team a lot of troubleshooting and a lot of back and forth with the PST (company that manufactures E-Prime) support team, in order to come up with an E-Prime task that worked. They graciously shared that file, and the HBCD materials

were edited to make the experimental design work within the file requirements, hence the need to re-configure the .wav files as mentioned in the *Stimulus Acquisition* portion above.

Different experiment parameters:

Initially the MMN was supposed to be a more expansive auditory oddball experiment that contained the standard and deviant sounds as well as a novel sound (to evoke a P300). The novel sounds would include things like a crash sound, a moo, or a dog bark. The novel component was eventually removed because it would require a lot of trials to acquire this version of the oddball effectively. This version of the oddball could have taken 10-15 mins by itself and the end goal was to keep the total active acquisition time of all four EEG tasks under 15 mins for the infant.

Another important portion of this task's history is the piloting of timing and trial parameters in order to be able to effectively evoke an MMN response. Over 100 exploratory EEG sessions (including adults, children, and infants) were ran at UMD in order to optimize the varying combinations, of:

Total number of trials: 325, 400, 650, 666, 800

Which phoneme sounds: /ba/, /da/, /ta/

Novel sounds: yes or no

Jitter the inter-stimulus interval time: yes (300-500ms) or no

Stimulus length: 100ms vs 200ms

Total trial length: 600ms vs 1020ms

These experimental parameters arose out of the various configurations of established EEG MMN studies. A balancing act between total trial length combined with number of trials versus the total task length emerged, as it was understood many trials would be needed, but an infant can only sit so long.

Signal Detection:

The Cedrus StimTracker was proposed to be an external device for timing the HBCD tasks. It would remove the need for monthly system timing tests (using the EGI AV device mentioned above) as well as providing another way to keep track of stimulus presentation timing in the event of a technical failure or simply to provide a redundant system. This worked well and relatively simply for the visual tasks, but a lot of troubleshooting and configuration was needed to accommodate the audio stimuli of the MMN.

Since the StimTracker is outside of the process that reads in the .wav file and generates the sound signal, it does not 'know' when E-prime started playing the .wav, and must use signal detection to identify the onset of the sound stimulus. However, since the stimuli for the MMN are natural phonemes, they have more varying audio characteristics than something like say a computer generated pure tone. As a result, the sound files have varying rise times and audio patterns that make it difficult to set a detection threshold. The goal was to have the StimTracker insert the DIN2 at the onset of the sound file, but identifying the threshold that would trigger that detection was difficult. Make it too sensitive, and it detects fluctuations within one sound file as different sounds, make it too insensitive and you might miss onsets all together. This was exacerbated with the presence of highly varying novel sounds prior to the novel sounds being removed from the design. Strangely, this issue was not apparent when using the EGI AV device to detect sound onset, thus further troubleshooting to achieve harmony between the StimTracker and MMN task was needed. The goal was to modify some portion of the experiment or system to make the MMN sound files easier to accurately be detected at their onset. One attempt was to increase the audio levels of the sounds which required altering/scaling the amplitude of the .wav. The thought was that this would increase the signal that is inputted into E-Prime. While this did seem to help the issue, too much scaling up of the amplitude significantly distorted and garbled how the .wav file sounded, which was unacceptable for this task. The next attempt and final solution ended up being to change the digital gain of the Chronos (the Chronos mediates the sound production in the E-Prime task). This increased the signal production, but on the output side (chronos is sending out an increased signal to StimTracker). This eventually allowed the MMN task to send out a signal that was robust enough that the StimTracker could always detect the sound's onset.

Alas, there is another component to the signal detection issue. In the process of troubleshooting it was discovered that there was another discrepancy. Some portion of the MMN event flag in the .mff seemed to suggest that the marking in the sound file was being inserted after the sound had occurred, not at its onset. After some sleuthing and collaboration between the UMD team, the Cedrus team, and the EGI team, it is discovered that EGI and Cedrus had been using opposite polar logics. Meaning, Cedrus was marking its signal detection on the onset of the signal and EGI was marking the offset of the signal. Despite Cedrus-EGI interfacing having been established for years, the issue only seemed to become apparent in this MMN troubleshooting because the sound files were 200ms long. The majority of StimTracker-EGI users were likely using much shorter pulses, and the precision/discrepancy was not noticeable. A firmware release was made by Cedrus to be able to toggle polarities in order to accommodate this difference *see Appendix: Cedrus Polarity Blog Post*. This change, paired with the Chronos digital gains edit, and the StimTracker configurations mentioned in the EEG Acquisition Manual, was able to solve the signal detection problem with the MMN.

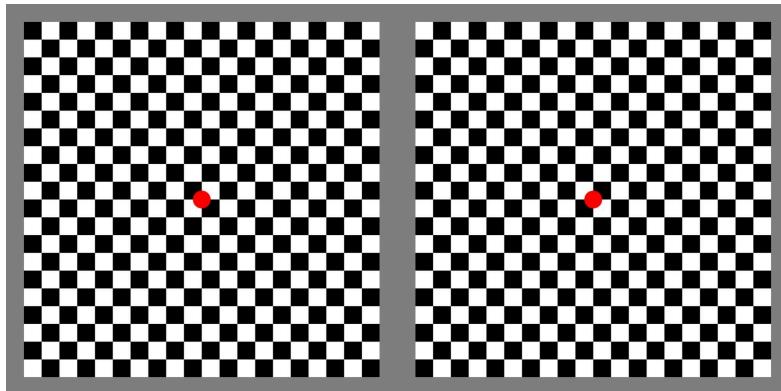
Modifications for Visits 4 and 6:

During the development of piloting materials for Visits 4 and 6, it was determined that the 11 minutes the task requires was too long for the attention span of older infants and toddlers. Furthermore, Elizabeth Norton's team at Northwestern University supplied data suggesting that shorter inter-stimuli intervals (ISIs) were needed to elicit the MMN ERP in older infants and toddlers. For these reasons, it was decided to shorten the ISI from 820ms to 600ms in Visits 4 and 6. This shortened the overall task length from 11 minutes to 8 and a half minutes. The ISI was kept at 820ms for Visit 3.

Visually Evoked Potential (VEP)

Stimulus acquisition

These images are 20x20 checkerboards composed of black and white squares, with a gray background, and a red circle in the center of the board¹⁷. There are two checkerboards in total and they are the inverses of each other. Where there is a white square in checkerboard A, there is a black square in checkerboard B. This is easiest to see if you compare corner squares between checkerboards. Due to the display parameters and aspect ratio of the monitors that are used to display the task, the presentation of the squares stretches slightly from squares to rectangles when displayed through E-Prime (1x1cm squares become roughly, 1.4cm (width) x 1.2cm (height)). This discrepancy is not ideal, but it was determined that it should not have an impact on the VEP, as the pilot data showed no issues.



Experimental Paradigm:

There are 120 trials.

One checkerboard equals one trial. The trials alternate between checkerboard A and B.

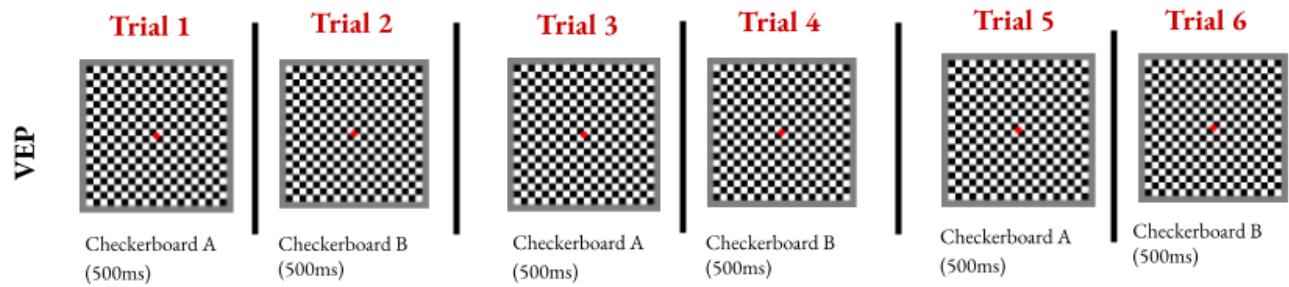
Timing:

Stimulus Duration: 500ms

InterStimulus Interval: N/A

Total Trial Length: 500ms

¹⁷ Checkerboards shared by Laurel Gabard-Durnam at Northeastern University



Task Background/History

From a design and stimuli perspective this task has remained relatively stable. Previous versions have included more trials but the task was cut down to its current length when it was determined fewer trials were effective.

Part Two: E-Prime

System Specs and General Features

These tasks were created using these softwares and versions:

E-Prime version 3.03.80

Net Station version 5.4.1

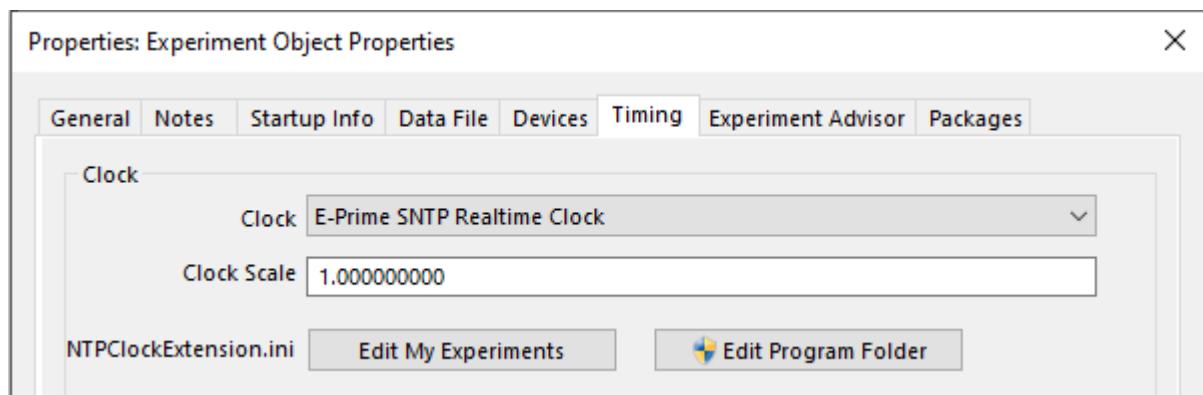
See Appendix: EEG Parameters Graphic for additional information on what equipment is used.

StimTracker

The Cedrus StimTracker is used as an additional way to detect stimulus onset and timing within a task. For visual tasks, it uses a photo sensor that is placed on the screen. The photo sensor detects the change in light from the screen. The use of white and black boxes in the corner of the screen are used as on/off switches throughout some of the tasks. The appearance of the white box indicates the onset of that slide. The StimTracker is then configured to transmit this data to Net Station as a DIN3 (Digital INput #3). This timing mechanism exists outside of the E-Prime timing, and can be used as timing-calibration as well as a redundant timing system. The StimTracker can also detect audio onset for the MMN via connection to the Chronos and speakers. The resultant event flag for the audio detection is DIN2 (Digital INput #2). More details on how the Stim Tracker is configured is available in the EEG Acquisition Manual.

SNTP Timing

This is the timing setting suggested by PST (E-Prime Software creators) and applies to each task file.



Startup Info

These tasks include the standard default prompts asked by E-Prime with the addition of a prompt that allows the experimenter to enter the DCCID/LORIS ID of the participant. Net Station requires that certain prompts (Age, Sex, Handedness, Researcher ID) are collected from E-Prime or it will not allow the Net Station session to record. However, extra prompts at the start of the task complicates acquisition so extraneous prompts were disabled (indicated by the red question mark with a strike through it). This means the prompt will not be asked at startup, and the prompt's default value is recorded instead. HBCD does not collect any of that info through E-Prime, so the entry of the default value doesn't matter. Originally, the Subject field was going to be used to collect the HBCD participant's ID, but two problems arose. One, the value entered into the Subject field, although logged in the edat, does not get automatically transferred to Net Station or the .mff. And two, although E-Prime has an internal digit cap at 2147483647, meaning values above this number are not permitted, the E-Prime extension for Net Station (EENS) has a much lower digit cap, at 32767. The DCCIDs would exceed this numerical limit which would result in an "overflow" runtime error. As a solution, the DCCID is collected at the start up, and logged in the CellList so that the participant's ID will travel with the data for pre-processing when it gets imported into MATLAB. The startup information also requests the visit number, so that this information will travel with the data for pre-processing.

Properties: Experiment Object Properties

X

General Notes Startup Info Data File Devices Timing Experiment Advisor Packages

Startup Info Parameters

Name	Prompt Text	Data Type	Default
DCCID	Please Enter the Par...	Numeric	1
Subject	Please enter the Su...	Numeric	1
Visit	Please enter the Visi...	Choice	V04
Session	Please enter the Ses...	Numeric	1
Group	Please enter Subject...	Numeric	1
Name	Please enter Subject...	String	
Age	Please enter Subject...	Numeric	0
Sex	Please enter Subject...	Choice	male
Hand...	Enter Subject's Han...	Choice	left

Add...

Remove

Edit...

Move Up

Move Down

- Use default values and do not prompt the user for all enabled startup info parameters
 Display a summary to confirm the values of the startup info on Run

External StartupInfo Parameters

- Load values from external startup info files into experiment context

Edit Local Values

Edit Global Values

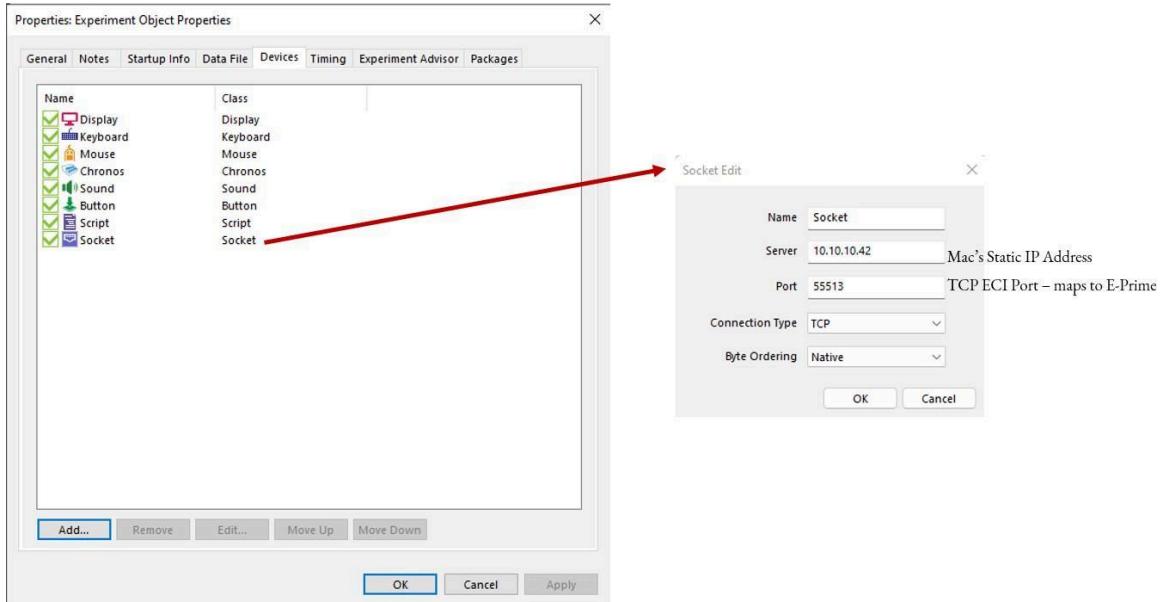
OK

Cancel

Apply

Devices

This listing of these devices is hierarchical, for example, since the Chronos device is used to administer the sound functionality in these tasks, it must be listed above the Sound device. The Socket device in these tasks references the Net Station-Mac to E-Prime PC connection. For the MMN task, the Chronos must come first, but for other tasks, it's less strict.



General Features:

All of the tasks utilize some or all of the following elements.

IDname

A parameter that is requested in the start up information of the task and is to log the DCCID (the unique ID) for the HBCD participant, via assigning it to an attribute that gets stored in the cell list. This script also logs the visit number and assigns it to the cell list.

```
LORID = c.GetAttrib("DCCID") 'assigning that variable the based on the LORIS ID entered in startup info'
CellList.SetAttrib 6, "CellLabel", LORID

Visit = c.GetAttrib("Visit") 'assigning that variable based on the visit number entered in startup info'
CellList.SetAttrib 7, "CellLabel", Visit 'logging the Visit variable in the CellList'
```

NSSendEvents

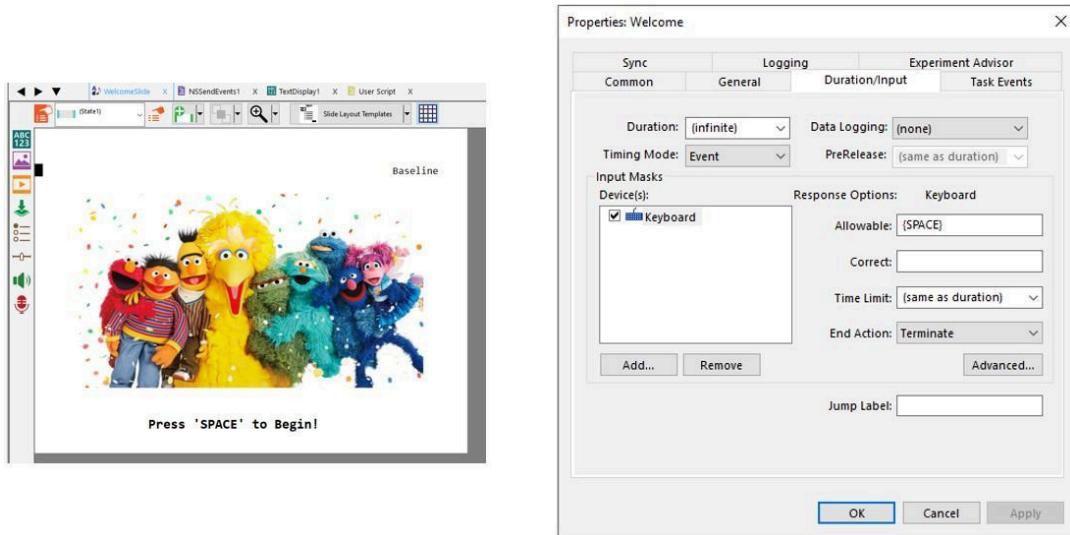
This is an inline script that sends the event flag from E-Prime to Net Station. The E-object containing this script may have slightly different names depending on the task. It consists of multiple commands depending on the objects in the procedure. It will have a 'SendTrialEvent' for every E-object in the procedure that has an event flag that needs to be logged. But will also have one SendTRSPEvent (TRSP stands for TRial SPecific Event) at the end that conveys additional trial information including the trial conditions.

```
1 'Sending event flags to NS to tell it when the Basline video begins'
2 NetStation_SendTrialEvent c, TextDisplay1
3
4 'send TRSP'
5 NetStation_SendTRSPEvent c, TextDisplay1
```

'Welcome' and 'Goodbye' slides

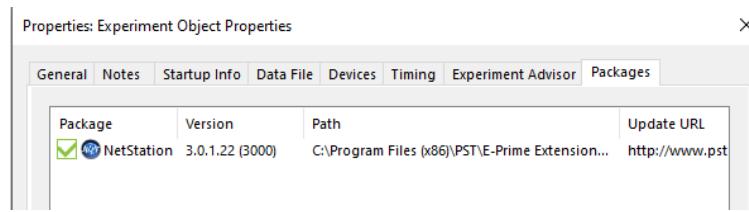
These slides visually convey the beginning or the end of the task, and only terminate when the experimenter is ready to continue. They help provide contextual cues for the experimenters in the participant room as well as the adult accompanying the baby. The Welcome screen is also helpful in that it gives the experimenter at the computer side one last checkpoint after loading the task but before actually presenting stimuli. Since transitions between tasks often occur quickly, the welcome slide can help cue the experimenter to transition back to active session conditions after a break, or give the experimenter in the room a clear opportunity to signal to the control room that they

need additional time before they start presenting the task. The welcome and goodbye screens are functionally the same, the only difference being that the sentence on the bottom changes from “Begin” to “Finish” or “End”. The Face tasks has an additional “Block Complete” slide that acts as a transition point between blocks if needed.



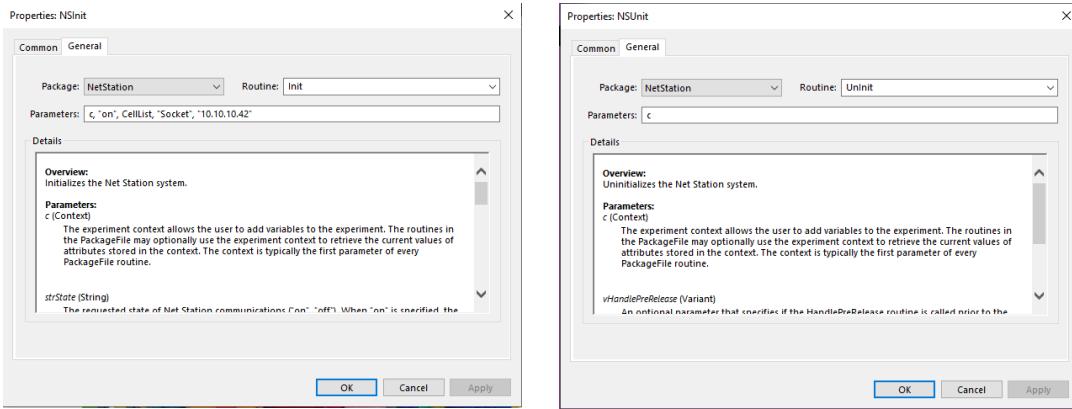
E-Prime Extensions for Net Station (EENS) Package Commands

This is the E-Prime package that facilitates the communication between E-Prime and Net Station. The package is contained on a license key usb and must be enabled in the experiment properties in the Packages tab. You can find more information about the commands in the E-Prime Extension for Net Station user manual (*see Appendix: Additional Resources*).

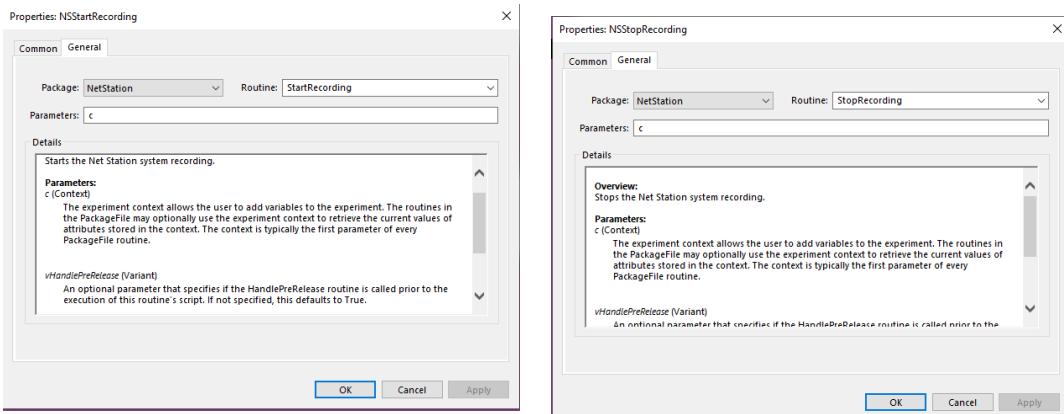


Once the EENS package is installed, a variety of package commands can be called in the E-Prime procedures. Although the specific name of the E-object may vary per task, the package commands utilized are:

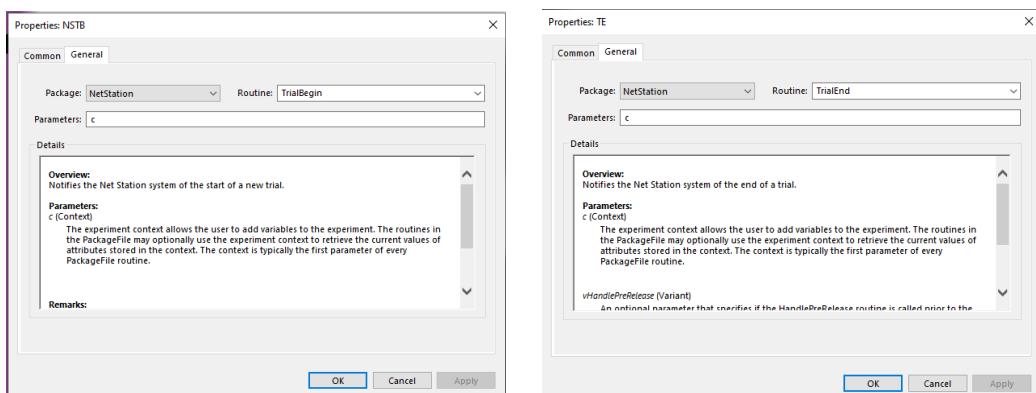
- Net Station Initialize & Net Station Uninitialize: package command that establishes a connection between E-Prime and Net Station.



- Net Station Start Recording & Net Station Stop Recording: package command that starts and stops Net Station recording.

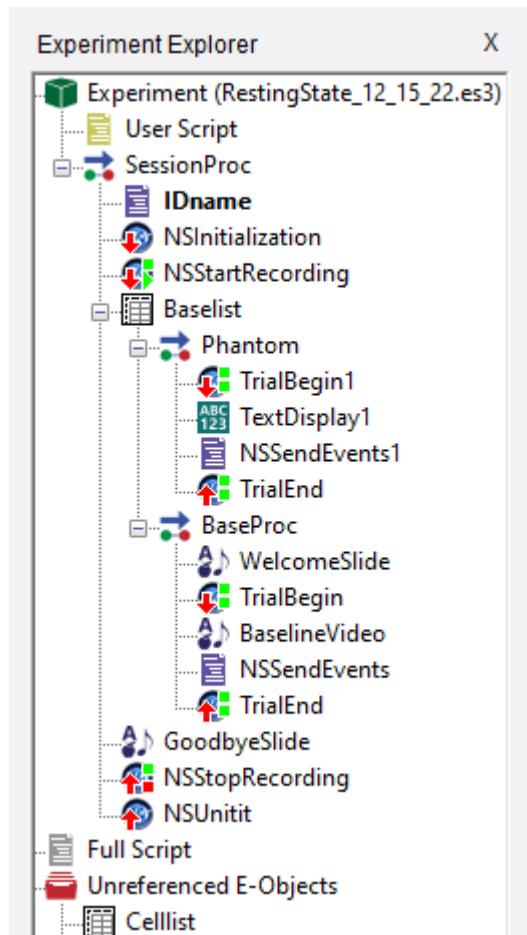


- Net Station Trial begin & Net Station Trial End: Marks the begin and end of the trial.



Resting State

An overview of the full experiment procedure is shown below.



User Script

The only contents of this user script include a line to create "LORID" which is used in the process of creating the IDname and visit number features. *See Part Two: E-Prime General Features: IDname.*

```
1  ''' To allow the LORIS ID adn Visit number to be logged in NetStation via CellList"
2  Dim LORID As String
3  Dim Visit As String 'to label NetStation and EDat files with the visit number
```

IDname

See Part Two: E-Prime General Features: IDname.

NSInitialization

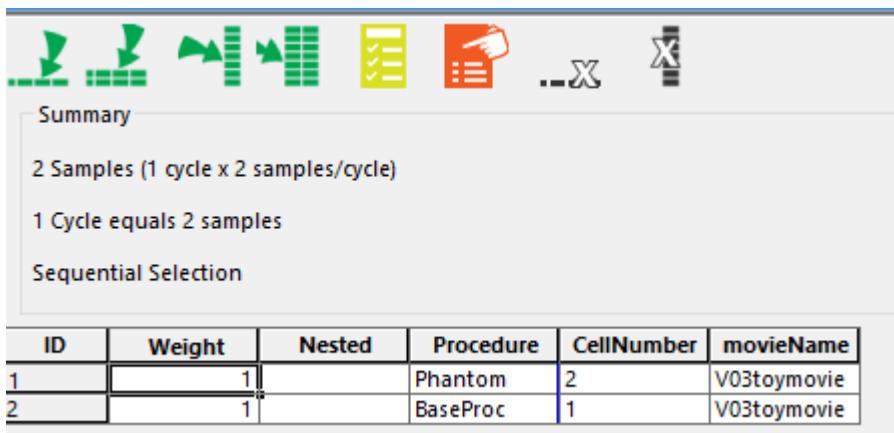
See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

NSStartRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

Baselist

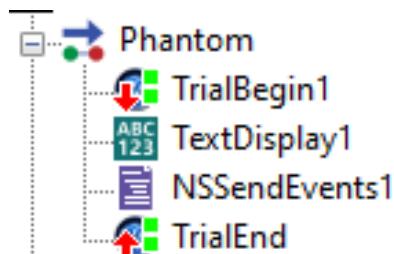
This list contains the two trials that occur within the task. The first, phantom, is a quick 'dummy' trial that was created to ensure that an event flag gets sent within the task, even if the trial containing the video gets interrupted. The second, BaseProc, contains the resting state video. Since BaseProc contains the entirety of the baseline video (3mins), if the participant has to end the video early, the trial would be interrupted and the Net Station Send Events command would never initiate (E-Prime wouldn't have made it to that portion of the procedure). As a result, the whole task would be missing any event flags. The Phantom trial prevents that. This list also specifies which stimuli video was played for the participant. This information is sent to NetStation as a TRSP.



ID	Weight	Nested	Procedure	CellNumber	movieName
1	1		Phantom	2	V03toymovie
2	1		BaseProc	1	V03toymovie

Phantom

"Dummy" trial referenced in Baselist.

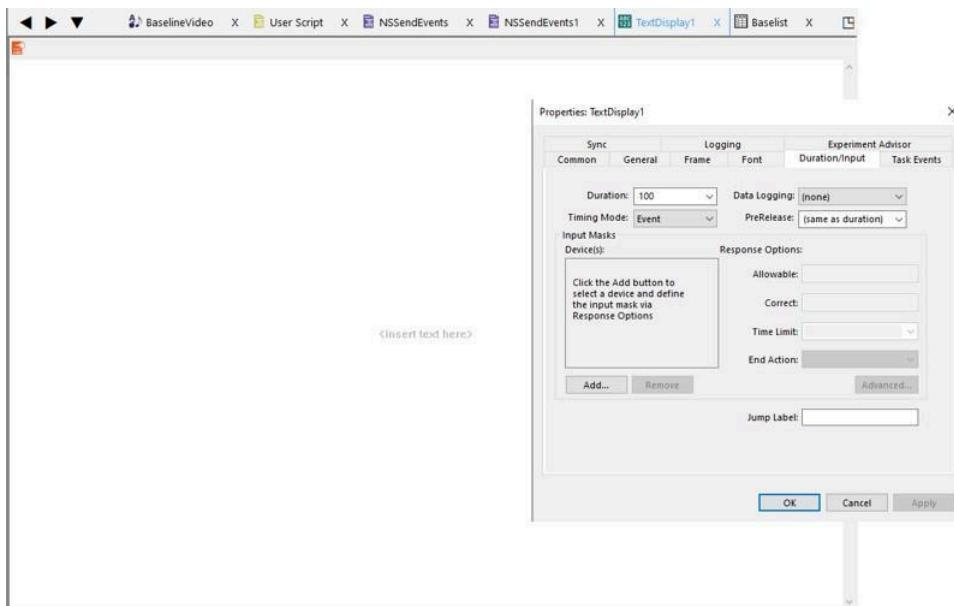


TrialBegin1

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

TextDisplay1

This slide flashes a blank white screen for 100ms. The sole purpose is to add some content to the phantom trial for the resting state task. It is an E-object to house the event flag.



NSSendEvents1

Sending the event flag for the phantom trial. See Part Two: E-Prime General Features: NSSendEvent

A screenshot of the E-Prime 'User Script' editor showing a script named 'NSSendEvents1'. The script contains the following code:

```
1 'Sending event flags to NS to tell it when the Baseline video begins'
2 NetStation_SendTrialEvent c, TextDisplay1
3
4 'send TRSP'
5 NetStation_SendTRSPEvent c, TextDisplay1
6
```

The code uses the 'NetStation_SendTrialEvent' and 'NetStation_SendTRSPEvent' functions to send event flags to the Net Station.

TrialEnd

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

BaseProc

The procedure that contains all the components of the Resting State Task.



WelcomeSlide

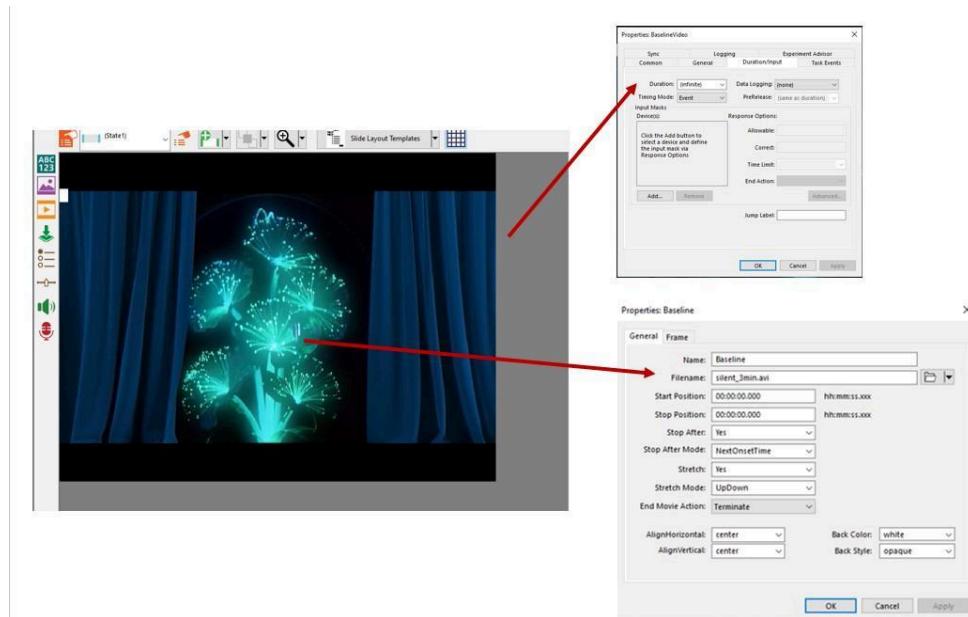
See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

TrialBegin

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

BaselineVideo

This is a slide object, with a movie sub object placed on the slide. The slide object itself has an infinite duration, whereas the movie object within, is set to terminate once the video is over (3 mins).



NSSendEvents

Sends Event Flags for Baseline trial. See Part Two: E-Prime General Features: NSSendEvent

```
1 'Sending event flags to NS to tell it when the Basline video begins'
2 NetStation_SendTrialEvent c, V03movie
3 'send TRSP'
4 NetStation_SendTRSPEvent c, V03movie, KeyList
5
```

TrialEnd

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

GoodbyeSlide

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

NSStopRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

NSUnitit

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

Celllist

Logs the video trial, visit number, and the LORIS ID.

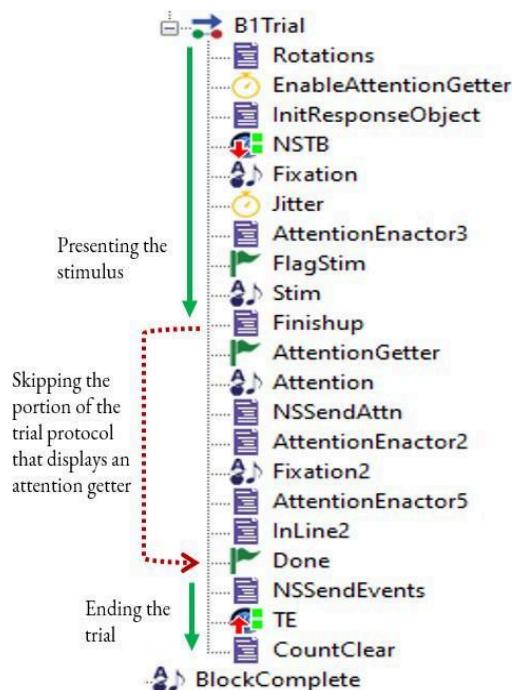
The screenshot shows the Celllist interface. At the top, there are several icons: a green arrow pointing right, a red arrow pointing right, a blue arrow pointing right, a yellow grid, a red clipboard, and a grey X. Below these icons is a section labeled "Summary". Under "Summary", the text "3 Samples (1 cycle x 3 samples/cycle)" and "1 Cycle equals 3 samples" is displayed. Below this is a section labeled "Sequential Selection". A table follows, with columns: ID, Weight, Nested, Procedure, CellNumber, and CellLabel. The data in the table is as follows:

ID	Weight	Nested	Procedure	CellNumber	CellLabel
1	1			1	Baseline
2	1			100	[LORID]
3	1			2	[Visit]

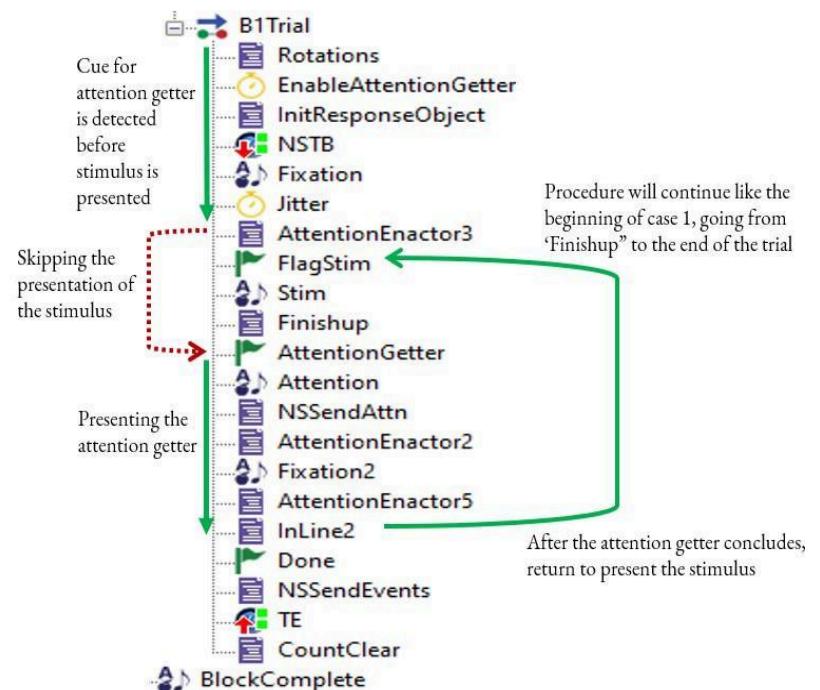
Face

Due to the nature of the Face task's ability to interrupt one portion of the procedure and jump to a different portion in order to execute the use of an attention getter, the events of a trial may not occur in the exact order listed from top to bottom. There are two scenarios that can occur each time a trial is presented in the face task that will dictate how the trial procedure will progress:

1) The attention-getter is not used



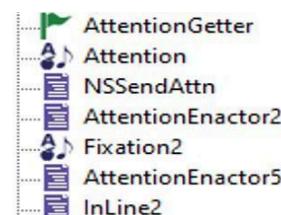
2) The attention-getter is used



Both of these scenarios are based primarily on the two following sequences below:



Sequence for stimulus presentation



Sequence for attention getter presentation

However, for the purpose of this manual, the E-objects will be described in the order of their appearance in the session procedure, in order to be consistent with the other portions of this document.

Startup Info - Face Task Specific

In addition to the prompts listed in *Part Two: E-Prime: System Specs and General Features* this task has an added prompt of "Version". It allows the experimenter to pick which block they would like to run in the event that they need to restart a portion of the task.

User Script

The user script establishes several variables that will be further customized for use later on in the experiment. They are established here for either clarity or because the variable needs to be created early on in the experiment hierarchy. The comments in the inline explain each component.

```
1  *** ATTENTION GETTER FUNCTIONALITY****  
2  Dim theSlide As Slide 'Declare a variable that can be assigned to the Stimulus object.  
3  Dim nResponseCount As Integer 'Declare variables for holding the current response count  
4  Dim theResponseData As ResponseData 'Declare variables for holding the current response data.  
5  Dim UnrefSlide As Slide 'Global variable to hold Unreferenced Slide name  
6  Dim theResponseObject As Wait 'Global variable (Trial pause mechanism)  
7  
8  
9  'To Randomly rotate the attention getter  
10 Dim attentiongettervars As String 'establishing a variable to hold the background filename to be called for that/those trial(s)'  
11 Dim AttnArray(9) 'creating an array from 0 to 9, of all the attn getters available so they can be randomly selected  
12 Dim currAttnIndex As Integer 'new variable declaration to keep track of current attention getter array index  
13 Dim j As Integer 'new variable declaration to keep track of current attention getter array index in debugging checks  
14  
15  
16 '***** Rotating Background Functionality *****  
17 Dim Block1Counter As Integer  
18 Dim backgroundvars As String 'establishing a variable to hold the background filename to be called for that/those trial(s)'  
19 Dim BackArray(4) As String 'creating an array from 0 to 4, of all the background filenames available, so that they can be randomly selected  
20 Dim StartTime As Long 'used to initialize a timer system to keep track of when backgrounds should rotate'  
21 Dim currBackImageIndex As Integer 'new variable declaration to keep track of current background image array index  
22 Dim i As Integer 'new variable declaration to keep track of current background image array index in debugging checks  
23  
24 '*** To allow the LORIS ID and visit number to be logged in NetStation via CellList"  
25 Dim LORID As String  
26 Dim Visit As String 'to label NetStation and EDat files with the visit number  
27  
28 '*****ITI*****  
29 Dim waitlengthvars As Integer  
30  
31 '***Restart block selection****  
32 Dim counterbal As String 'creating a variable to call a particular block based on counterbalance selected'
```

SessionProc

All of the task is contained within this procedure.

BlockSelect

This inline script takes the info that the experimenter selected in the start-up prompts (Standard, (RESTART ONLY) Upright/Inverted, or (RESTART ONLY) Upright/Object) and selects which case should be executed. If standard was selected the SessionList will enact both the blocks as usual, via random selection. But if one of the other two options were selected, this inline script assigns the opposite block of what was selected, to have a weight of 0, meaning that block will not run.

```
1  """ Taking the selected counterbalance option from startup info and running that particular block"""
2  counterbal = c.GetAttrib("Version") 'assigning that variable the based on the response selected in startup info'
3
4
5  Select Case c.GetAttrib("Version")
6    Case "(RESTART ONLY) Upright/Inverted"
7      SessionList.SetWeight 2, 0
8    Case "(RESTART ONLY) Upright/Object"
9      SessionList.SetWeight 1, 0
10   End Select
11
12 SessionList.Reset
```

IDname

See Part Two: E-Prime General Features: IDname.

NSInit

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

NSStartRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

Welcome

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

BackgroundEnable2

This inline script creates an array containing the five background images. It then randomizes the array. The experiment will cycle through the backgrounds in this random order for the duration of the experiment. It also sets the variable tracking the background image index to 0, thereby setting the image counter to be the start of the array.

```
1  '''This inline script establishes the array that the random selection of backgrounds will come from'''
2
3  'defining the array of background filenames'
4  BackArray(0) = "B10.png"
5  BackArray(1) = "B11.png"
6  BackArray(2) = "B12.png"
7  BackArray(3) = "B13.png"
8  BackArray(4) = "B14.png"
9
10 'debugging statement put in during development, displays array index of each image before randomization
11 'Debug.Print "ORIGINAL Array Order: "
12 'For i = 0 To UBound(BackArray)
13   ' Debug.Print "background image assigned to array index " & i & " = " & BackArray(i)
14 'Next i
15
16 'Randomize the array
17 RandomizeArray BackArray
18
19 'debugging statement put in during development, displays array index of each image after randomization
20 'DEBUG: Output the new array order
21 'Debug.Print ebLf & "NEW Array Order: "
22
23 'For i = 0 To UBound(BackArray)
24   ' Debug.Print "background image assigned to array index " & i & " = " & BackArray(i)
25 'Next i
26
27 'Initialize the variable that tracks which array index is being used to set the background image to index 0
28 currbackImageIndex = 0
```

TimeMarker1

This inline script uses the clock function to make a reference for time dependent functions (i.e. the rotating attention getters and rotating background).

```
1  '''Time Marker For Time Dependent functions : rotating background and rotating attention getter'''
2  StartTime = Clock.Read 'marking the moment right before the stimuli appears'
3
4
```

AttnRotation2

This inline script creates an array containing 10 attention getter videos. It then randomizes the array. When the experimenter cues the attention getter, EPrime will cycle through the attention getters in this random order for the duration of the experiment. The script also sets the variable tracking the attention getter index to 0, thereby setting the counter to be the start of the array. In the August 2023 updates to the task, this script was moved out of the Block1 and Block2 procedures and into the main SessionProc procedure, in order to ensure that the attention getters rotated sequentially through the full array before repeating any videos, even if the block changes before the end of the array is reached.

```

1  ***** Rotating Attn Getter Functionality *****
2
3  'defining the array of attn getter filenames'
4  AttnArray(0) = "RoosterAttentionGetter.mp4"
5  AttnArray(1) = "TrainAttentionGetter.mp4"
6  AttnArray(2) = "CowAttentionGetter.mp4"
7  AttnArray(3) = "FrogAttentionGetter.mp4"
8  AttnArray(4) = "CarAttn_Eq.mp4"
9  AttnArray(5) = "BirdAttentionGetter.mp4"
10 AttnArray(6) = "FrenchHornAttentionGetter.mp4"
11 AttnArray(7) = "RobotAttentionGetter.mp4"
12 AttnArray(8) = "TelephoneAttentionGetter.mp4"
13 AttnArray(9) = "DuckAttn_Eq2.mp4"
14
15 'debugging statement put in during development, displays array index of each image before randomization
16 'Debug.Print "ORIGINAL Array Order: "
17 'For j = 0 To UBound(AttnArray)
18 '  Debug.Print "attention getter assigned to array index " & j & " = " & AttnArray(j)
19 'Next j
20
21
22 'randomize the array
23 RandomizeArray AttnArray
24
25 'debugging statement put in during development, displays array index of each image after randomization
26 'DEBUG: Output the New array order
27 'Debug.Print ebLf & "NEW Array Order: "
28
29
30 'For j = 0 To UBound(AttnArray)
31 '  Debug.Print "background image assigned to array index " & j & " = " & AttnArray(j)
32 'Next j
33
34 'Initialize the variable that tracks which array index is being used to set the background image to index 0
35 currAttnIndex = 0
36

```

Session List

This list object establishes the two blocks used in the experiment; upright/inverted and upright/object, and their associated procedures. Each Block has a weight of 1, meaning that each one is only completed once, and they are randomly selected so that each participant could start with inverted or objects first.

Summary				
2 Samples (1 cycle x 2 samples/cycle)				
1 Cycle equals 2 samples				
Random Selection (No Repeat After Reset)				
ID	Weight	Nested	Procedure	BlockDescription
1	1		Block1	Upright/Inverted
2	1		Block2	Upright/Object

Block1

In this E-prime file, block 1 contains the upright/inverted condition, and block 2 has identical procedural components but with the upright/object condition instead.

Although these are listed as 1 and 2 within the E-Prime file, they may get presented in either order due to randomly selected counterbalancing.

List1 + UprightINV + InvertedList

This is Block 1's blocklist. It denotes 50 trials each of two conditions, Upright and Inverted. The list is pulling references from two other Nested lists. One of which is UprightINV and the other InvertedList. Each of those nested lists contain the names of all of the possible files that could be pulled for that condition type.

The screenshot shows three separate blocklist windows in E-Prime:

- List1:** Summary: 100 Samples (1 cycle x 100 samples/cycle). 1 Cycle equals 100 samples. Random Selection (No Repeat After Reset).

ID	Weight	Nested	Procedure	CellNumber
1	50	UprightINV	B1Trial	1
2	50	InvertedList	B1Trial	2
- UprightINV:** Summary: 36 Samples (1 cycle x 36 samples/cycle). 1 Cycle equals 36 samples. Random Selection (No Repeat After Reset).

ID	Weight	Nested	Procedure	image
1	1			F10_calm.png
2	1			F11_calm.png
3	1			F12_calm.png
4	1			F13_calm.png
5	1			F14_calm.png
6	1			F15_calm.png
7	1			F16_calm.png
8	1			F17_calm.png
9	1			F18_calm.png
10	1			F19_calm.png
11	1			F20_calm.png
12	1			F21_calm.png
13	1			F22_calm.png
14	1			F23_calm.png
15	1			F24_calm.png
16	1			F25_calm.png
17	1			F26_calm.png
18	1			F27_calm.png
19	1			F28_calm.png
20	1			F29_calm.png
21	1			F30_calm.png
22	1			F31_calm.png
23	1			F32_calm.png
24	1			F33_calm.png
25	1			F34_calm.png
26	1			F35_calm.png
27	1			F36_calm.png
28	1			
29	1			
30	1			
31	1			
32	1			
33	1			
34	1			
35	1			
36	1			
- InvertedList:** Summary: 36 Samples (1 cycle x 36 samples/cycle). 1 Cycle equals 36 samples. Random Selection (No Repeat After Reset).

ID	Weight	Nested	Procedure	image
1	1			F10_calm.png
2	1			F11_calm.png
3	1			F12_calm.png
4	1			F13_calm.png
5	1			F14_calm.png
6	1			F15_calm.png
7	1			F16_calm.png
8	1			F17_calm.png
9	1			F18_calm.png
10	1			F19_calm.png
11	1			F20_calm.png
12	1			F21_calm.png
13	1			F22_calm.png
14	1			F23_calm.png
15	1			F24_calm.png
16	1			F25_calm.png
17	1			F26_calm.png
18	1			F27_calm.png
19	1			F28_calm.png
20	1			F29_calm.png
21	1			F30_calm.png
22	1			F31_calm.png
23	1			F32_calm.png
24	1			F33_calm.png
25	1			F34_calm.png
26	1			F35_calm.png
27	1			F36_calm.png
28	1			
29	1			
30	1			
31	1			
32	1			
33	1			
34	1			
35	1			
36	1			

A red arrow points from the "Nested" column in the List1 table to the "image" column in the UprightINV table. Another red arrow points from the "Nested" column in the List1 table to the "image" column in the InvertedList table.

B1Trial

The procedure that contains all the components of a Face Task Trial.

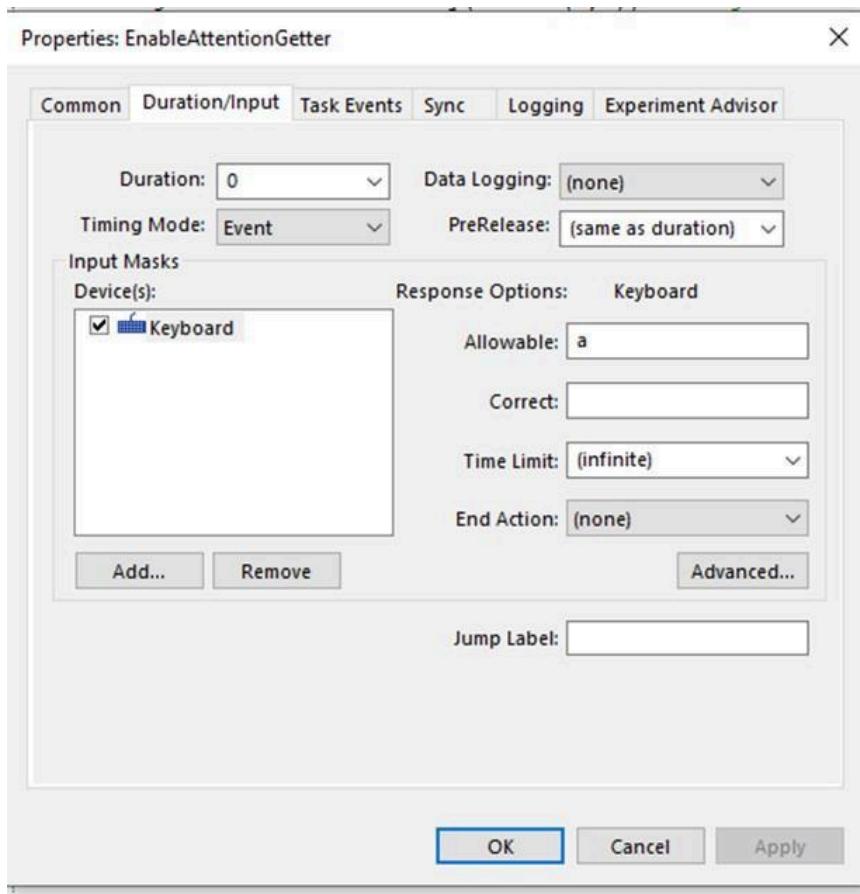
Rotations

This inline script manages the rotations for the background, attention getters, and ITI delay. It updates the background image variable to equal the current image index within the background image array. The script tracks the time elapsed since the background last rotated. Once the time passes 120 seconds, the background cycles to the next image in the array. Lines 13-16 in the script track whether the final image in the array has been displayed, and returns to the beginning of the array to re-display each background in the same order. The script also assigns the current attention getter filename in the array index to the variable that holds the filename when an attention getter is cued. It also generates a random number between 500 and 700 to be the duration of the ITI delay in ms. Each block of the experiment contains a copy of this script.

```
1 '''''' mechanism for rotating background and rotating attention getter''''''
2
3 'UPDATE THE BACKGROUND IMAGE EVERY 120 S
4
5 'debugging put in during development, shows what image is stored as current background image
6 Debug.Print ebLf & "Currently assigning background images from array index " & currBackImageIndex
7
8 backgroundvars = BackArray(currBackImageIndex) ' set backgroundvars equal to the background image name in the current BackArray index (which is 0 at the star
9 'debugging check put in during development
10 Debug.Print "Current background image is " & backgroundvars
11
12 If (Clock.Read - StartTime) >= 120000 Then 'if the current time in the experiment minus the timestamp of when the stimulus appeared is greater than 120s'
13   If currBackImageIndex = UBound(BackArray) Then 'if you reach the end of the array, restart from the beginning
14     currBackImageIndex = 0 'reset the current index counter back to 0
15   Else
16     'not yet at the end of the array, so increment the array index
17     currBackImageIndex = currBackImageIndex + 1
18   End If
19   StartTime = Clock.Read 'reset the timer reference'
20 End If
21
22 c.SetAttrib "BackgroundImage", backgroundvars 'make the stored variable, actually the one to appear'
23
24 'MECHANISM FOR ATTENTION GETTER
25
26 'debugging statement put in during development
27 Debug.Print ebLf & "Currently assigning attention getter from array index " & currAttnIndex
28
29 attentiongettervars = AttnArray(currAttnIndex) 'set attentiongettervars to equal the attention getter filename in the current AttnArray index (which is 0 at
30
31 'debugging statement put in during development
32 Debug.Print "Current attention getter is " & attentiongettervars
33
34 c.SetAttrib "AttnGet", attentiongettervars 'make the stored variable, actually the one to appear'
35
36 '''''' setting variables for random ITI ''
37 waitlengthvars= Random(500,700) ' Randomly selecting a value for the ITI '
38 c.SetAttrib "waitlength", waitlengthvars 'Setting the randomly selected value to the attribute to actually be used'
```

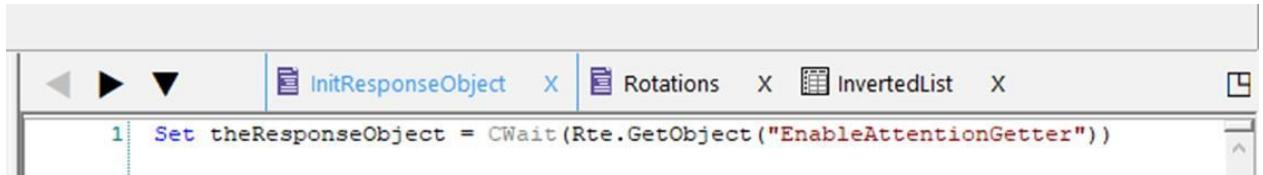
EnableAttentionGetter

This is a wait object that is set to have a duration of 0. It essentially works as a placeholder object that can be used to collect keyboard responses, in this case, 'a' to activate the attention getters.



InitresponseObject

This inline script assigns the EnableAttentionGetter wait object from above to the variable “theresponseObject” that was created in the user script.

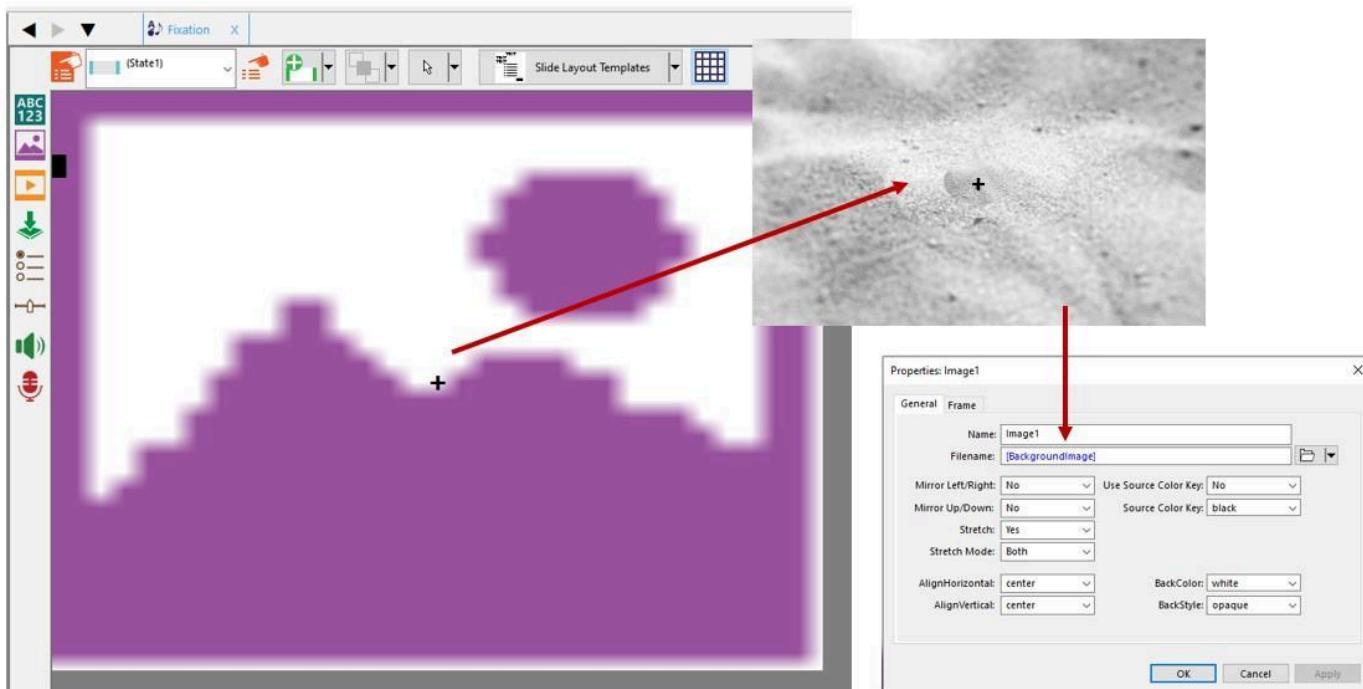


NSTB

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

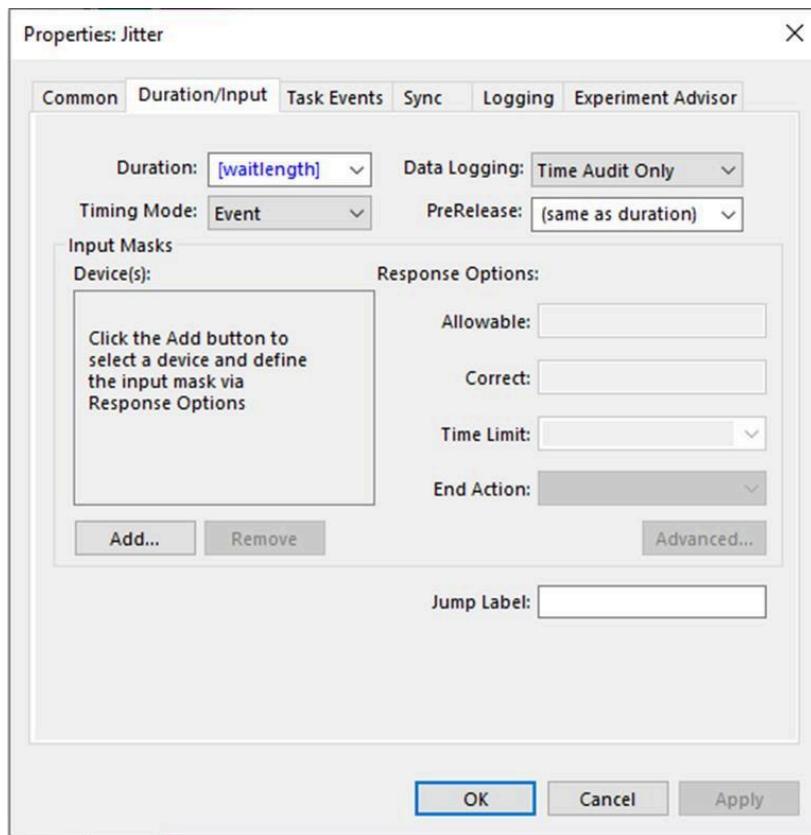
Fixation

This object provides a stimulus-free screen that is presented before the actual stimulus is presented. The background on this object will match the background of the stim object and will change depending on what is randomly selected and loaded every 120s. This slide will stay on the screen indefinitely, and the experimenter presses the spacebar to terminate this slide and progress to the stimulus. If the infant is not looking at the screen, the experimenter can leave this slide up as long as needed, and return to the progression when they are ready. However, there is a delay (random duration between 500-700ms) that is instituted by the jitter object below, before the stimulus actually gets presented. That means that if the presenter is consistently pressing the spacebar, the stimulus only appears on the screen 500-700 ms after their spacebar press.



Jitter

This wait object is responsible for executing the jitter, or random variation in presentation time, that is part of the Face task design. This object temporarily stops the experiment from continuing after the presentation of the fixation cross. The duration of this wait object is a random duration between 500-700 ms and is set by the waitlength attribute designated in the Rotations inline script.



AttentionEnactor3

Right before the stimulus is presented, this checks if theResponseObject (which is assigned to be the EnableAttentionGetter wait object) has registered any presses of the 'a' key. If it has, it will jump over the stimulus presentation sequence and take the experiment to the AttentionGetter flag, which marks the beginning of the attention getter sequence. When the 'a' key is pressed, this script progresses to the index of the attention getter array that has not been displayed yet. It progresses through each index sequentially, and then returns to the start of the array. If no 'a' key is registered the

experiment will progress linearly and continue to the stimulus presentation sequence.

```
1 'If the counts Do Not match, there are responses To process.
2 If nResponseCount <> theresponseObject.InputMasks.Responses.Count Then
3
4     'Increment response count.
5     nResponseCount = nResponseCount + 1
6
7     'Access the current response information so that it can be used
8     'later in the script for logging and determining which action to take.
9     Set theresponseData = theresponseObject.InputMasks.Responses(nResponseCount)
10    If Not theresponseData Is Nothing Then
11
12        'Log the response to the context
13        c.SetAttrib theresponseObject.Name & ".RESP_" & nResponseCount, theresponseData.RESP
14        c.SetAttrib theresponseObject.Name & ".RT_" & nResponseCount, theresponseData.RT
15
16        'If "a" Is pressed run the Unreferenced Pause Slide
17        If theresponseData.RESP = "a" Then
18            If currAttnIndex = UBound(AttnArray) Then 'if you reach the end of the array, restart from the beginning
19                currAttnIndex = 0 'will select the image in array index 0 at the start of the next trial
20            Else
21                currAttnIndex = currAttnIndex + 1 'not yet at the end of the array, so increment the array index
22            End If
23
24            GoTo AttentionGetter
25
26        End If
27
28        ' Must set input properties manually if running from Unreferenced E-Objects
29        'UnrefSlide.InputMasks.Add Keyboard.CreateInputMask("(SPACE)", "", CLng(UnrefSlide.Duration), CLng("1"), ebEndResponseActionTerminate, CLogical("Yes"))
30        'UnrefSlide.Run
31    End If
32
33 End If
34
```

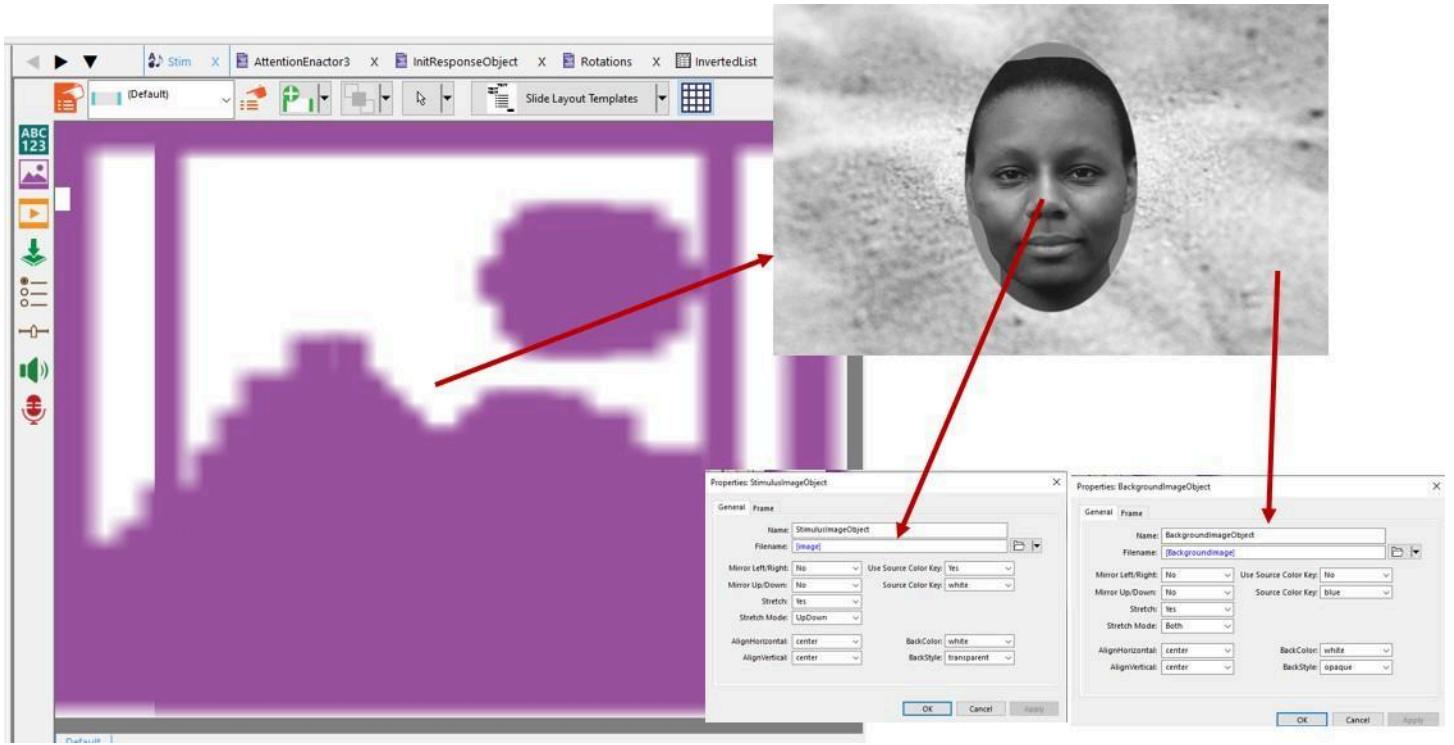
FlagStim

This object has no settings, it is simply a label that is used as a reference point within the procedure.



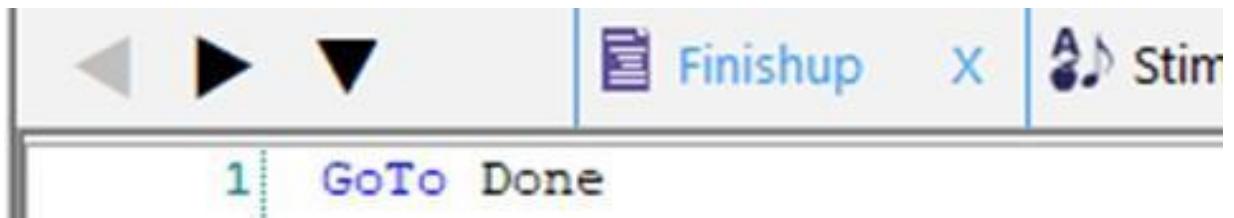
Stim

This slide object presents the experimental stimuli for the task. It consists of two image display objects layered over each other. One image display object, named StimulusImageObject, presents the faces or objects, and overlays it on top of the BackgroundImageObject that displays the background image (which rotates every 120s). This slide object has a duration of 500 ms.



Finishup

If the attention getter HAS NOT been activated during the duration of the trial, the procedure will continue to the 'finish' up flag which jumps the experiment to the 'Done' flag, which effectively skips the entire portion of the protocol that would play an attention getter.



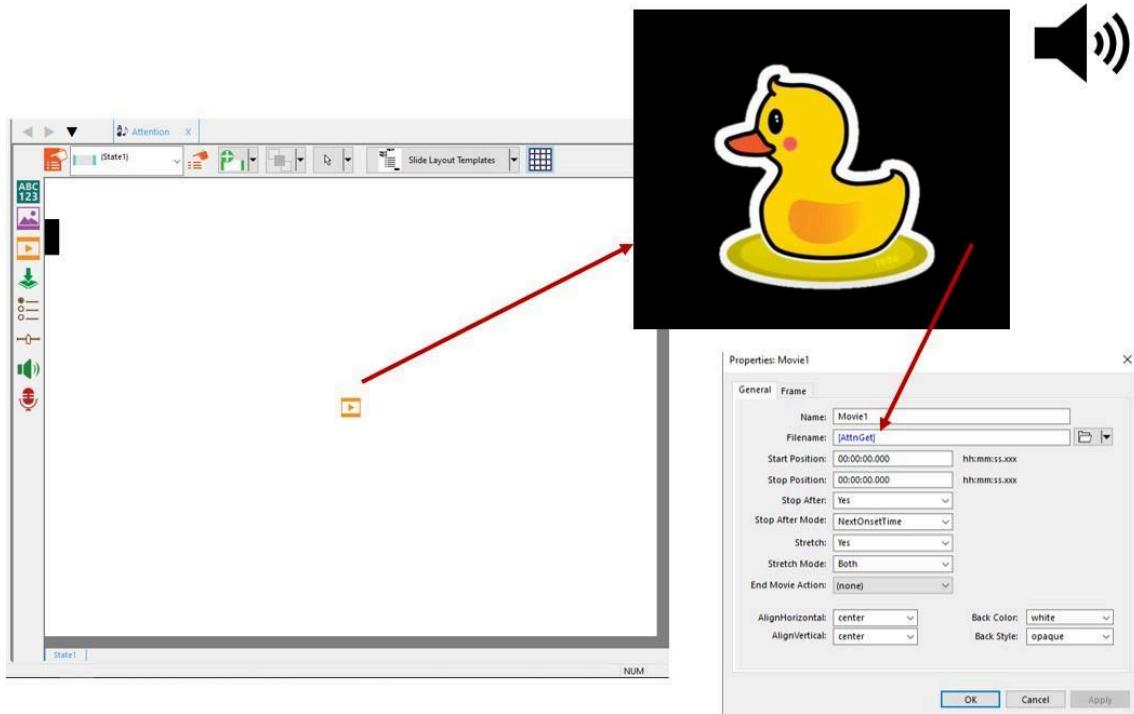
AttentionGetter

This object has no settings, it is simply a label that is used as a reference point within the procedure.



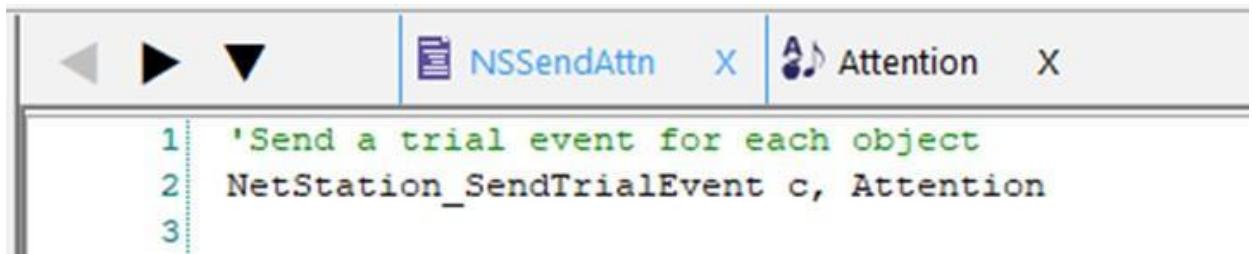
Attention

This is a slide object that contains a movie object. This is the E-object that actually presents the attention getter. Instead of having a singular file referenced in the movie object's properties, it references the 'AttnGet' attribute which holds different file names based on which attention getter has been randomly assigned to the attribute through the AttnRotation and Rotations inline scripts.



NSendAttn

Separate Net Station send events inline script for the attention slide object. See Part Two: E-Prime General Features: NSSendEvent.



The screenshot shows the E-Prime software interface with the following details:

- Top menu bar: Back, Forward, Stop, NSSendAttn (selected), X, Attention (selected), X.
- Script editor window:

```
1 'Send a trial event for each object
2 NetStation_SendTrialEvent c, Attention
3
```

AttentionEnactor2

Same contents as AttentionEnactor3 above. It allows the attention getter to be replayed if needed.

Fixation2

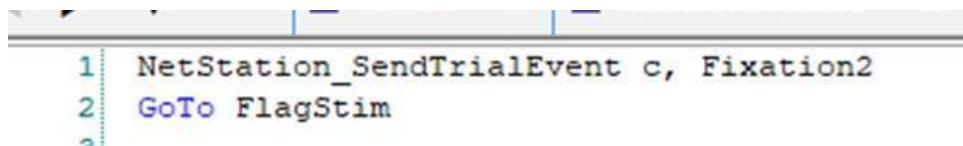
Same contents as Fixation above.

AttentionEnactor5

Same contents as AttentionEnactor3 above. It allows the attention getter to be replayed if needed.

InLine2

If the attention getter was activated during the duration of the trial, after it is finished playing, this inline sends it back to the FlagStim flag and will present the sequence for stimulus presentation that it had skipped in order to present the attention getter.



The screenshot shows the E-Prime software interface with the following details:

- Top menu bar: Back, Forward, Stop, InLine2 (selected), X.
- Script editor window:

```
1 NetStation_SendTrialEvent c, Fixation2
2 GoTo FlagStim
3
```

Done

This object has no settings, it is simply a label that is used as a reference point within the procedure.



NSSendEvents

This inline's SendTRSP has an additional argument to include the "KeyList". A description of the KeyList is included further down. *See Part Two: E-Prime General Features: NSSendEvent*

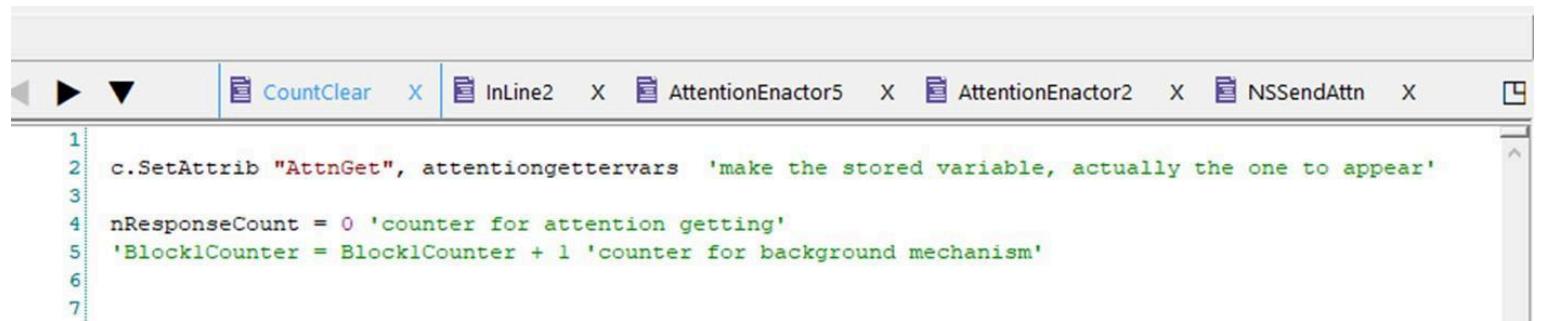
```
1  'Send a trial event for each object
2  NetStation_SendTrialEvent c, Stim
3  NetStation_SendTrialEvent c, Jitter
4  NetStation_SendTrialEvent c, Fixation
5
6
7
8
9  'Send the trial a specific event
10 NetStation_SendTRSPEvent c, Stim, KeyList
11
12
```

TE

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

Count Clear

This inline script just resets the counter that keeps track of keyboard responses for the Attention Getter.



The screenshot shows the E-Prime software interface. At the top, there's a toolbar with icons for play, stop, and other controls. Below the toolbar, a list of objects is displayed: CountClear X, InLine2 X, AttentionEnactor5 X, AttentionEnactor2 X, NSSendAttn X. The main window contains a script editor with the following code:

```
1 c.SetAttrib "AttnGet", attentiongettervars 'make the stored variable, actually the one to appear'
2 nResponseCount = 0 'counter for attention getting'
3 'Block1Counter = Block1Counter + 1 'counter for background mechanism'
4
5
6
7
```

BlockComplete

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

Block2

Exactly the same as block 1, the objects have numbers appended to the end to make sure the names of E-objects are not repeated across the blocks. For example, the slide object named "Stim" in block one will be named "Stim1" in block 2.

Goodbye

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

NSStopRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

NSUnit

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

CellList

Includes the experimental conditions as well as a log for the background and for the LORIS ID and visit number.

ID	Weight	Nested	Procedure	CellNumber	CellLabel
1	1			1	uprightINV
2	1			2	inverted
3	1			3	objects
4	1			4	uprightOBJ
5	1			200	Background
6	1			100	[LORISID]
7	1			2	[Visit]

KeyList

Keylist is similar to a cell list, as it is an unreferenced object that records additional information you may want logged about the parameters surrounding a particular trial. What makes it different is that it is recording additional information that is not the experimental conditions (i.e.: upright, inverted, object) but may be needed/helpful later. Here, it is used to log which background image was used behind the stimuli, and which exact file was used in that trial. This may be helpful later on if people wanted to conduct analysis specifically relating to certain individual faces, and simply knowing whether the image was “upright” or not, isn’t sufficient.

ID	Weight	Nested	Procedure	KeyName	KeyID	KeyDataType
1	1			image	imag	TEXT
2	1			BackgroundImage	bgim	TEXT

Mismatch Negativity (MMN)

User script

Establishes some variables that help create the presentation dimensions for some of the screens shown later on. Also creates some variables of the pause function, as well as the visit number and LORID.

```
1| Dim gCnvs As Canvas
2| Dim gCnvsCenterX As Integer
3| Dim gCnvsCenterY As Integer
4| Dim gStimX As Integer
5| Dim gStimY As Integer
6| Dim gStimWidth As Integer
7| Dim gStimHeight As Integer
8|
9| Dim displaytext(10) As String
10| 'Dim blocknum(9) as String
11| Dim blockloop As Integer
12| Dim bb,cc As String
13|
14| Dim blocktype(3), orderfile(), TwoSound(), cfgtxt() As String
15| Dim blocknum, ordernum, cfgnum As Integer
16| Dim soundno1, soundno2, soundno As Integer
17| Dim soundfile As String
18| Dim cfgpara(4) As String
19|
20| Dim ITIDuration As Integer
21|
22| Dim blknum As Integer 'this for cell number
23| Dim PracDone, REEGDone As Boolean
24|
25| Dim pb As Integer
26|
27| Dim ObsStd, ObsDev, OBS As Integer
28|
29| Dim prconly As Integer
30|
31| 'Dim counterbal As String 'creating a variable to call a particular block based on counterbalance selected'
32|
33| 'Dim counterbal As String 'creating a variable to call a particular block based on counterbalance selected'
34|
35| '' PAUSE FUNCTIONALITY''''''
36|
37| Dim nResponseCount As Integer 'Declare variables for holding the current response count.
38| Dim theResponseData As ResponseData 'Declare variables for holding the current response data.
39| Dim theResponseObject As Wait 'Global variable for trial pause mechanism
40|
41| '' To allow LORIS ID and Visit number to be relayed to NetStation via Cell List ''
42| Dim LORID As String
43| Dim Visit As String 'to label NetStation and EDat files with the visit number
```

SessionProc

All of the task is contained on this procedure.

Init

This inline further initializes display settings. It also has the portion of code that sends the LORID and visit number to the CellList.

```
1| ' This code initializes the display for the experimenter
2| Dim displaytext(10) As String
3|
4| Set gCnvs = Display.Canvas
5| gCnvsCenterX = Display.XRes / 2
6| gCnvsCenterY = Display.YRes / 2
7| gStimWidth = .06 * Display.XRes
8| gStimHeight = .10 * Display.YRes
9| c.SetAttrib "StimWidth", CStr(gStimWidth)
10| c.SetAttrib "StimHeight", CStr(gStimHeight)
11|
12| '' To allow LORIS ID to be relayed to NetStation via Cell List ''
13| LORID = c.GetAttrib("DCCID") 'assigning that variable the based on the LORIS ID entered in startup info'
14| CellList.SetAttrib 5, "CellLabel", LORID
15|
16| 'Sending Visit number to NetStation and labeling EDat files with it
17| Visit = c.GetAttrib("Visit") 'assigning that variable based on the visit number entered in startup info'
18| CellList.SetAttrib 6, "CellLabel", Visit 'logging the Visit variable in the CellList'
19|
```

Initialize

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

Blocklist

This list has the two possible versions of MMN that the participant could hear. The 'Ba' procedure has the /ba/ sound as the standard sound and the 'Da' procedure has the /da/ as the standard sound. E-Prime randomly counterbalances between these two options via random selection.

The screenshot shows the E-Prime software interface with the 'Blocklist' tab selected. The window displays a table of experimental procedures:

ID	Weight	Nested	Procedure	PracticeMode
1	1		Ba	No
2	1		Da	No

Ba

Procedure that contains the version of MMN that has 'ba' as the standard sound.

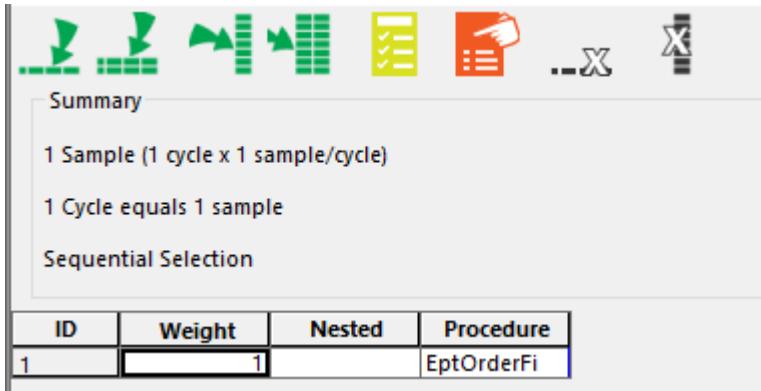
LabelG

This object has no settings, it is simply a label that is used as a reference point within the procedure.



OrderList

There used to be a practice block with a practice procedure in this task, but it was removed to only contain the one experimental procedure as seen below.



EptOrderFile

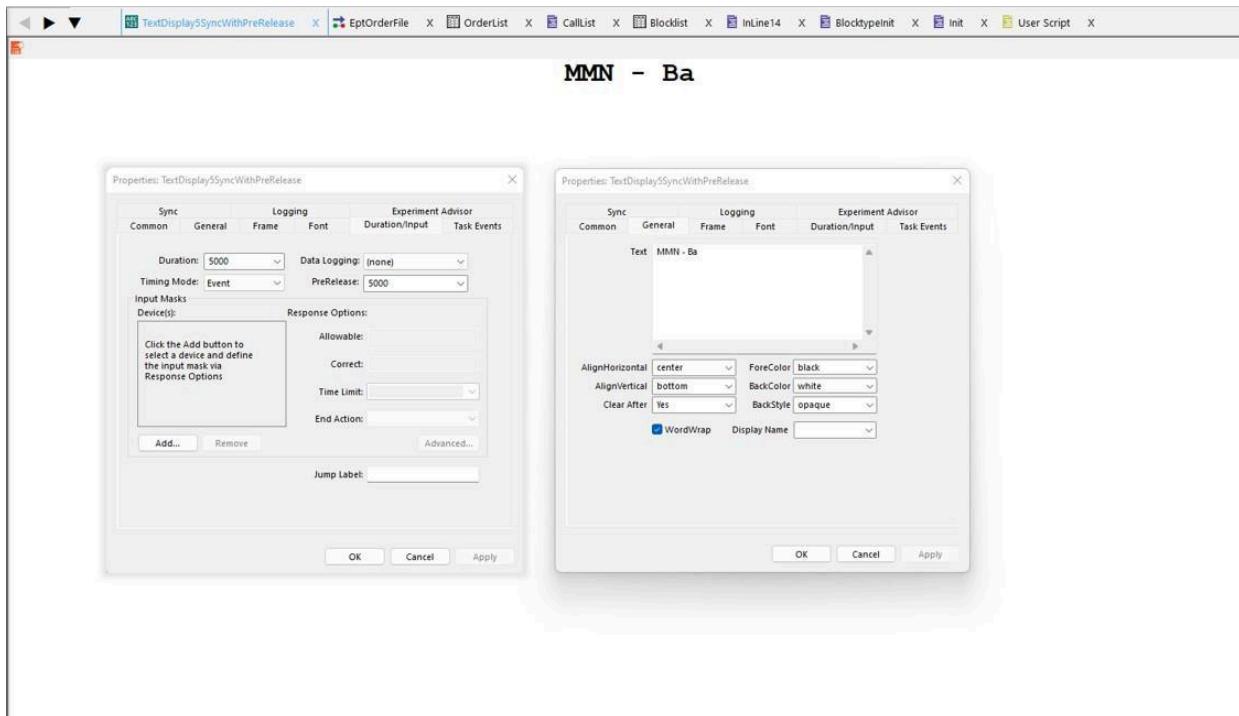
The procedure that contains all the components of the MMN trial structure.

NSStartRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

TextDisplay5SyncWithPreRelease

This screen displays a white background. The label of the task, in this case, "MMN - Ba", is also displayed as well.



TrialList

This list contains all 667 trials of the task. This list is set to “sequential selection” because the order of the trials are in a known set configuration, and should not be scrambled. Sound type is 1 = standard and 2 = deviant. The CellNumber refers to the conditions needed for analyzing the data (standard, deviant, pre-deviant) see *CellList* below for more details.

ID	Weight	Nested	Procedure	SoundType	Stimulus	TrialNum	CellNumber
1	1		TrialProc	1	ba.wav	1	1
2	1		TrialProc	1	ba.wav	2	1
3	1		TrialProc	1	ba.wav	3	1
4	1		TrialProc	1	ba.wav	4	1
5	1		TrialProc	1	ba.wav	5	1
6	1		TrialProc	1	ba.wav	6	1
7	1		TrialProc	1	ba.wav	7	1
8	1		TrialProc	1	ba.wav	8	1
9	1		TrialProc	1	ba.wav	9	1
10	1		TrialProc	1	ba.wav	10	2
11	1		TrialProc	2	da1.wav	11	3
12	1		TrialProc	1	ba.wav	12	1
13	1		TrialProc	1	ba.wav	13	1
14	1		TrialProc	1	ba.wav	14	1
15	1		TrialProc	1	ba.wav	15	2
16	1		TrialProc	2	da1.wav	16	3
17	1		TrialProc	1	ba.wav	17	1
18	1		TrialProc	1	ba.wav	18	1
19	1		TrialProc	1	ba.wav	19	1
20	1		TrialProc	1	ba.wav	20	1

TrialProc

The procedure that contains all the components of the MMN.

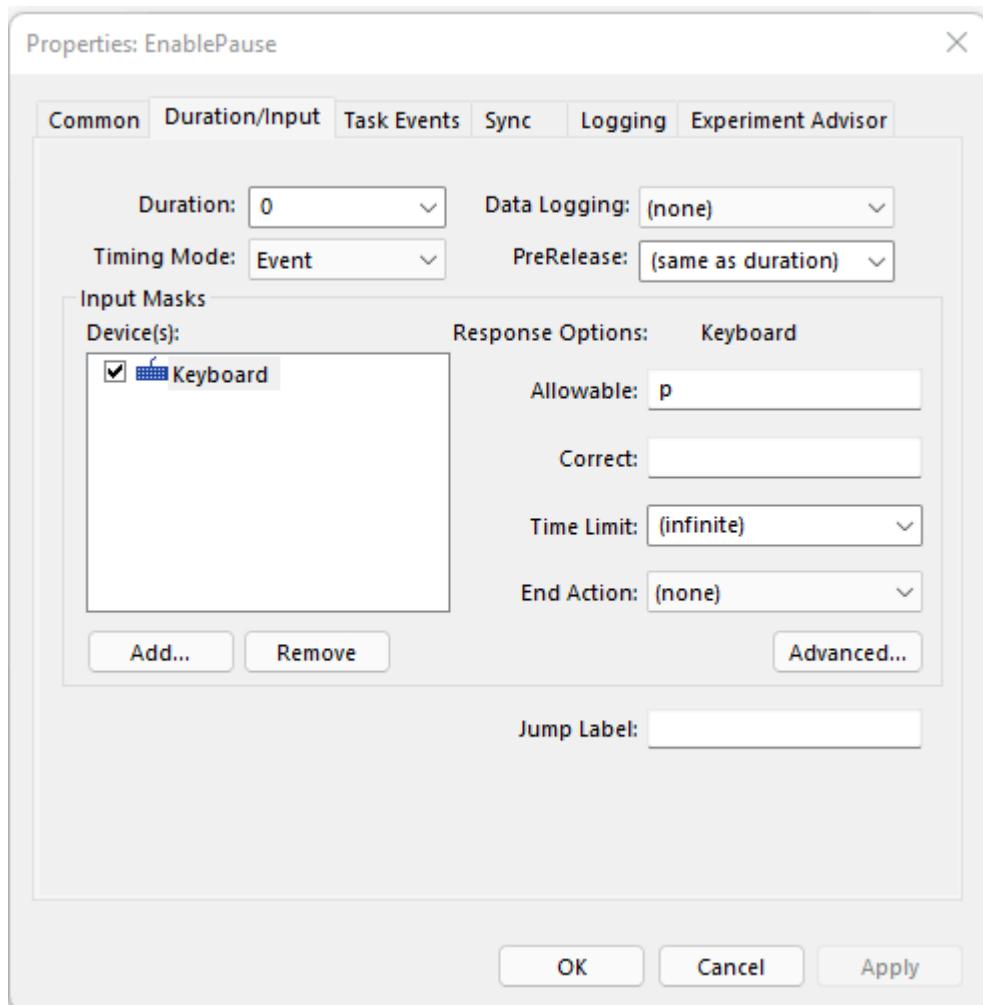
CountClear

Restarts the counter that is used in the pause function.

```
nResponseCount = 0 'counter for pause'
```

EnablePause

This is a wait object that is set to have a duration of 0. It essentially works as a placeholder object that can be used to collect keyboard responses, in this case, 'p' to activate the pause function.



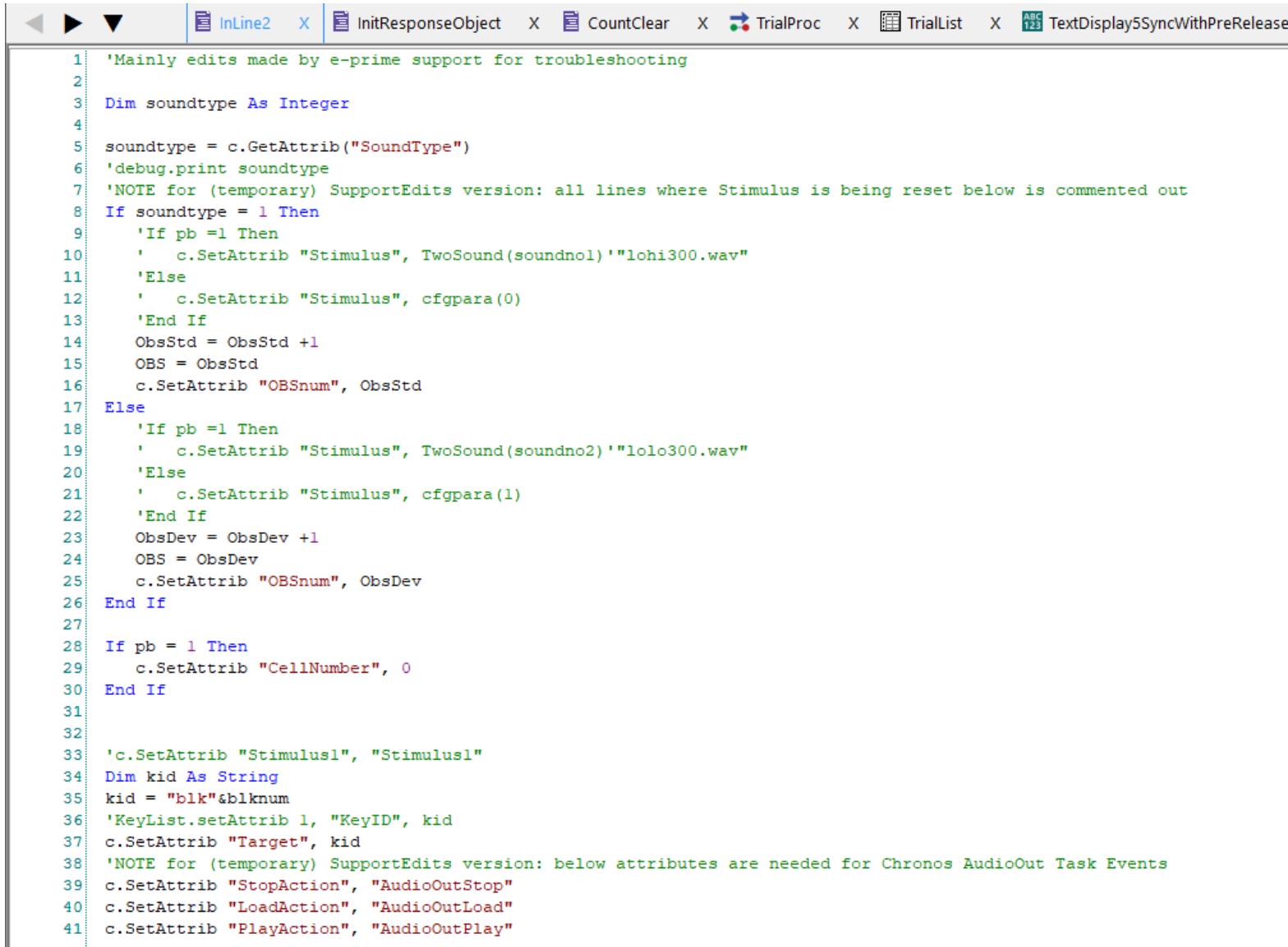
InitresponseObject

This inline script assigns the "EnablePause" wait object from above to the variable 'theresponseObject' that was created in the user script.

```
Set theresponseObject = CWait(Rte.GetObject("EnablePause"))
```

InLine2

This script was inserted by the folks at PST to aid the troubleshooting process.



The screenshot shows a software interface with multiple tabs open at the top. The 'InLine2' tab is the active one, indicated by a blue border. Other tabs include 'InitResponseObject', 'CountClear', 'TrialProc', 'TrialList', and 'TextDisplay5SyncWithPreRelease'. The main window displays a block of VBScript code. The code is primarily used for troubleshooting, specifically for audio playback. It includes logic to set stimulus files based on a variable 'soundtype' (which is read from a configuration attribute 'SoundType'). It also handles cell number assignment ('CellNumber') and initializes Chronos AudioOut Task Events ('KeyList.setAttrib'). The code is heavily commented with explanatory notes and temporary support edits.

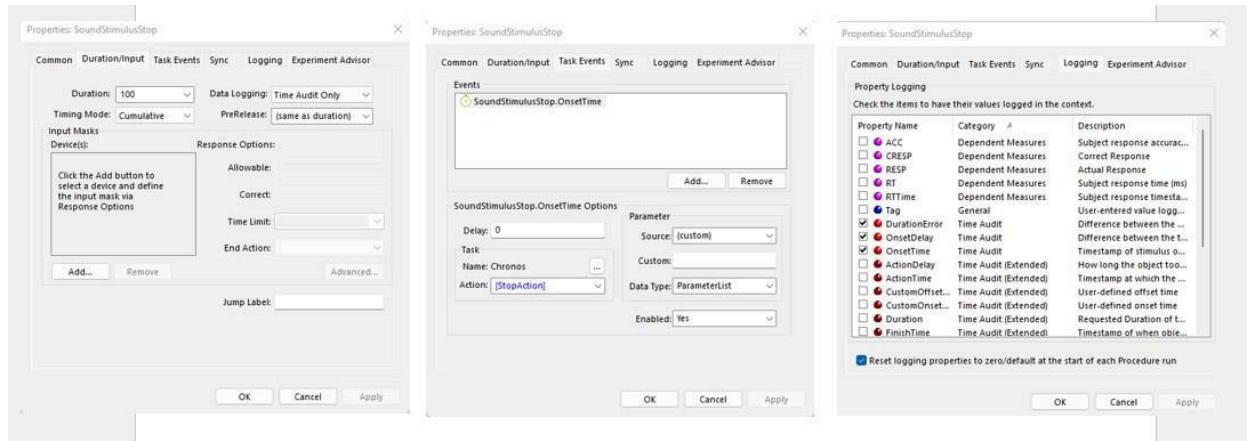
```
1 'Mainly edits made by e-prime support for troubleshooting
2
3 Dim soundtype As Integer
4
5 soundtype = c.GetAttrib("SoundType")
6 'debug.print soundtype
7 'NOTE for (temporary) SupportEdits version: all lines where Stimulus is being reset below is commented out
8 If soundtype = 1 Then
9   'If pb =1 Then
10    '  c.SetAttrib "Stimulus", TwoSound(soundno1)"lohi300.wav"
11   'Else
12    '  c.SetAttrib "Stimulus", cfgpara(0)
13   'End If
14   ObsStd = ObsStd +1
15   OBS = ObsStd
16   c.SetAttrib "OBSnum", ObsStd
17 Else
18   'If pb =1 Then
19    '  c.SetAttrib "Stimulus", TwoSound(soundno2)"lolo300.wav"
20   'Else
21    '  c.SetAttrib "Stimulus", cfgpara(1)
22   'End If
23   ObsDev = ObsDev +1
24   OBS = ObsDev
25   c.SetAttrib "OBSnum", ObsDev
26 End If
27
28 If pb = 1 Then
29   c.SetAttrib "CellNumber", 0
30 End If
31
32
33 'c.SetAttrib "Stimulus1", "Stimulus1"
34 Dim kid As String
35 kid = "blk"&blknum
36 'KeyList.setAttrib 1, "KeyID", kid
37 c.SetAttrib "Target", kid
38 'NOTE for (temporary) SupportEdits version: below attributes are needed for Chronos AudioOut Task Events
39 c.SetAttrib "StopAction", "AudioOutStop"
40 c.SetAttrib "LoadAction", "AudioOutLoad"
41 c.SetAttrib "PlayAction", "AudioOutPlay"
```

NSTrialbegin

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

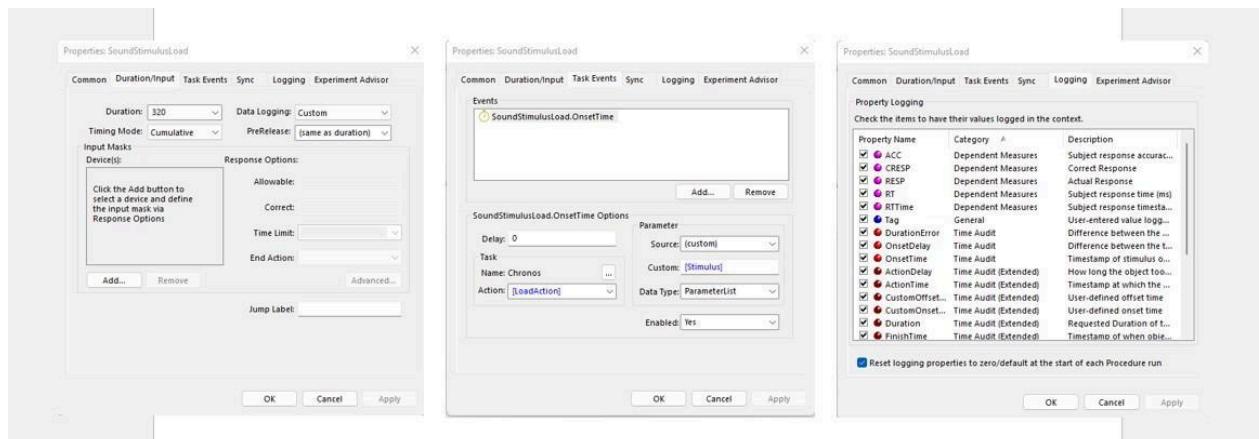
SoundStimulusStop

Chronos command to stop any sound that may be playing.



SoundStimulusLoad

Chronos command to load the .wav file from the "Stimulus" column from the trial list. Loading the sound .wav file here helps with precision of timing. In the V03 version of the task this has a duration of 320ms. In the V04-V06 version of the task the duration was lowered to 200ms, as part of the overall lowering of the ISI from 820ms to 600ms.



ClearChronosHistory

This portion was suggested by E-Prime, it prepares Chronos for the next sound to be played.

```

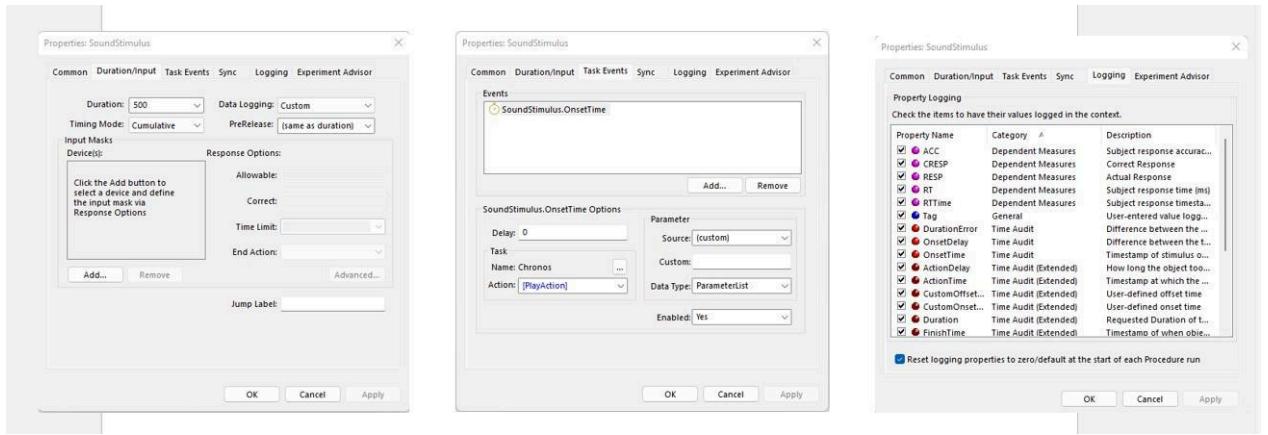
    <--> [ ] ClearChronosHistory X [ ] InLine2 X [ ] InitResponseObject X [ ] CountClear X [ ] TrialProc X

1  Chronos.History.RemoveAll
2
3  ' The line above was recommended by e-prime support,
4  ' to clear the chronos history before playing the new sound
5
6  ' There are three objects in the trial that are critical to presenting
7  ' the sounds:
8  ' 1. The SoundStimulusStop object has a chronos task event which
9  '     stops any sound from the previous trial from playing
10 ' 2. The SoundStimulusLoad object has a chronos task event which pre-loads
11 '     the sound for the current trial. The sound type is assigned by the TrialList.
12 '     It's essential to load the sound in advance, to prevent delays.
13 ' 3. The SoundStimulus object has a chronos task event which actually plays the sound
14

```

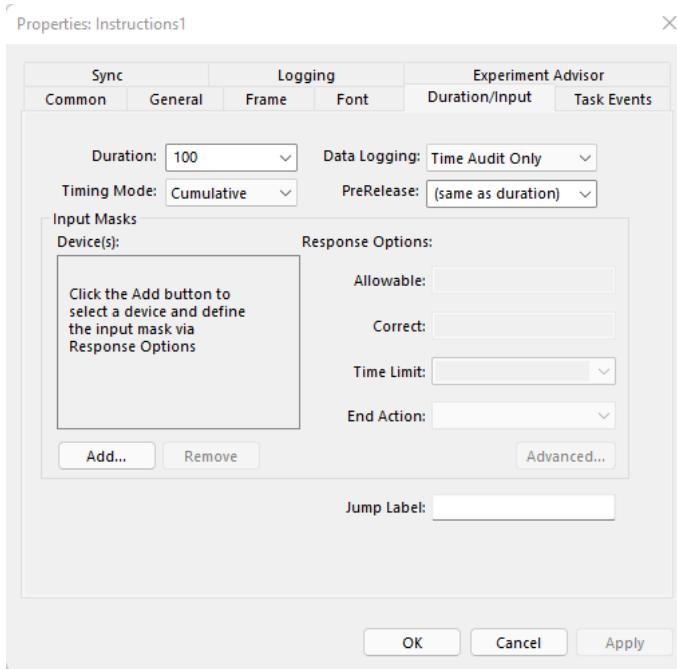
SoundStimulus

Chronos command to play the sound from the .wav.



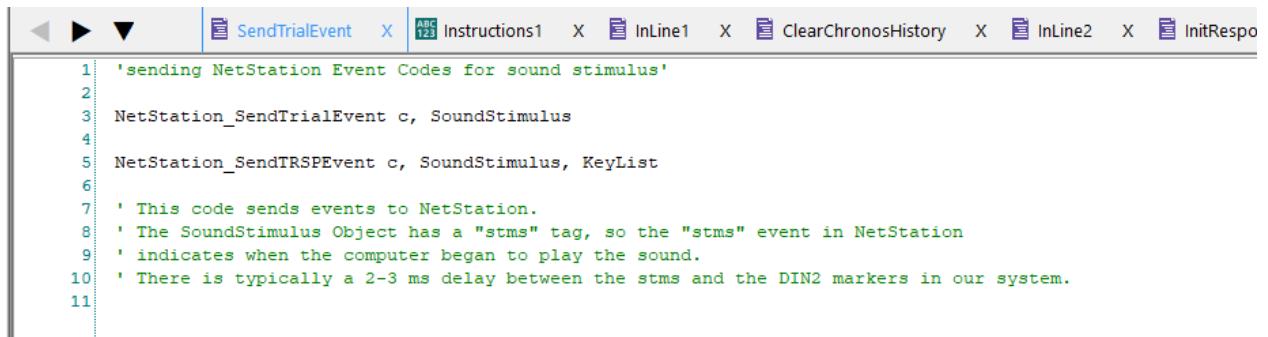
Instructions1

Blank text object with a duration of 100ms. It is simply an E-object that helps ensure the total trial duration adds up to the desired value. This was deleted in the V04-V06 version, in order to lower the ISI from 820ms to 600ms.



SendTrialEvent

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

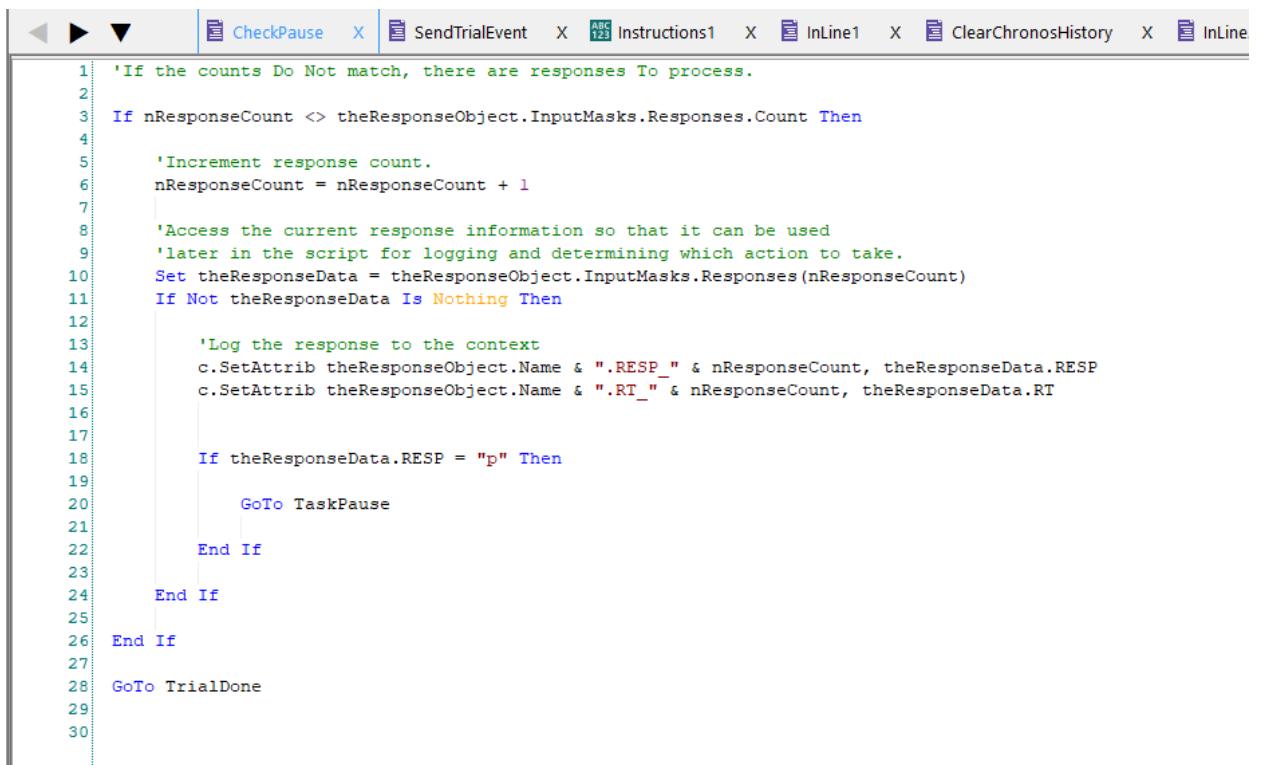


The screenshot shows the E-Prime software interface with multiple tabs at the top: SendTrialEvent (active), Instructions1, InLine1, ClearChronosHistory, InLine2, and InitRespo. The SendTrialEvent tab contains the following VBS code:

```
1 'sending NetStation Event Codes for sound stimulus'
2
3 NetStation_SendTrialEvent c, SoundStimulus
4
5 NetStation_SendTRSPEvent c, SoundStimulus, KeyList
6
7 ' This code sends events to NetStation.
8 ' The SoundStimulus Object has a "stms" tag, so the "stms" event in NetStation
9 ' indicates when the computer began to play the sound.
10 ' There is typically a 2-3 ms delay between the stms and the DIN2 markers in our system.
11
```

CheckPause

This inline script checks to see if theresponseObject (which is assigned to be the EnablePause wait object) has registered any presses of the 'p' key. If it has, it will jump to the TaskPause flag, which precedes the pause screen. If no 'p' key is registered the experiment will continue to the TrialDone flag, and finish the trial.

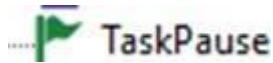


The screenshot shows the E-Prime software interface with multiple tabs at the top: CheckPause (active), SendTrialEvent, Instructions1, InLine1, ClearChronosHistory, and InLine. The CheckPause tab contains the following VBS code:

```
1 'If the counts Do Not match, there are responses To process.
2
3 If nResponseCount <> theresponseObject.InputMasks.Responses.Count Then
4
5     'Increment response count.
6     nResponseCount = nResponseCount + 1
7
8     'Access the current response information so that it can be used
9     'later in the script for logging and determining which action to take.
10    Set theresponseData = theresponseObject.InputMasks.Responses(nResponseCount)
11    If Not theresponseData Is Nothing Then
12
13        'Log the response to the context
14        c.SetAttrib theresponseObject.Name & ".RESP_" & nResponseCount, theresponseData.RESP
15        c.SetAttrib theresponseObject.Name & ".RT_" & nResponseCount, theresponseData.RT
16
17        If theresponseData.RESP = "p" Then
18
19            GoTo TaskPause
20
21        End If
22
23    End If
24
25 End If
26
27 GoTo TrialDone
28
29
30
```

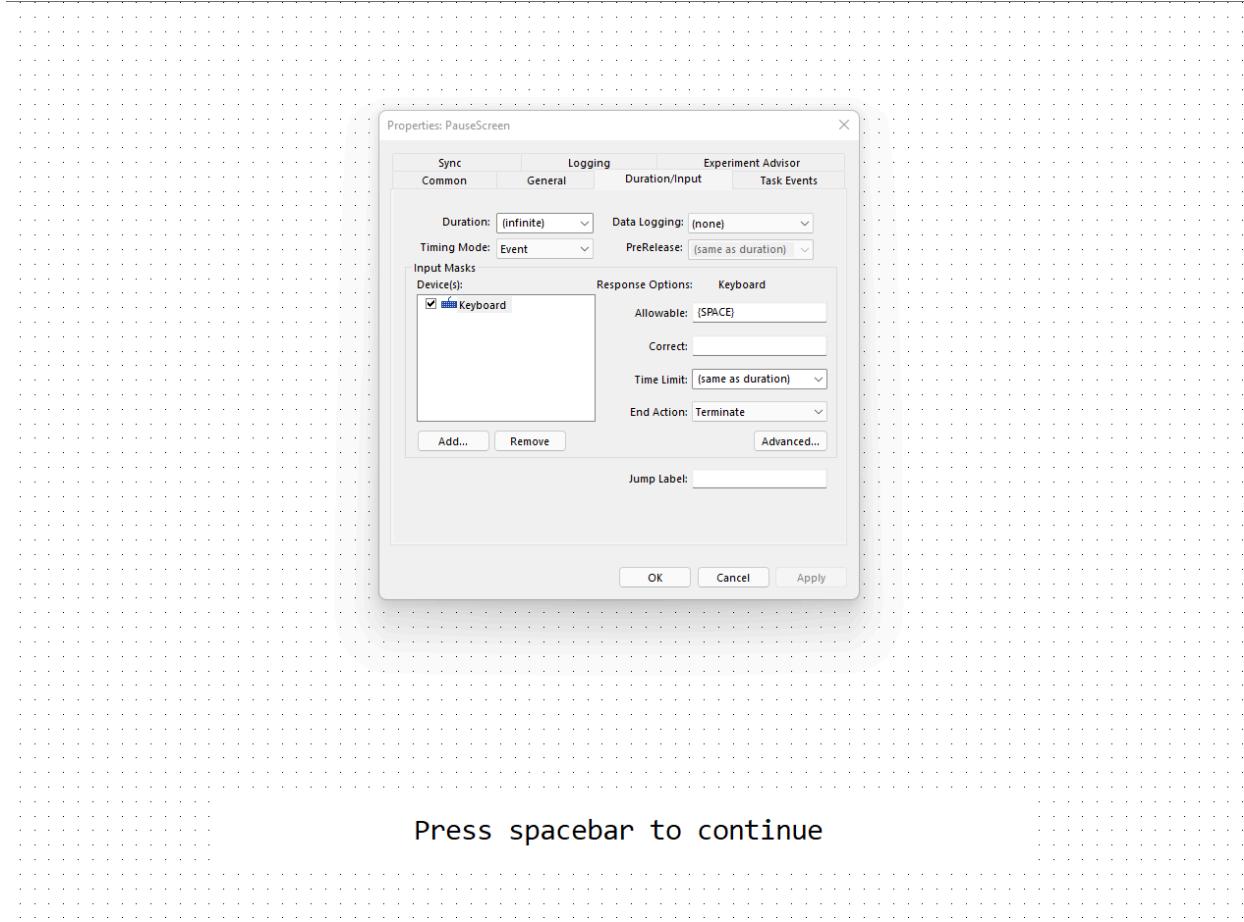
TaskPause

Label used to mark the PauseScreen.



PauseScreen

White screen with the instructions “Press Spacebar to Continue”. This is the screen that will sit on the monitor when the experimenter has prompted the pause.



SendPauseEvent

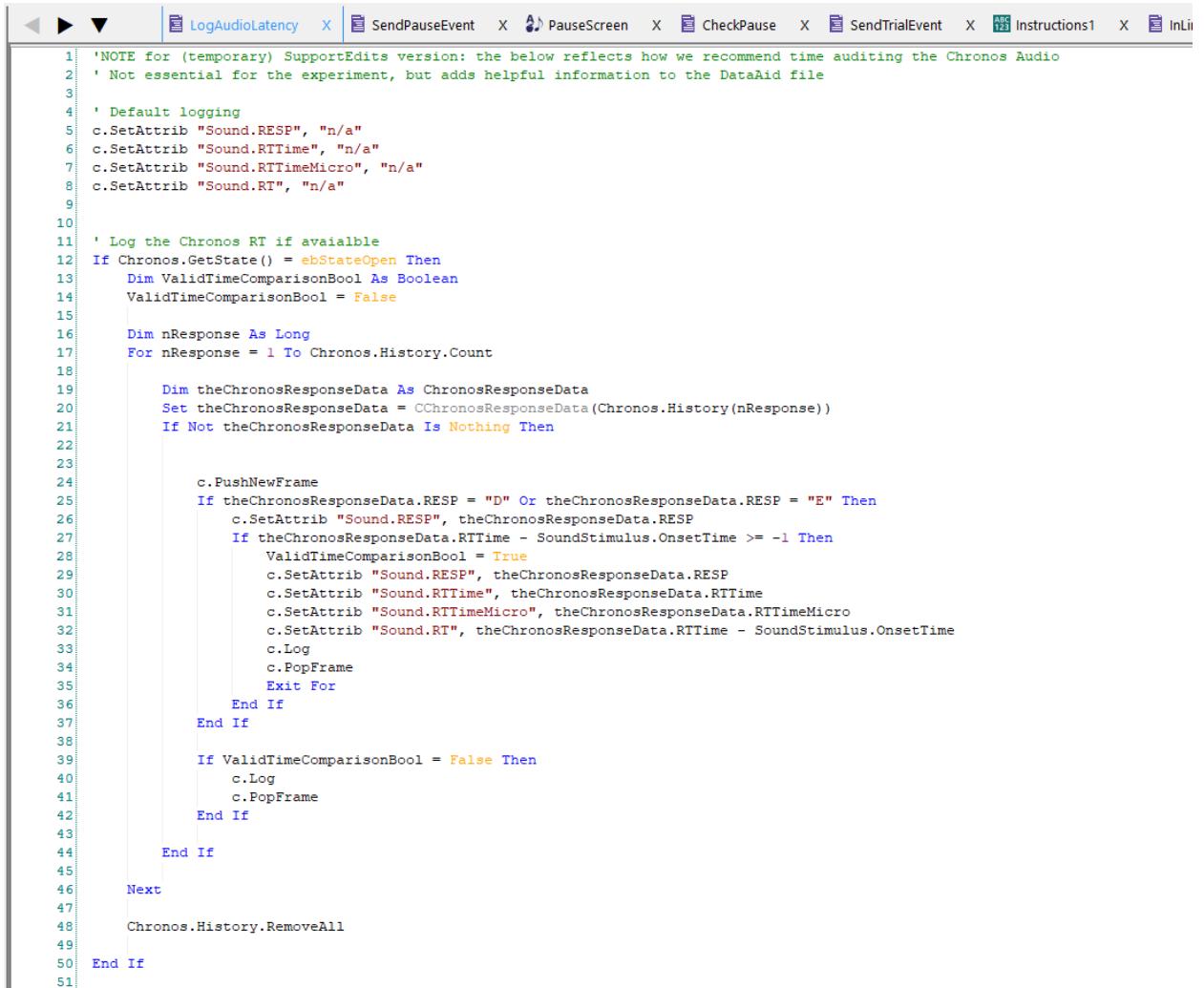
Sends the event flag for PauseScreen. See Part Two: E-Prime General Features: NSSendEvent.

NSTrialend

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

LogAudioLatency

This inline script contains info from troubleshooting and support. It's not essential, but can be used to log additional information from the Chronos to the edat.



```
1 'NOTE for (temporary) SupportEdits version: the below reflects how we recommend time auditing the Chronos Audio
2 ' Not essential for the experiment, but adds helpful information to the DataAid file
3
4 ' Default logging
5 c.SetAttrib "Sound.RESP", "n/a"
6 c.SetAttrib "Sound.RTTime", "n/a"
7 c.SetAttrib "Sound.RTTimeMicro", "n/a"
8 c.SetAttrib "Sound.RT", "n/a"
9
10
11 ' Log the Chronos RT if available
12 If Chronos.GetState() = ebStateOpen Then
13     Dim ValidTimeComparisonBool As Boolean
14     ValidTimeComparisonBool = False
15
16     Dim nResponse As Long
17     For nResponse = 1 To Chronos.History.Count
18
19         Dim theChronosresponseData As ChronosResponseData
20         Set theChronosresponseData = CChronosResponseData(Chronos.History(nResponse))
21         If Not theChronosresponseData Is Nothing Then
22
23             c.PushNewFrame
24             If theChronosresponseData.RESP = "D" Or theChronosresponseData.RESP = "E" Then
25                 c.SetAttrib "Sound.RESP", theChronosresponseData.RESP
26                 If theChronosresponseData.RTTime - SoundStimulus.OnsetTime >= -1 Then
27                     ValidTimeComparisonBool = True
28                     c.SetAttrib "Sound.RESP", theChronosresponseData.RESP
29                     c.SetAttrib "Sound.RTTime", theChronosresponseData.RTTime
30                     c.SetAttrib "Sound.RTTimeMicro", theChronosresponseData.RTTimeMicro
31                     c.SetAttrib "Sound.RT", theChronosresponseData.RTTime - SoundStimulus.OnsetTime
32                     c.Log
33                     c.PopFrame
34                     Exit For
35                 End If
36             End If
37
38             If ValidTimeComparisonBool = False Then
39                 c.Log
40                 c.PopFrame
41             End If
42
43         End If
44
45     Next
46
47     Chronos.History.RemoveAll
48
49
50 End If
51
```

TrialDone

Label that signifies the portion of the procedure that ends the trial, if the pause is not used the procedure will continue to this point.



CountClear

This inline script just resets the counter that keeps track of keyboard responses for the pause function.

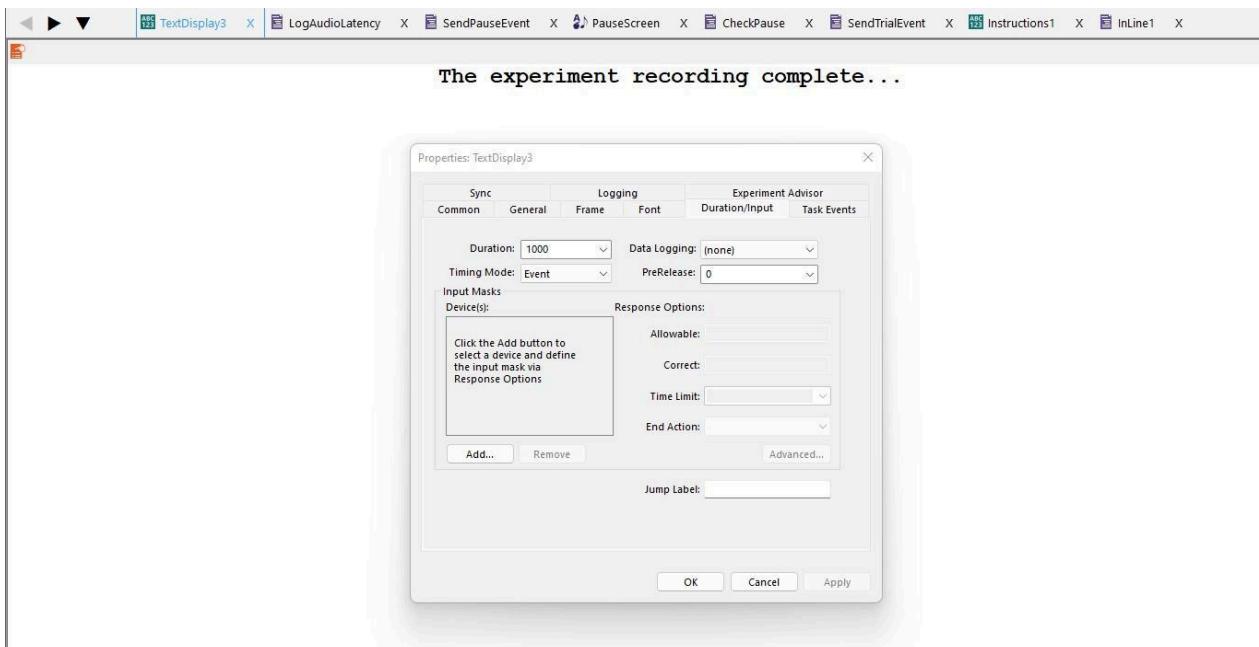
```

1 nResponseCount = 0 'counter for pause'
2

```

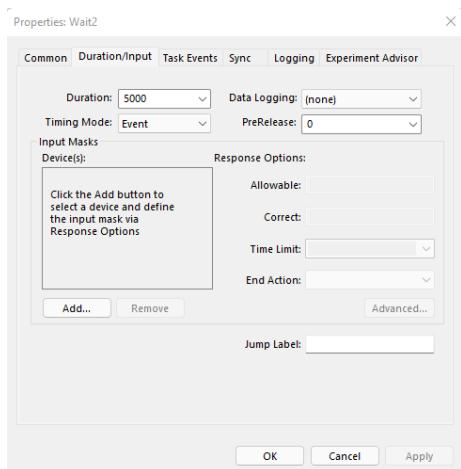
TextDisplay3

This is the end screen for the MMN task. A little different than the other tasks since it's the feature that came with the original Rutgers file. Displays text that signals that the experiment is done.



Wait2

5 second buffer after the end of the task before Net Station stops recording.



NSStopRecording

See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

Label7

This object has no settings, it is simply a label that is used as a reference point within the procedure.



Label8

This object has no settings, it is simply a label that is used as a reference point within the procedure.



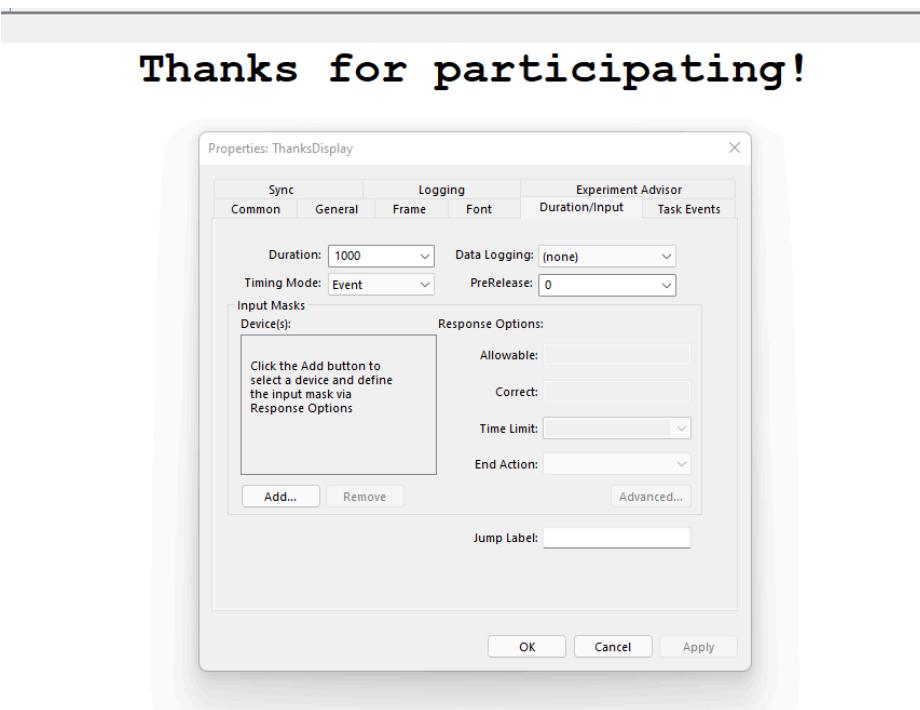
Da

The Da block is the exact same as the Ba block except that the .wav file in the trial list has been switched so that the 'da' is the standard sound. Despite this block coming second in the E-Prime file, it is counterbalanced for random selection in the Blocklist, and has an equal probability of being the version presented.

Done



ThanksDisplay



Uninitialize

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

CellList

From a design perspective there are two conditions for the MMN, but for the purposes of analyzing the data, there are 3. The conditions are standard (STND), deviant (DVNT), and preceding the deviant (PDEV).

The screenshot shows the CellList dialog box. At the top, there are several icons: a green downward arrow, a green double-headed arrow, a green right arrow, a green grid, a yellow grid, an orange clipboard with a pencil, a grey minus sign, and a black X. Below these are three lines of text: "Summary", "6 Samples (1 cycle x 6 samples/cycle)", and "1 Cycle equals 6 samples". A bold "Sequential Selection" label is present. The main area is a table:

ID	Weight	Nested	Procedure	CellNumber	CellLabel
1	1			1	STND
2	1			2	PDEV
3	1			3	DVNT
4	1			33	REEG
5	1			100	LORID
6	1			3	[Visit]

KeyList

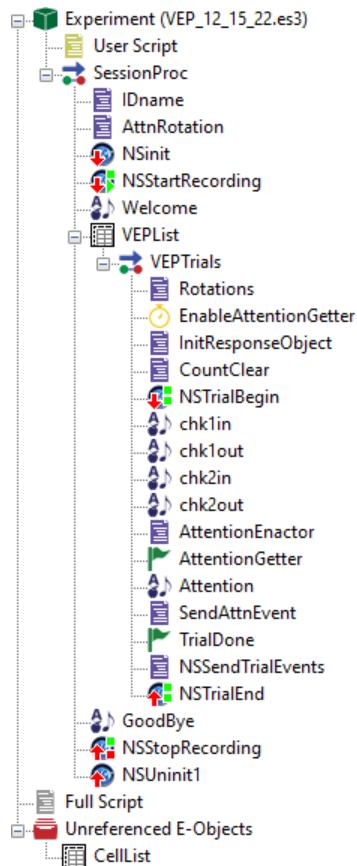
Keylist is similar to a cell list, as it is an unreferenced object that records additional information you may want logged about the parameters surrounding a particular trial. What makes it different is that it is recording additional information that is not the experimental conditions but may be needed/helpful later. Here, it is used to log which ISI the task version used and which exact sound file was used in that trial. It also logs blk+, which is a leftover variable from an older version of the tasks.

The screenshot shows the KeyList dialog box. At the top, there are several icons: a green downward arrow, a green double-headed arrow, a green right arrow, a green grid, a yellow grid, an orange clipboard with a pencil, a grey minus sign, and a black X. Below these are three lines of text: "Summary", "3 Samples (1 cycle x 3 samples/cycle)", and "1 Cycle equals 3 samples". A bold "Sequential Selection" label is present. The main area is a table:

ID	Weight	Nested	Procedure	KeyName	KeyID	KeyDataType
1	1			Target	BLK+	TEXT
2	1			Stimulus	bada	TEXT
3	1			ISI	isis	TEXT

Visually Evoked Potential (VEP)

In order to have the necessary color switching rectangles for the StimTracker (*See Part Two: E-Prime System Specs: StimTracker*), the presentation of one 500ms checkerboard was split into two 250ms checkerboards of the same image, where each one received different StimTracker rectangles. Although this has no implication for the presentation of the stimuli, it did affect how the E-prime protocol was developed. As a result, 120 trials in the experimental design translates to 60 trials in this E-Prime task, but from an observational standpoint the task presentation is the same.



User Script

The user script establishes several variables that will be further customized for use later on in the experiment. They are established here for either clarity or because the variable needs to be created early on in the experiment hierarchy.

```
1  ''' ATTENTION GETTER FUNCTIONALITY'''
2  Dim theSlide As Slide 'Declare a variable that can be assigned to the Stimulus object.
3  Dim nResponseCount As Integer 'Declare variables for holding the current response count
4  Dim theResponseData As ResponseData 'Declare variables for holding the current response data.
5  Dim UnrefSlide As Slide 'Global variable to hold Unreferenced Slide name
6  Dim theresponseObject As Wait 'Global variable (Trial pause mechanism)
7
8
9  'To Randomly rotate the attention getter
10 Dim attentiongettervars As String 'establishing a variable to hold the background filename to be called for that/the
11 Dim AttnArray(9) 'creating an array from 0 to 9, of all the attn getters available so they can be randomly selected
12
13 ''' To allow the LORIS ID and Visit number to be logged in NetStation via CellList"
14 Dim LORID As String
15 Dim Visit As String 'to label NetStation and EDat files with the visit number
16
```

SessionProc

All of the task is contained on this procedure.

IDname

See Part Two: E-Prime General Features: IDname.

AttnRotation

This script creates an array that contains the name of each of the attention getter videos. It also creates a variable that pulls a random attention getter from that array. Note: This script uses a simpler randomization scheme than the FACE Task, because attention getters are used infrequently and RAs would not use all 10 before the end of the task.

```
1  '''''' Rotating Attn Getter Functionality '''''''
2
3  'defining the array of attn getter filenames'
4  AttnArray(0) = "RoosterAttentionGetter.mp4"
5  AttnArray(1) = "TrainAttentionGetter.mp4"
6  AttnArray(2) = "CowAttentionGetter.mp4"
7  AttnArray(3) = "FrogAttentionGetter.mp4"
8  AttnArray(4) = "CarAttn_Eq.mp4"
9  AttnArray(5) = "BirdAttentionGetter.mp4"
10 AttnArray(6) = "FrenchHornAttentionGetter.mp4"
11 AttnArray(7) = "RobotAttentionGetter.mp4"
12 AttnArray(8) = "TelephoneAttentionGetter.mp4"
13 AttnArray(9) = "DuckAttn_Eq2.mp4"
14
15 attentiongettervars = AttnArray(Random(0,9))'setting one attn getter to be the first to appear before timing system kicks in'
```

NSinit

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

NSStartRecording

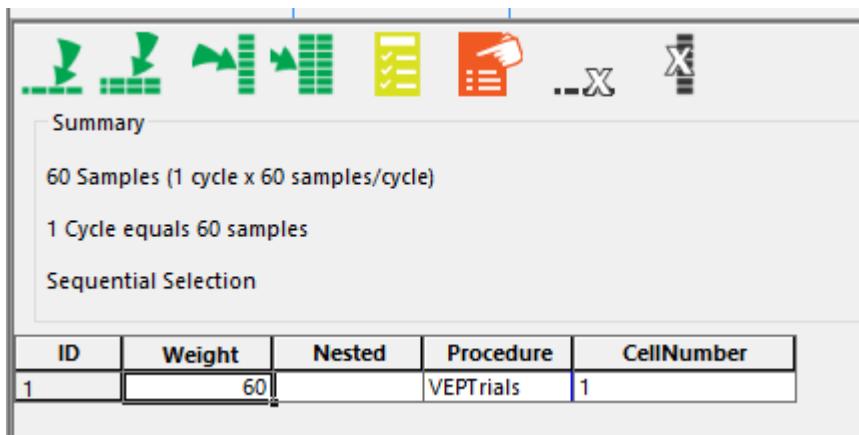
See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

Welcome

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

VEPList

Unlike other tasks, this list does not include multiple blocks. It simply runs through the trial procedure 60 times.



VEPTrials

The procedure that contains all the components of a VEP Task Trial.

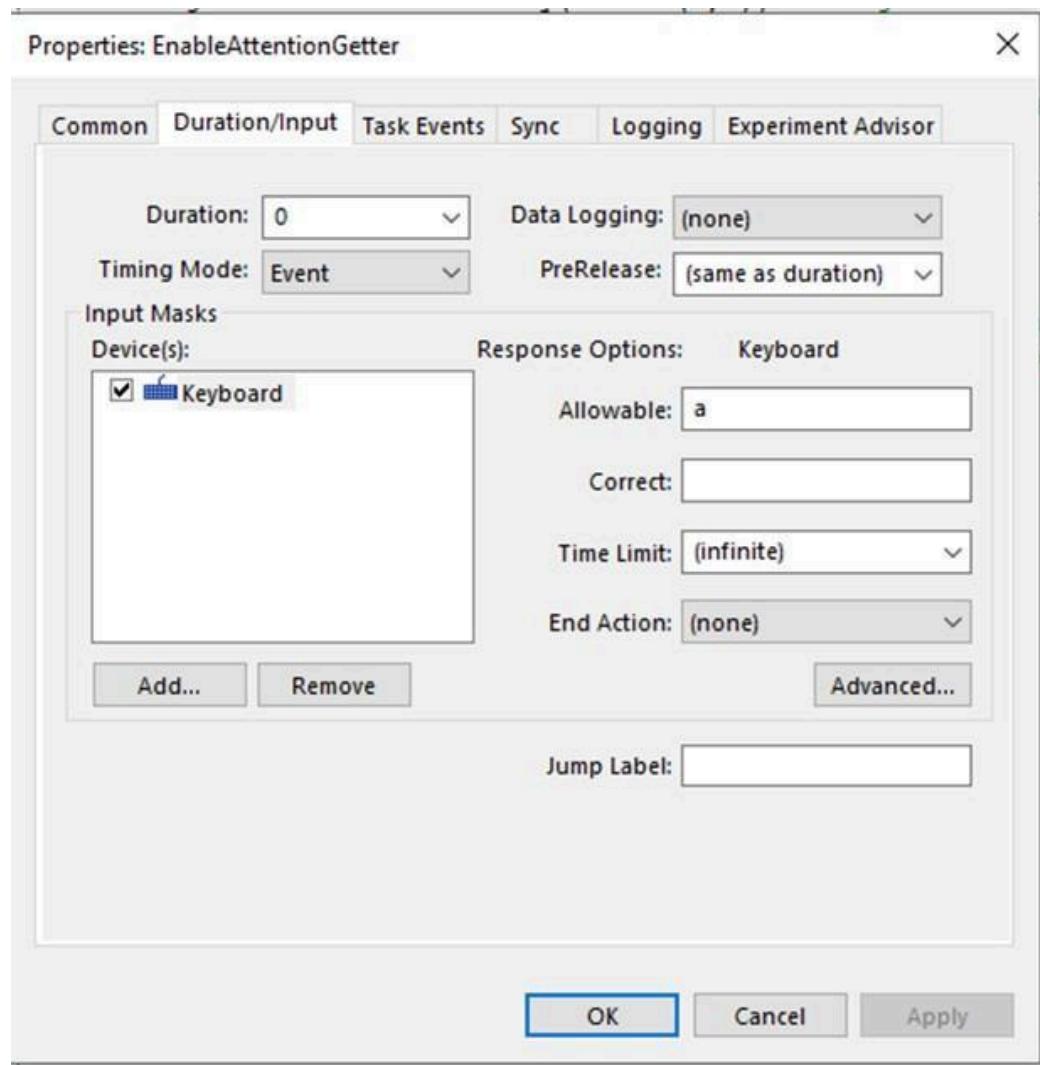
Rotations

This script ensures that at the top of every trial the attention getter is replaced with another randomly selected attention getter.

```
1  ....... mechanism for rotating attention getter.....
2
3  attentiongettervars = AttnArray(Random(0,9)) 'assign a random attn getter from the array to be placed in the trial background holding variable'
4  c.SetAttrib "AttnGet", attentiongettervars 'make the stored variable, actually the one to appear'
```

EnableAttentionGetter

This is a wait object that is set to have a duration of 0. It essentially works as a placeholder object that can be used to collect keyboard responses, in this case, 'a' to activate the attention getters.



InitresponseObject

This inline script assigns the EnableAttentionGetter wait object from above to the variable theresponseObject that was created in the user script.

```
1 | Set theresponseObject = CWait(Rte.GetObject("EnableAttentionGetter"))
```

CountClear

This inline script just resets the counter that keeps track of keyboard responses for the Attention Getter.

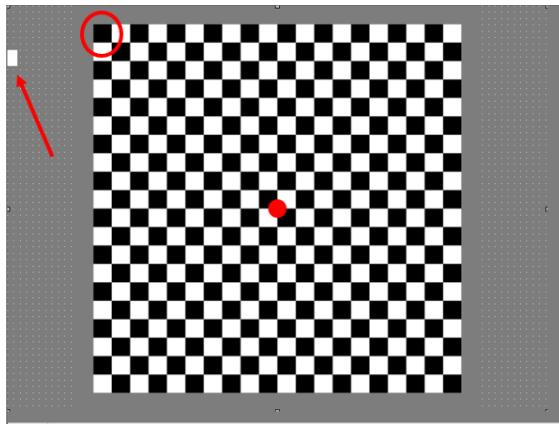
```
1 c.SetAttrib "AttnGet", attentiongettervars 'make the stored variable, actually the one to appear'
2 nResponseCount = 0 'counter for attention getting'
3 'Block1Counter = Block1Counter + 1 'counter for background mechanism'
```

NSTrialBegin

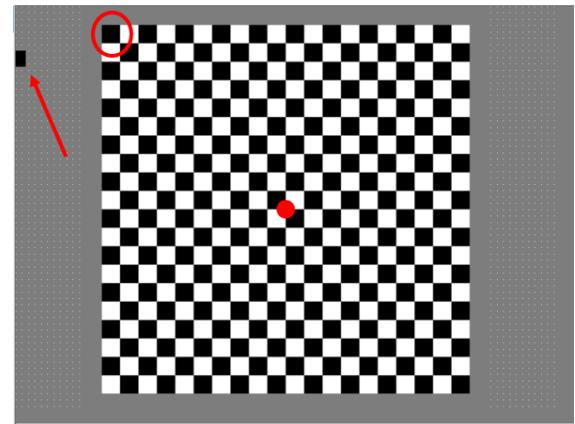
See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

Chk1in + chk1out

These two slide objects contain pictures of the same checkerboard as shown below by the red circle highlighting a corner black square. Each one has a duration of 250ms. The difference between the two is that chk1in has a white rectangle in the upper left corner while chk1out has a black rectangle in the upper left corner, noted by the red arrows. These are here for StimTracker purposes (See Part Two: E-Prime System Specs: StimTracker).



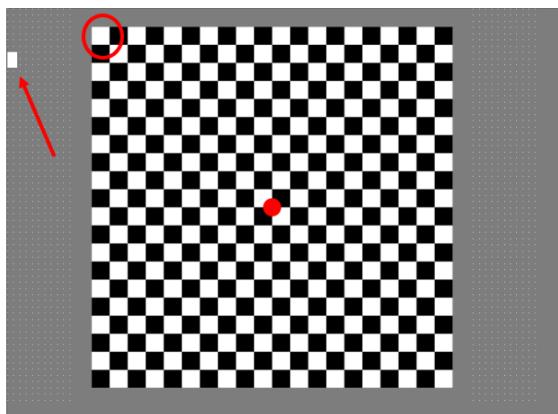
chk1in



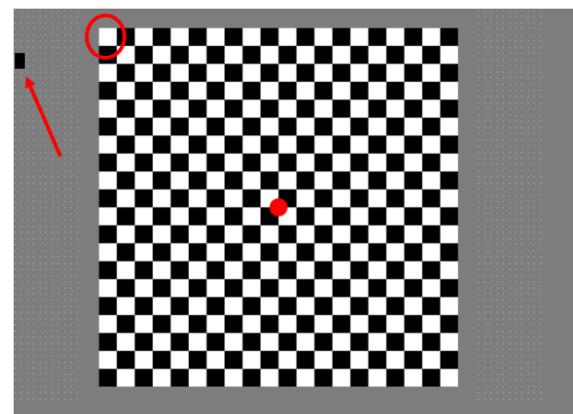
chk1out

Chk2in + chk2out

These two slide objects contain pictures of the same checkerboard as shown below by the red circle highlighting a corner white square. Each one has a duration of 250ms. The difference between the two is that chk2in has a white rectangle in the upper left corner while chk2out has a black rectangle in the upper left corner, noted by the red arrows. These are here for StimTracker purposes (See Part Two: E-Prime System Specs: StimTracker).



chk2in



chk2out

AttentionEnactor

Right before the stimulus is presented, this checks if theResponseObject (which is assigned to be the EnableAttentionGetter wait object) has registered any presses of the 'a' key. If it has, it will jump to the AttentionGetter Rag, which marks the beginning of the attention getter sequence. If no 'a' key is registered the experiment will progress linearly and finish the trial normally.

```
1 'If the counts Do Not match, there are responses To process.
2 If nResponseCount <> theResponseObject.InputMasks.Responses.Count Then
3
4     'Increment response count.
5     nResponseCount = nResponseCount + 1
6
7     'Access the current response information so that it can be used
8     'later in the script for logging and determining which action to take.
9     Set theResponseData = theResponseObject.InputMasks.Responses(nResponseCount)
10    If Not theResponseData Is Nothing Then
11
12        'Log the response to the context
13        c.SetAttrib theResponseObject.Name & ".RESP", nResponseCount, theResponseData.RESP
14        c.SetAttrib theResponseObject.Name & ".RT", nResponseCount, theResponseData.RT
15
16        'If "a" is pressed run the Unreferenced Attention Slide
17        'If theResponseData.RESP = "a" Then
18
19            'Set UnrefSlide = CSlide(Roe.GetObject("Attention"))
20
21            'If "a" Is pressed run the Unreferenced Pause Slide
22            If theResponseData.RESP = "a" Then
23
24                GoTo AttentionGetter
25
26            End If
27
28            'Must set input properties manually if running from Unreferenced E-Objects
29            'UnrefSlide.InputMasks.Add Keyboard.CreateInputMask("(SPACE)", "", CLog(UnrefSlide.Durat
30            'UnrefSlide.Run
31        End If
32
33    End If
34
35
```

In the commented-out portions of this slide are some remnants of a different attempted mechanism.
Ignore any green portions about unreferenced E-objects

Checking to see if a key press has occurred

If a key press has occurred, and it was an 'a' go to the flag marking the beginning of the attention getter sequence

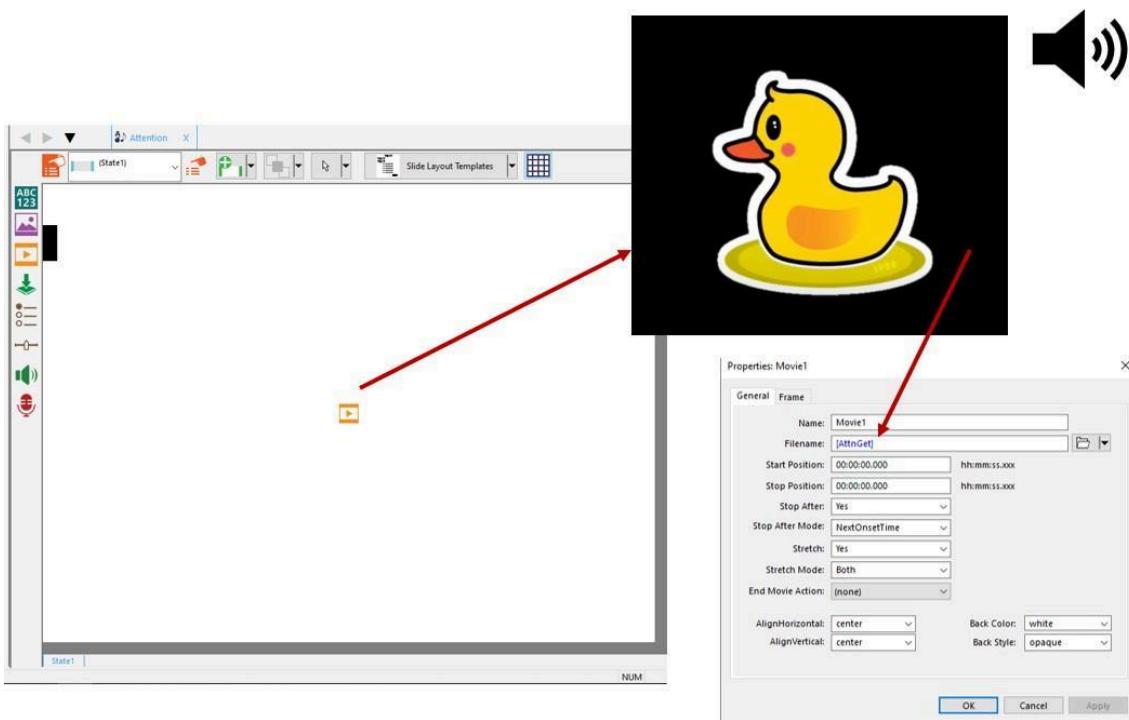
AttentionGetter

This object has no settings, it is simply a label that is used as a reference point within the procedure.



Attention

This is a slide object that contains a movie object. This is the E-object that actually presents the attention getter. Instead of having a singular file referenced in the movie object's properties, it references the 'AttnGet' attribute which holds different filenames based on which attention getter has been randomly assigned to the attribute through the AttnRotation and Rotations inline scripts.



SendAttnEvent

Inline specifically for sending event commands for attention getter. *See Part Two: E-Prime General Features: NSSendEvent*

```
1 'Send a trial event for each object
2 NetStation_SendTrialEvent c, Attention
3
```

TrialDone

This object has no settings, it is simply a label that is used as a reference point within the procedure.



NSSendTrialEvents

See Part Two: E-Prime General Features: NSSendEvent

```
1
2 'Send trial event to neststation for each presentation in trial
3 NetStation_SendTrialEvent c, chklin
4 NetStation_SendTrialEvent c, chk2in
5
6 'Send trial specific event
7 NetStation_SendTRSPEvent c, chklin
8
```

NSTrialEnd

See Part Two: E-Prime General Features: Net Station Trial begin & Net Station Trial End.

Goodbye

See Part Two: E-Prime General Features: 'Welcome' and 'Goodbye' slides.

NSStopRecording

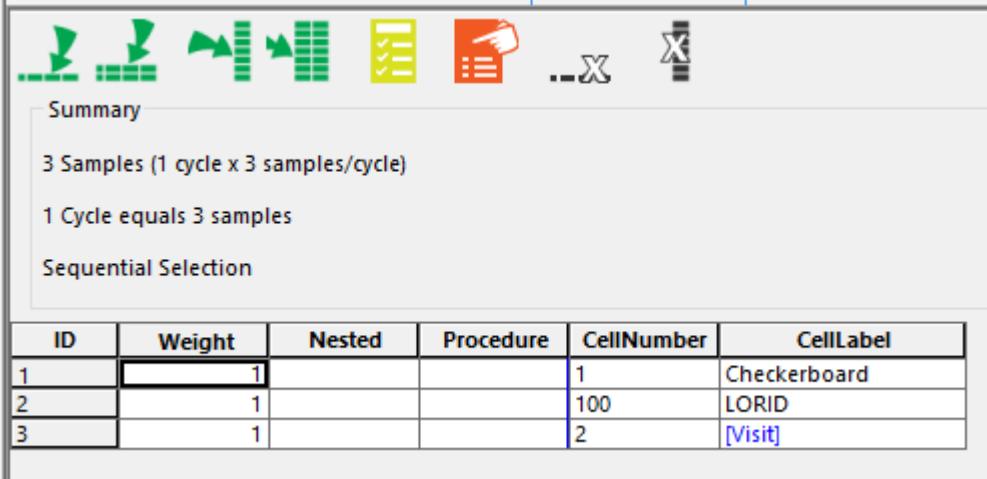
See Part Two: E-Prime General Features: Net Station Start Recording & Net Station Stop Recording.

NSUninit1

See Part Two: E-Prime General Features: Net Station Initialize & Net Station Uninitialize.

CellList

This CellList simply stores the LORID ID, Visit number, and a generic condition variable for the checkerboards.



ID	Weight	Nested	Procedure	CellNumber	CellLabel
1	1			1	Checkerboard
2	1			100	LORID
3	1			2	[Visit]

APPENDIX:

Version Release History

10.4.22

- Official release consolidating all versions up to this point.
- MMN has a pause functionality. Press 'P' during the task to pause, and spacebar to resume.
- FACE has an attention getter. Press 'A' to queue up the attention getter, and spacebar to play it. After the attention getter is done running, press spacebar to continue with the task.

12.15.22

- Face:
 - Indigenous faces were added
 - Cell conditions were changed to 1= uprightINV, 2=inverted, 3=objects, 4=uprightOBJ
 - Everytime an attention getter is activated, a random one is selected.
 - Made it such that after the attention getter is played, it goes back to the fixation cross instead of on to the next face.
 - The jitter was fixed to not be skippable in E-Prime (previously if you click quickly enough you could manage to skip the jitter). Now no matter how fast you click it will at least take the jitter duration to progress.
 - The black and white squares under the photocell were scaled down to fit more tightly under the photocell to help prevent saccades.
 - Placeholder for logging the race of the faces was removed
- MMN:
 - Instead of having the examiner manually select the ba or da block, E-prime randomly picks for them.
- VEP:
 - The black and white squares under the photocell were scaled down to fit more tightly under the photocell to help prevent saccades.
 - An attention getter function was added. A to play attention getter, space to continue.
- Baseline:
 - The black and white squares under the photocell were scaled down to fit more tightly under the photocell to help prevent saccades.

7.14.23

- Cleaned up misc. items in unreferenced objects section of each task
- Fixed some typos in the inline scripts of some tasks

- Added the ability to select blocks for Face Task. This question is prompted at startup. The selection has 3 options:
 - “Standard”: will allow E-Prime to randomly select which block occurs first, this should be how the first attempt of each participant goes, this is the default.
 - (RESTART ONLY) Upright/Inverted: this is for if after the initial attempt, the participant needs to restart just one particular block of Face. For example, if the infant successfully makes it through all of the first block, but only 10% of the second block and they want to reattempt just the second block.
 - (RESTART ONLY) Upright/Object: this is for if after the initial attempt, the participant needs to restart just one particular block of Face. For example, if the infant successfully makes it through all of the first block, but only 10% of the second block and they want to reattempt just the second block.
- Changed logging setting that was prompting 8 back to back dialogue boxes saying “Entry not permitted to be zero” everytime the E-Studio file for VEP was opened.

9.19.2023

- Replaced background images for the FACE task with images provided by UFL that had a more consistent blurring scheme.
- Modified the randomization scheme for the background images, such that a random order is determined for the image presentation. Each background image is displayed in this order before any image is repeated.
- Modified the randomization scheme for the attention getters, such that a random order is determined for the presentation. Each attention getter is displayed in this order before any is repeated.

11.29.2023

- Updated file names to include multiple visit labels (V03, V04, V06).
- Developed 2 sets of tasks, one for Visit 3 and one for Visits 4 and 6:
 - Resting State: a new video stimuli was developed for Visits 4 and 6, Visit 3 continues to use the original stimuli. Added TRSP information specifying which video stimuli was displayed
 - MMN: the inter-stimulus interval was shortened from 820ms to 600ms for Visits 4 and 6. The Visit 3 version of the task continues to use the 820ms ISI. Added TRSP information specifying which sound file was

played for each trial and which ISI was used

- VEP and FACE: these tasks were not modified for Visits 4 and 6

Additional Resources

E-Prime Google Group (E-Prime users helping other E-Prime Users):

<https://groups.google.com/g/e-prime>

E-Prime Command Reference (for inline script coding) : <https://pstnet.com/ecr/>

E-Prime Knowledge Base Documentation (in-depth explanations of settings/configurations, how-tos, known issues, official usage and compatibility guidelines, etc) :<https://support.pstnet.com/hc/en-us>

Supplemental documents such as software or equipment user manuals can be found here:

<https://drive.google.com/drive/folders/1hCgPLTDXdV9rZoyIHcGDvAWo5MzXSFC8?usp=sharing>

Support Website for the Cedrus StimTracker: <https://cedrus.com/support/stimtracker/index.htm>

Face Task Background Images

Development of Background Images

March 2022

- Geometric background images

June 2022

- Emails exchanged between the UMD team and John Richards discussing the background images.
 - The 120 s rotating background is suggested
 - The face images are grayscale, so the background images should be converted to grayscale as well
 - Geometric backgrounds were discarded in favor of simple nature backgrounds

August 2023

- Issues were raised about the background images not having the same sized concentric circles or having consistent blurring.
- Images were replaced with a new set provided by Lisa Scott's lab at UFL that resolved these issues

See [this document](#) for a full description of the development of the Face task background images

Updates to the Face Task in August 2023

In August, 2023, the randomization scheme for the background images and attention getters was updated. Additionally, the old background images were replaced with a new series of five images provided by the UFL team. These images had a consistent blurring scheme, which the previous background images did not have. The inline scripts controlling the background image rotation were also updated. In the old rotation scheme, a random background image was pulled from an array of five images every 120 seconds, but the scripts did not track which images had been displayed during the experiment. As a result, some images were displayed multiple times, while other images were not displayed at all. This method was replaced with a new randomization scheme, wherein the array of background images was randomized at the start of the experiment, and then presented sequentially in that randomly determined order. The 120 second duration for each image was kept. *See Part 2: E-Prime - Face*, for explanations of what each script controlling the background rotation does, in the event that the Face task needs to be updated again in the future. Finally, the randomization scheme for selecting attention getters was updated to match the new randomization scheme for the backgrounds. In the old version, an attention getter was randomly pulled from the list of videos each time one was called. In the updated version, the array of attention getter videos is randomized at the start of the experiment, and the attention getters are displayed in sequential order whenever the experimenter cues them.

An Important Note about Background Rotations

The background images are set to rotate every 120 seconds. However, in practice the background may not rotate exactly every 120 seconds. The background will not rotate in the middle of the trial, because the background rotating while a face is displayed for the baby would distract from the face. Additionally, the background does not rotate while an attention getter is displayed. As a result, the background rotations

can take a little longer than 120 seconds, depending on the trial length and how often attention getters are used. During the August 2023 development of the face task, the possibility of changing the EPrime code so that the background would rotate at exactly 120 seconds every time was discussed. However, it was not possible to implement this change without having the background change while a face is displayed, and it was decided that the slight delays that can occur in the existing EPrime task were not significant enough to pose an issue.

Indigenous Faces Photo Protocol

These instructions outline the parameters that were used to acquire the photos needed to expand the photo set of the Faces Task to include indigenous faces.

Criteria: Any person in OK that identifies as indigenous

1. Camera/recording: Logitech Web Camera and Zoom
 1. Zoom features: HD camera setting
2. Lighting: Used Video Conferencing light by Fansrock
 1. Name of item on Amazon: [Fansrock Video Conference Lighting, Webcam Lighting for Remote Working, Zoom Lighting for Laptop/Computer, Zoom Calls, Live Streaming, Self Broadcasting, Video Light for Zoom Meeting with Sturdy Clip](#)
3. Used 3 different light settings with 3 different ranges away from window to 1nd best well-rounded lighting for photos
4. Light Diffuser over Window: (white sheet)
 1. Name of product used: ALZO Silk Diffusion Fabric White, 1 Yard Long 60 Inches Wide
5. Participants had no makeup on, no glasses, piercings were allowed to stay in, hair out of face, entire face showing, straight face staring at laptop camera/no smiling.
- G. Photos were screenshotted off Zoom by UMD team and sent to UFL.
7. UFL team selected best for each participant.

Face Races

Each of the filenames of the face images correspond to a key to identify the race of the person.

F1-6: Asian

F7, F25, F26, F28-F30: White

F8-13: Black

F14-F19: Hispanic

F20-24 and F27: South Asian

F31-36: Indigenous

Informal Face Race Validation Survey:

This survey was shared amongst members of the HBCD EEG working group and their lab members and colleagues. This was conducted prior to the recruitment of Indigenous participants, so only 4 racial groups are included. A total of 98 total responses were recorded.

The Survey:

7/10/23, 11:52 AM

Face Race

Face Race

DIRECTIONS: Look at the picture of the face in each item. For each item select which race you believe best represents the face shown. There are 24 faces in total.

* Indicates required question

1. Email *

2. 1 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

3. 2 *



Mark *only one oval*.

Asian
 Black
 White
 Hispanic/Latinx

4. 3'



Mark *only one oval*.

Asian
Black
White
Hispanic/Latinx

5. 4 *



Mark *only one oval.*

Asian
Black
White
Hispanic/Latinx

6. 5 *



Mark *only one oval*.

Asian
Black
White
Hispanic/Latinx

7. 6 *



Mark *only one oval*.

Asian
Black
White
Hispanic/Latinx

8. 7 *



Mark *only one oval*.

Asian
Black
White
Hispanic/Latinx

9. 8 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

10. 9 ^r



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

11. 10 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

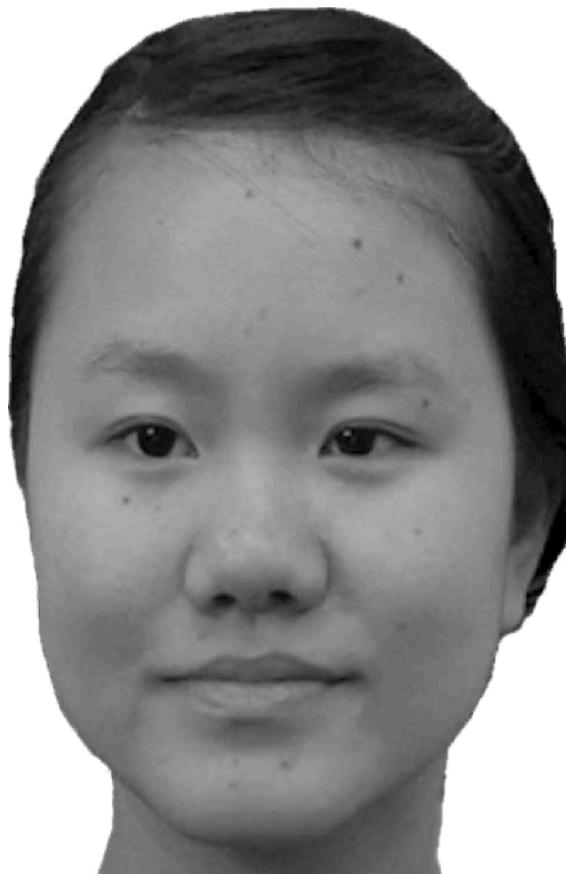
12. 11 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

13. 12 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

7/10/23, 11:52 AM

Fac* Race

14. 13 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

15. 14 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

16. 15 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

17. 16 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

18. 17 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

19. 18*



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

20. 19 *



Mark only one oval.

Asian
 Black
 White
 Hispanic/Latinx

21. 20 *



Mark only one oval.

- Asian
- Black
- White
- Hispanic/Latinx

22. 21 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

23. 22 *



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

24. 23 *



Mark only one oval.

- Asian
- Black
- White
- Hispanic/Latinx

2b. 24'



Mark only one oval.

Asian
Black
White
Hispanic/Latinx

This content is neither created nor endorsed by Google.

The responses:

Here are the aggregated responses to each item:

7/10/23, 11:57 AM

Face Race

Face Race

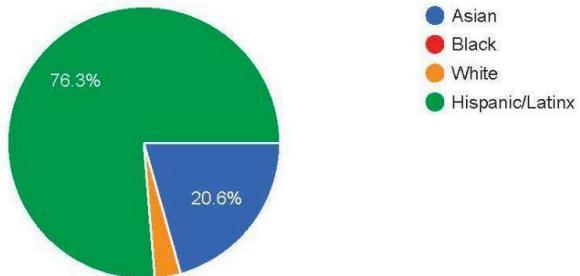
97 responses

[Publish analytics](#)

1

 Copy

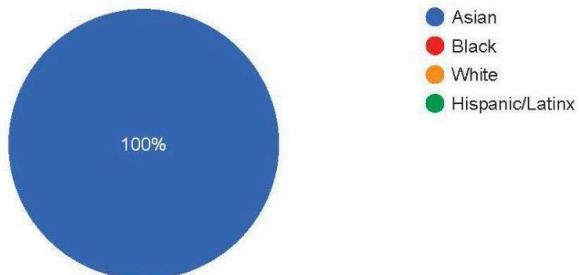
97 responses



2

 Copy

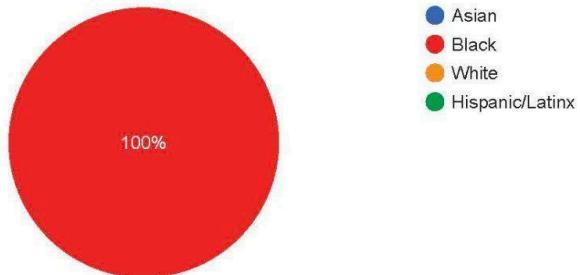
97 responses



3

 Copy

97 responses



7/10/23, 11:57 AM

Face Race

4

LV Copy

97 responses

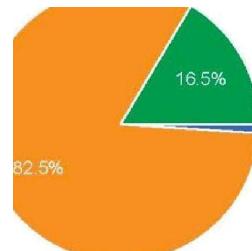
@ Asian @ Black @ White
Hispanic/Latinx

LV Copy

5

97 responses

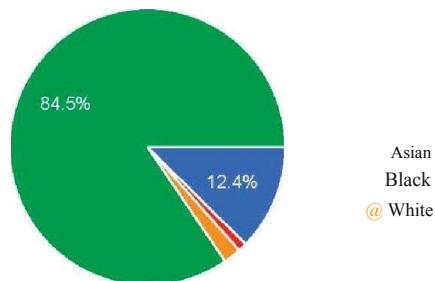
@ Asian @ Black @ White
@ Hispanic/Latinx



LV Copy

6

97 responses



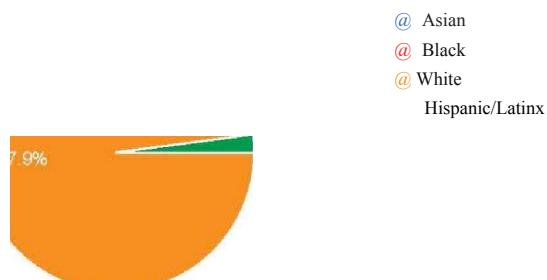
7/10/23, 11:57 AM

Face Race

LV Copy

7

97 responses



LV Copy

8

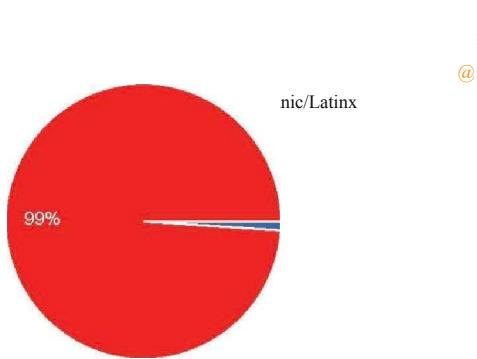
97 responses



LV Copy

9

97 responses



329

b tpe://docx, geoglt.
cov/foimx/d1gBLK92GMG76xMwTPHO63cB3boJq9/XqwvJw
tYPco3-0/viewanaiyñcl

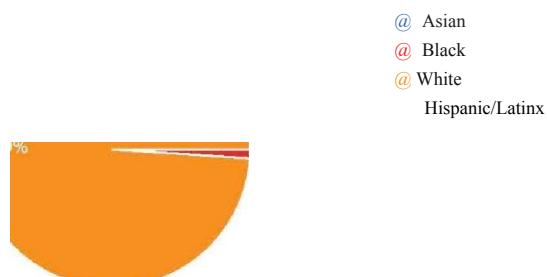
7/10/23, 11:57 AM

Face Race

10

LV Copy

97 responses



11

LV Copy

97 responses

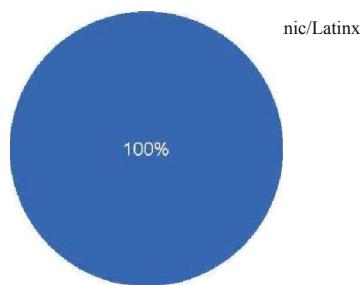


12

LV Copy

97 responses

Asian
White
Black
Hispanic/Latinx



7/10/23, 11:57 AM

Face Race

LV Copy

13

97 responses

@ Asian @ Black @ White
Hispanic/Latinx



LV Copy

14

97 responses

@ Asian @ Black @ White
@ Hispanic/Latinx



LV Copy

15

97 responses



@ Hispanic/Latinx

Asian
Black
@ White
@ Hispanic/Latinx

7/10/23, 11:57 AM

Face Race

LV Copy

16

97 responses

@ Asian @ Black @ White
Hispanic/Latinx

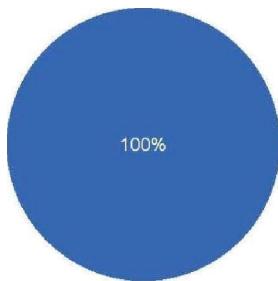


LV Copy

17

97 responses

@ Asian @ Black @ White
@ Hispanic/Latinx



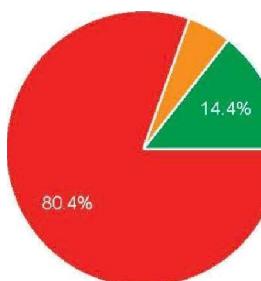
LV Copy

18

97 responses

Asian
Black
@ White

nic/Latinx



7/10/23, 11:57 AM

Face Race

LV Copy

19

97 responses

@ Asian @ Black @ White
Hispanic/Latinx

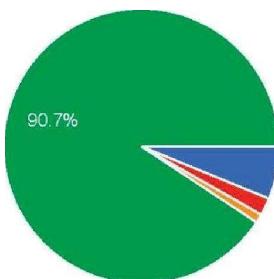


LV Copy

20

97 responses

@ Asian @ Black @ White
@ Hispanic/Latinx



LV Copy

21

97 responses

Asian
Black
@ White
@ Hispanic/Latinx



7/10/23, 11:57 AM

Face Race

22

LV Copy

97 responses

@ Asian @ Black @ White

Hispanic/Latinx

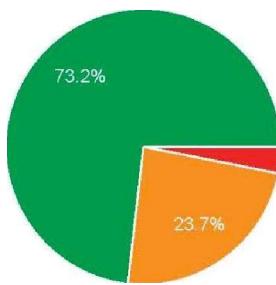


LV Copy

23

97 responses

@ Asian @ Black @ White
@ Hispanic/Latinx

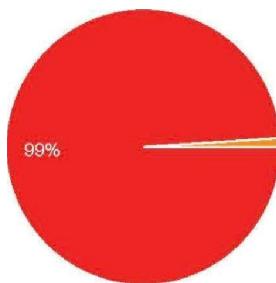


LV Copy

24

97 responses

Asian
Black
@ White
@ Hispanic/Latinx



This survey was created and endorsed by Google. [Report Abuse](#) [Terms of Service](#) [Privacy Policy](#)

Google Forms

Benasich MMN Modification Manual

Instructions for altering the Rutgers Infancy Studies Lab's auditory oddball experiment template

1. Convert sound files in Audacity 2.3.1.0

E-prime support has told us that our sound files need to have the following properties: .wav file format, 2 channels (stereo), 15 bit sample, 48 kHz. The following instructions are how we edit our sound files to this format in the free Audacity program:

1. Obtain original sound files from 300 system
2. Drag and drop the original sound into Audacity
3. Click within track box after the sound to move cursor
4. Click 'Generate/ Silence' and change duration to 1 second
5. Using time shift tool, move the silent track until it connects with the end of the sound
- G. Click arrow next to sound name to view menu, change format to 16-bit PCM
7. Select whole track (Command-A). Copy audio track (Command-C). Paste track below first one (Command-V) Make sure you see two identical tracks next to each other.
8. On top track, click arrow by name again, then click 'Make stereo track'
9. At bottom of screen, change project rate to 48000 Hz
10. File/ Export/ Export as Wav
11. Repeat for all sounds individually
12. Place sound files in the same folder as the e-prime oddball experiment template

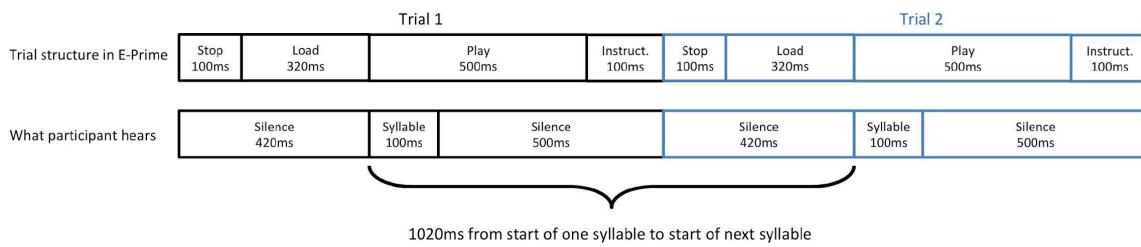
2. Change E-Prime settings

1. Open the experiment template in E-Prime
2. Review screenshots of the settings at the end of this document and change your settings to match
3. Try running the experiment before making any changes to ensure it works with your system

3. Alter E-Prime experiment to fit your experiment's requirements

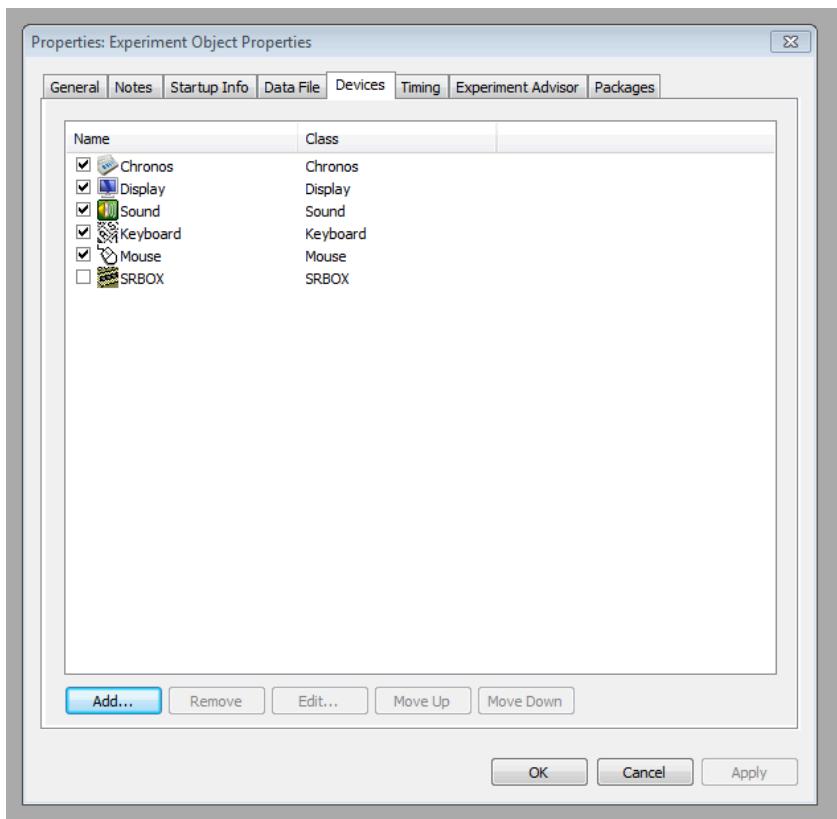
1. Double click on the TrialList object. This list determines which sound is played on each trial in the experiment. Trials are presented in the order displayed here.
 - a. SoundType and CellNumber are numeric codes for each stimulus
 - b. In the Stimulus column, type the name of the sound file you want to play on each trial
 - c. We have found it easier to edit this list in a text file or excel spreadsheet and then copy/paste the final result into the TrialList object.
2. Next, adjust the trial length to fit the requirements of your experiment:

- a. For the Visit 3 paradigm, our speech sounds play for 100ms, then there are 920ms of silence between each sound. When you view the 'Elapsed Time' column in the Net Station timing test, you will see a total of 1020ms between each stms or DIN2 marker.
- b. For Visit 3, the trial length of 1020ms is achieved within E-prime using four different objects (SoundStimulusStop, SoundStimulusLoad, SoundStimulus, and Instructions1). The durations of the SoundStimulusStop (100ms), SoundStimulusLoad (320ms), SoundStimulus (500ms), and Instructions1 (100ms) objects sum to a total of 1020ms. It is recommended that each object have a duration of at least 100ms, to allow time for the next object to load.

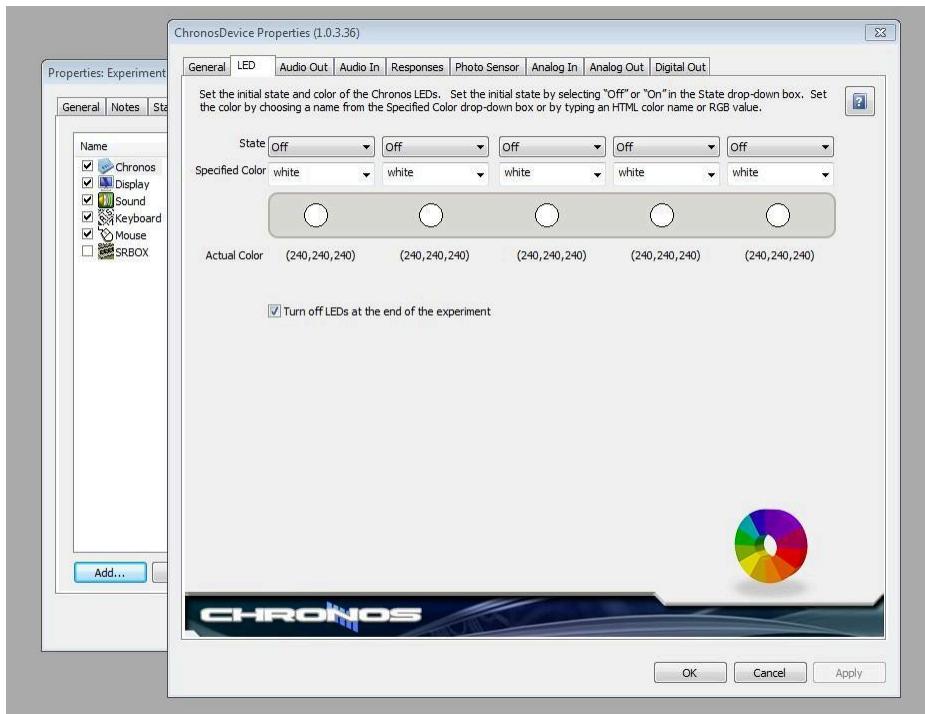
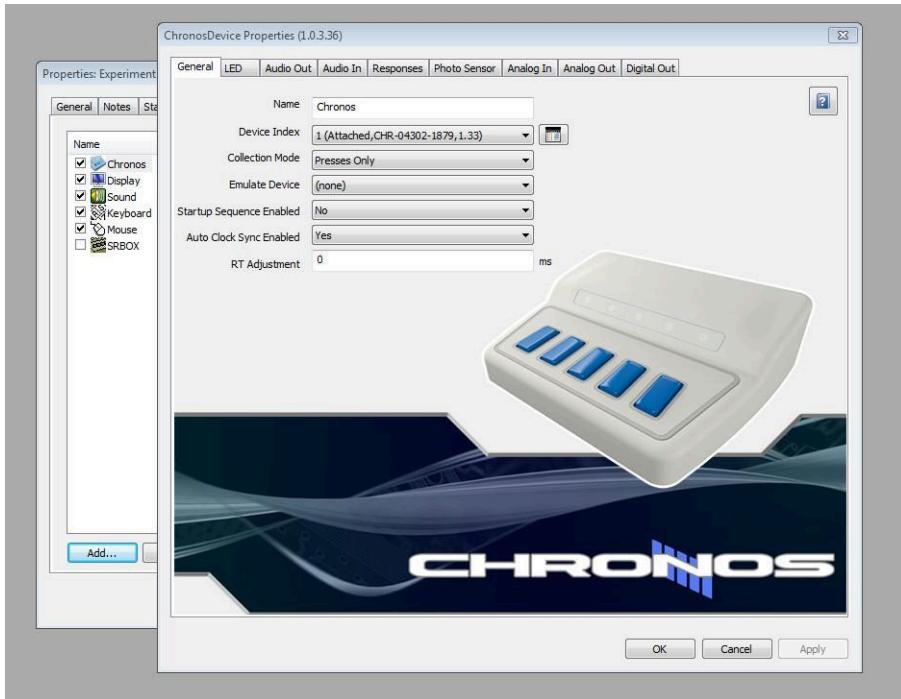


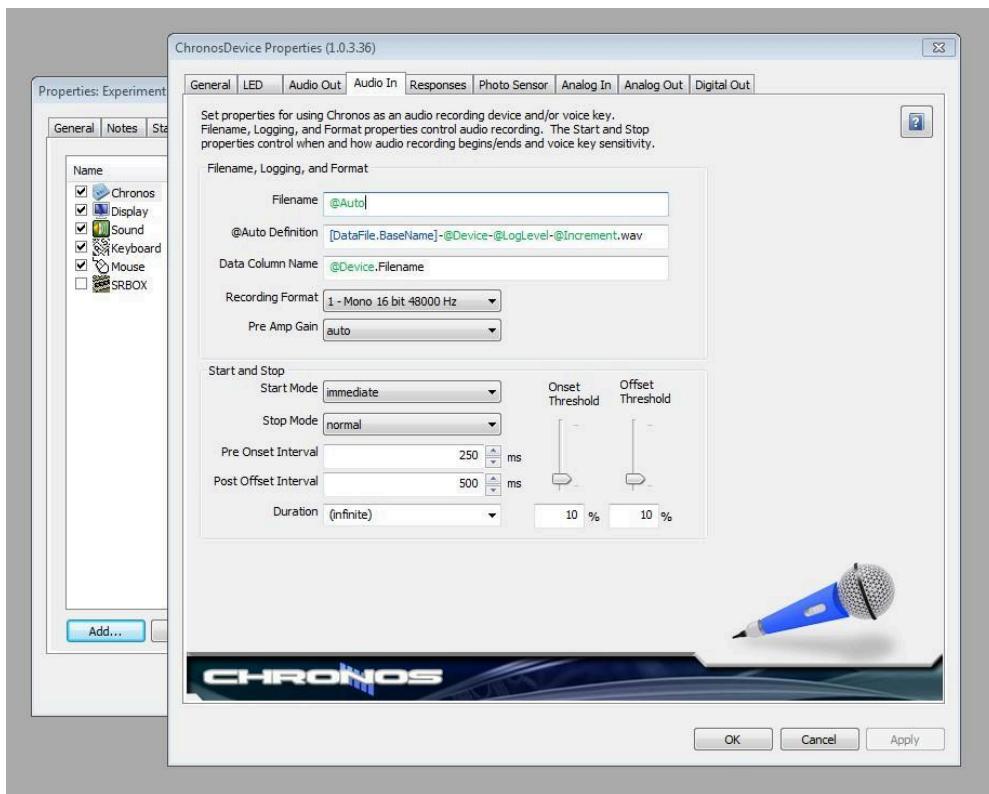
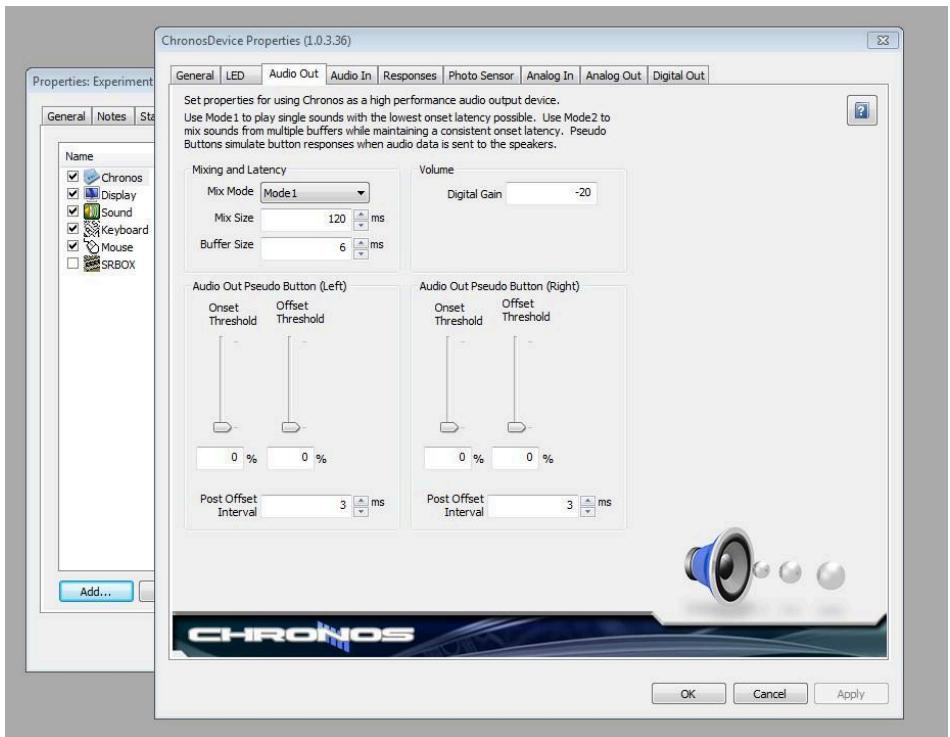
- c. To change the length of the silent interval for Visits 4 and 6, the Instructions 1 object was deleted (removing 100ms) and the SoundStimulusLoad object duration was changed to 200ms (removing 320 ms)
 - d. To change the length of the silent interval for different experimental paradigms, we can alter the duration of the SoundStimulusLoad object. You could also alter the SoundStimulus duration, but this is the object that actually plays the sound file so its duration needs to be at least as long as your stimulus.
3. Update the following display objects with the appropriate information for your experiment. These do not affect the experiment in any way, they are only used to provide the experimenter with information about the paradigm.
 - a. Open the CallList object inside BlockProc. Starting on line 23, alter the message string to show the appropriate characteristics of the experiment you're working on (experiment type, order file, etc).
 - b. Edit the TextDisplay5 object in the same manner
 - c. Edit the TextDisplay5Synch object in the same manner (can copy/paste from TextDisplay5)
 4. Save and test the experiment

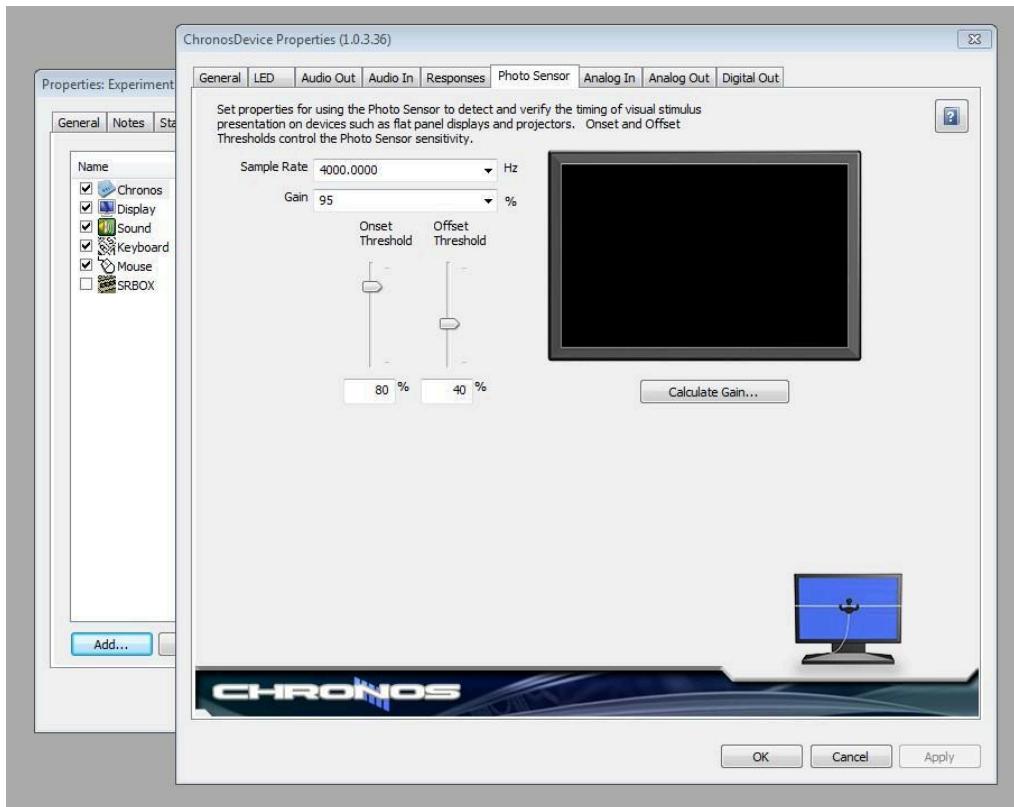
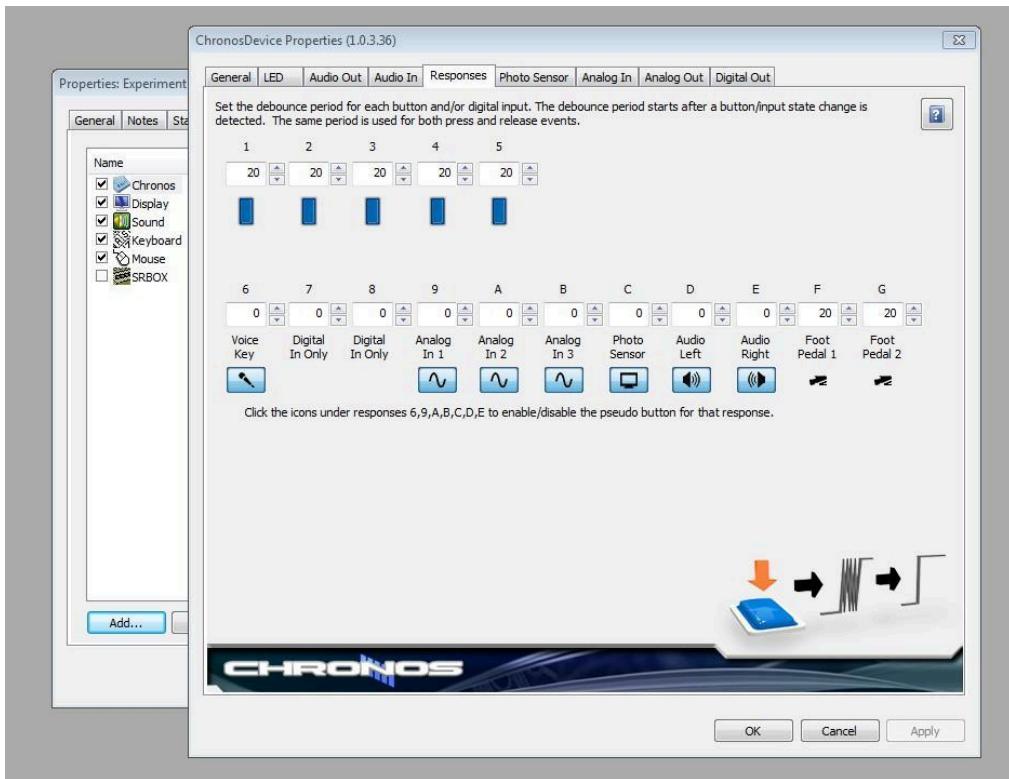
E-Prime Settings

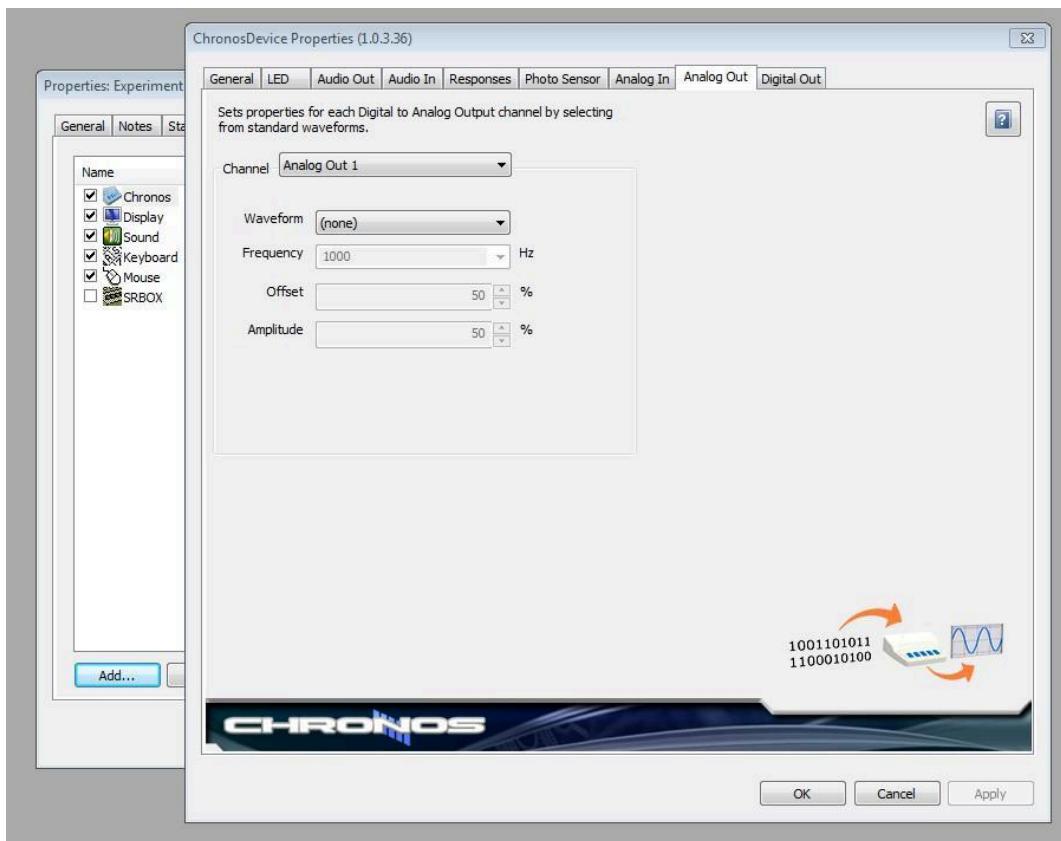
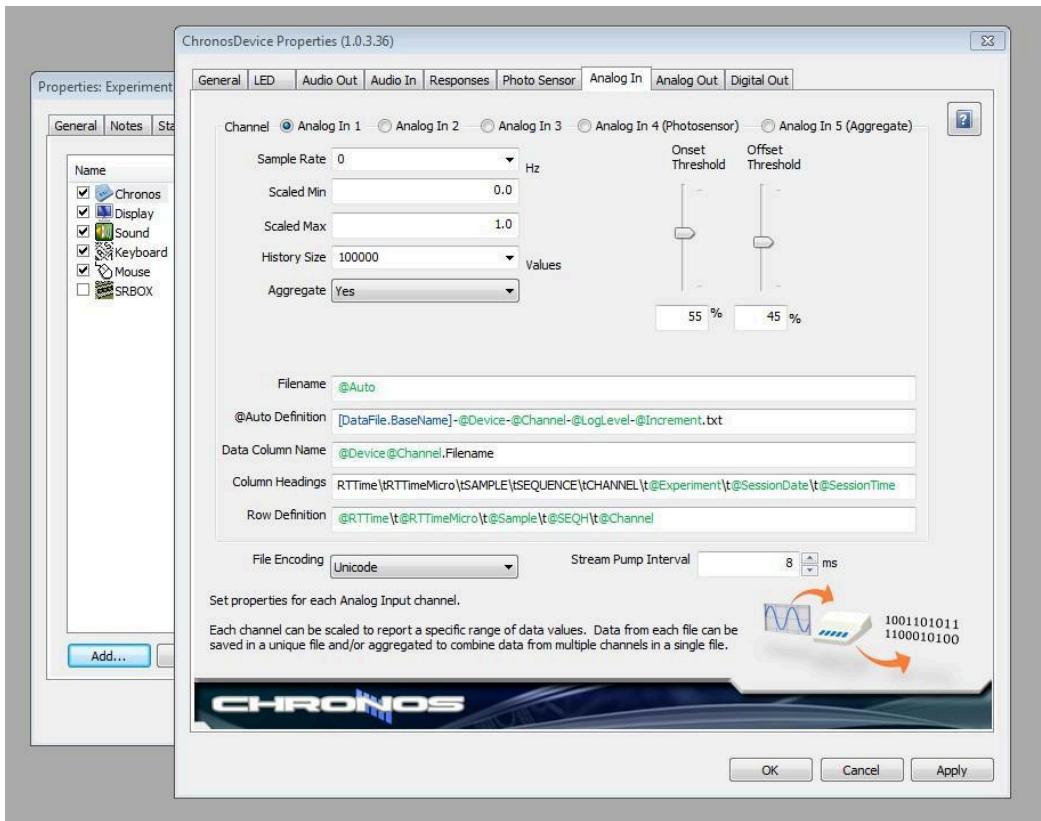


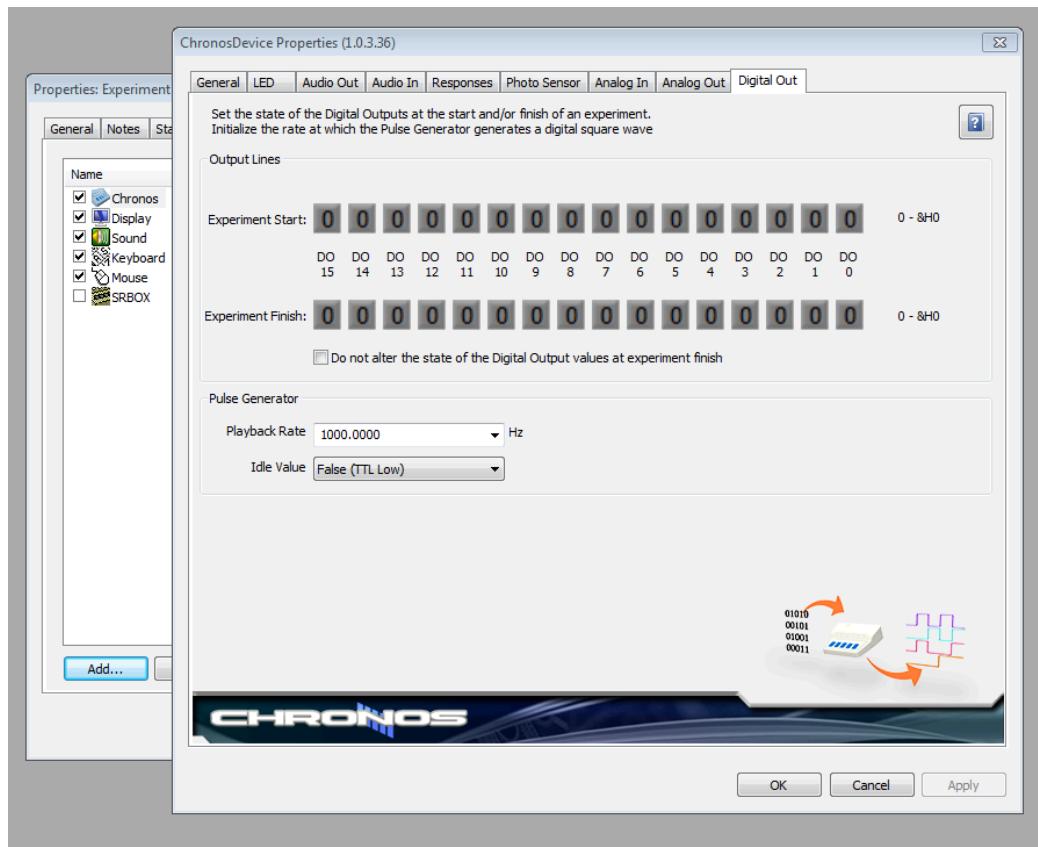
Note: Chronos must be the first device!

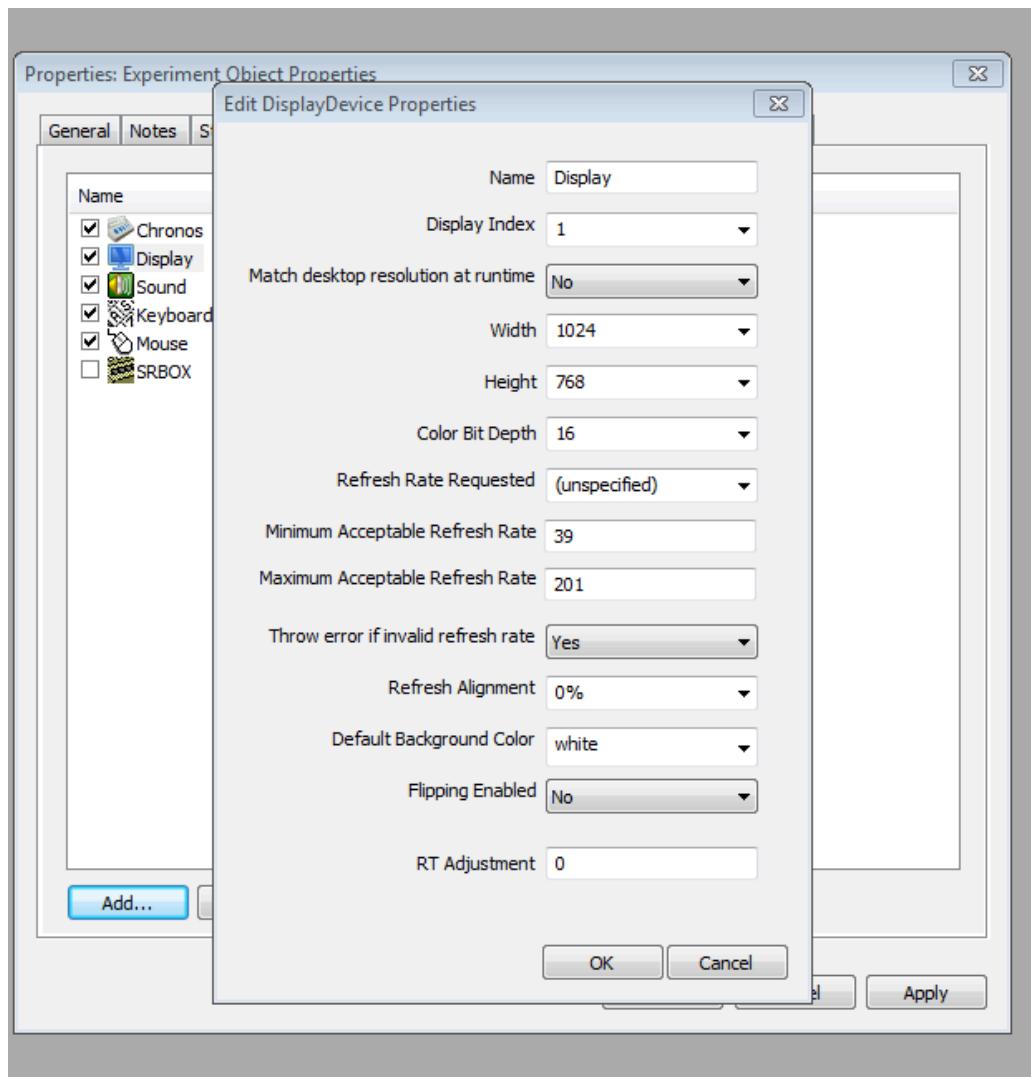


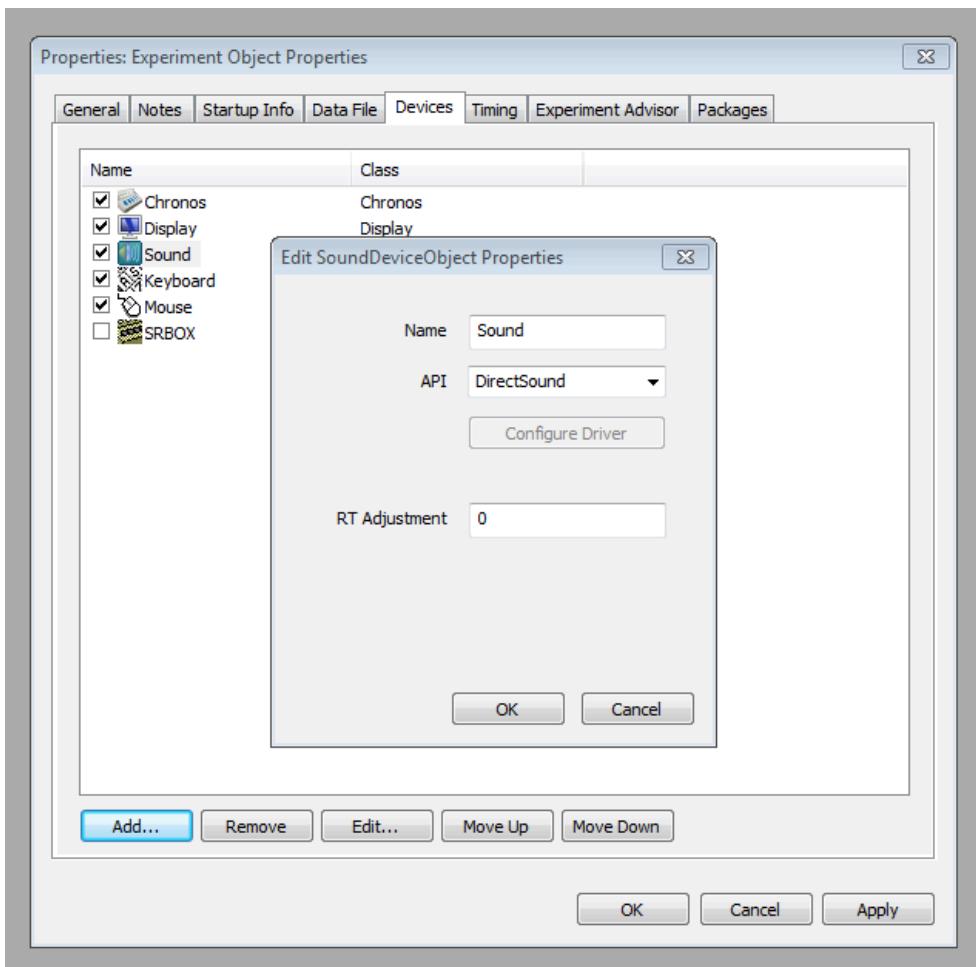




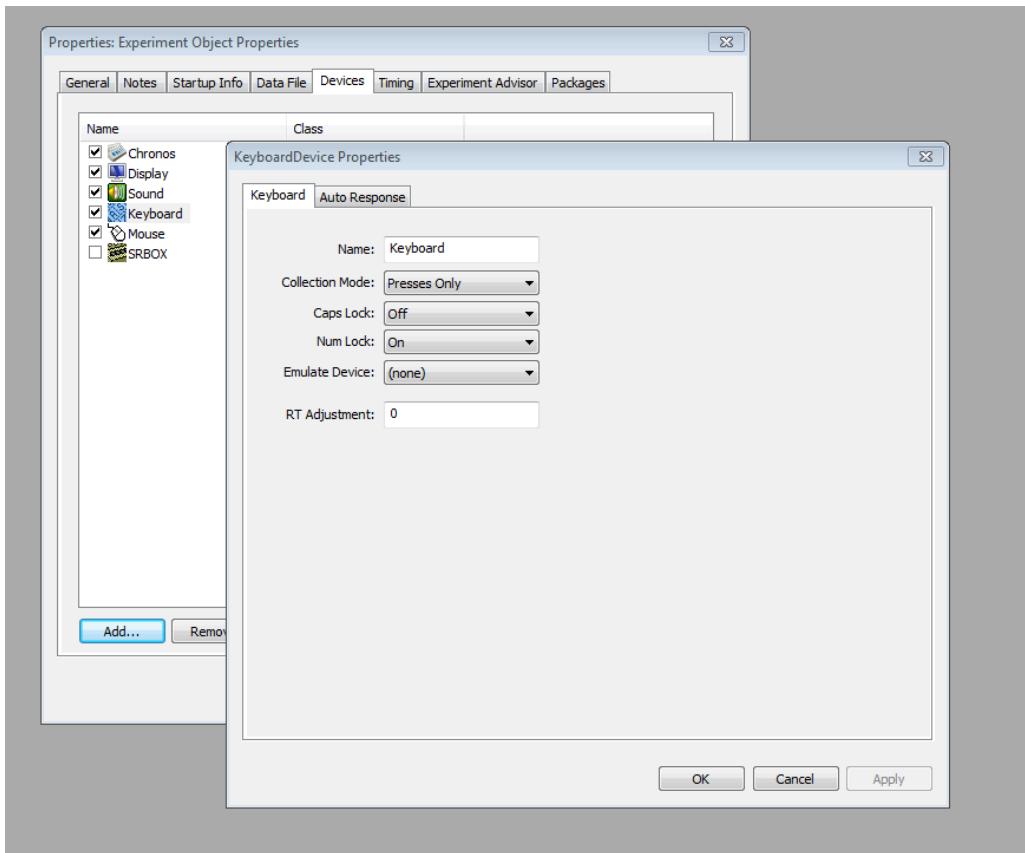


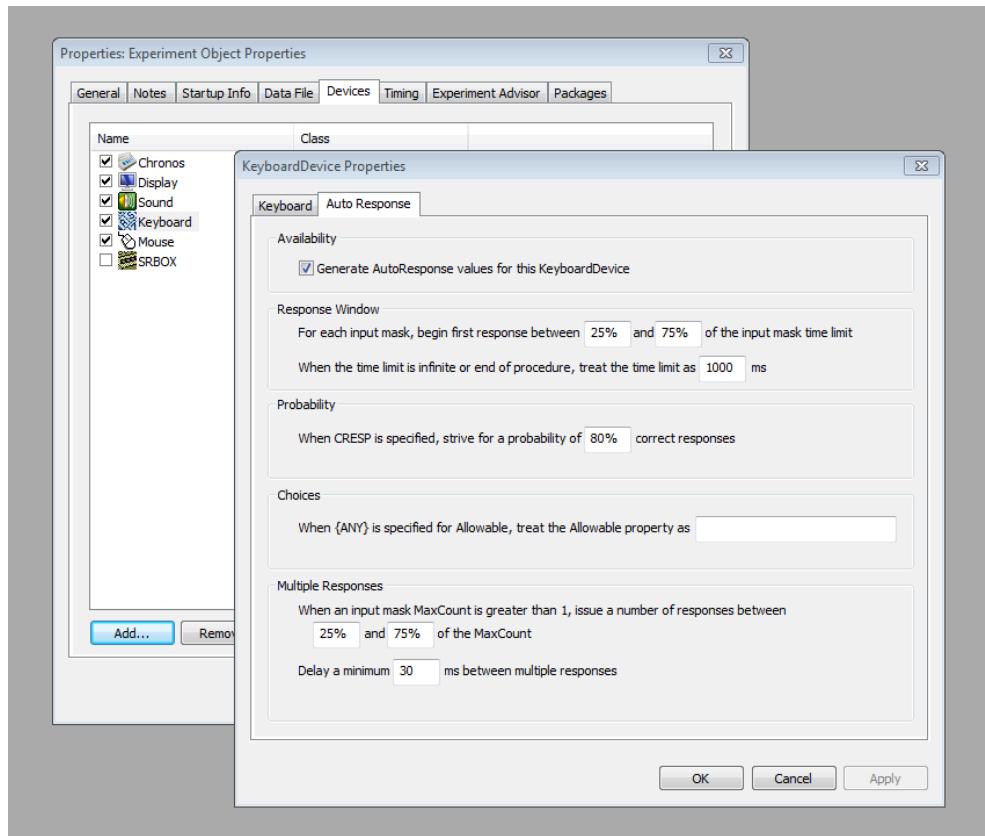


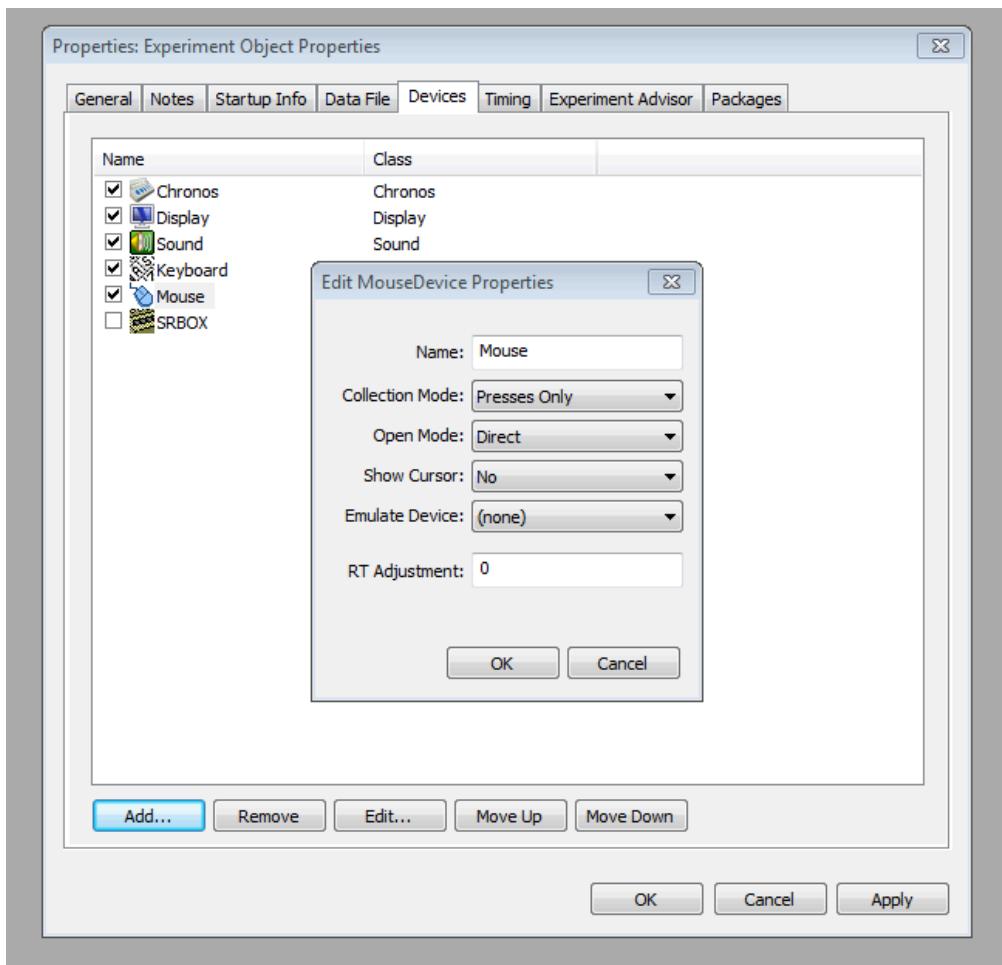


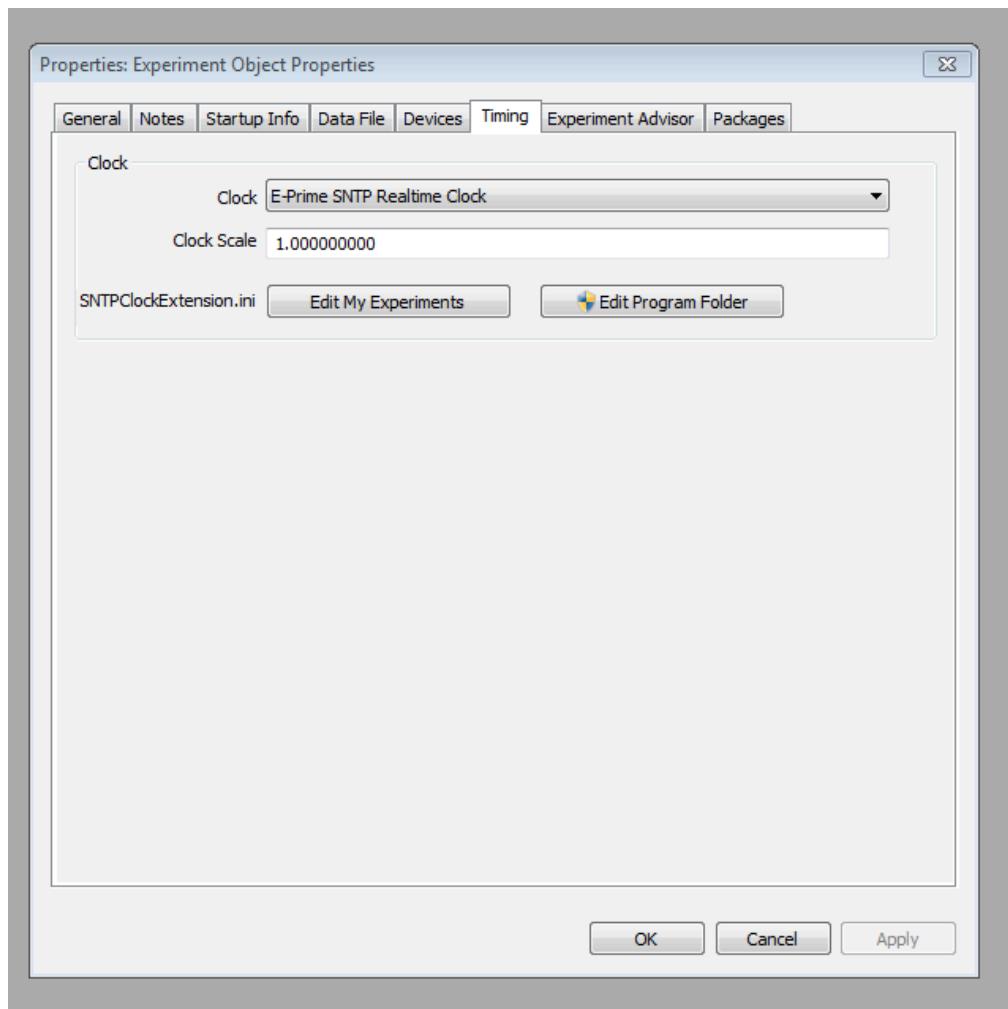


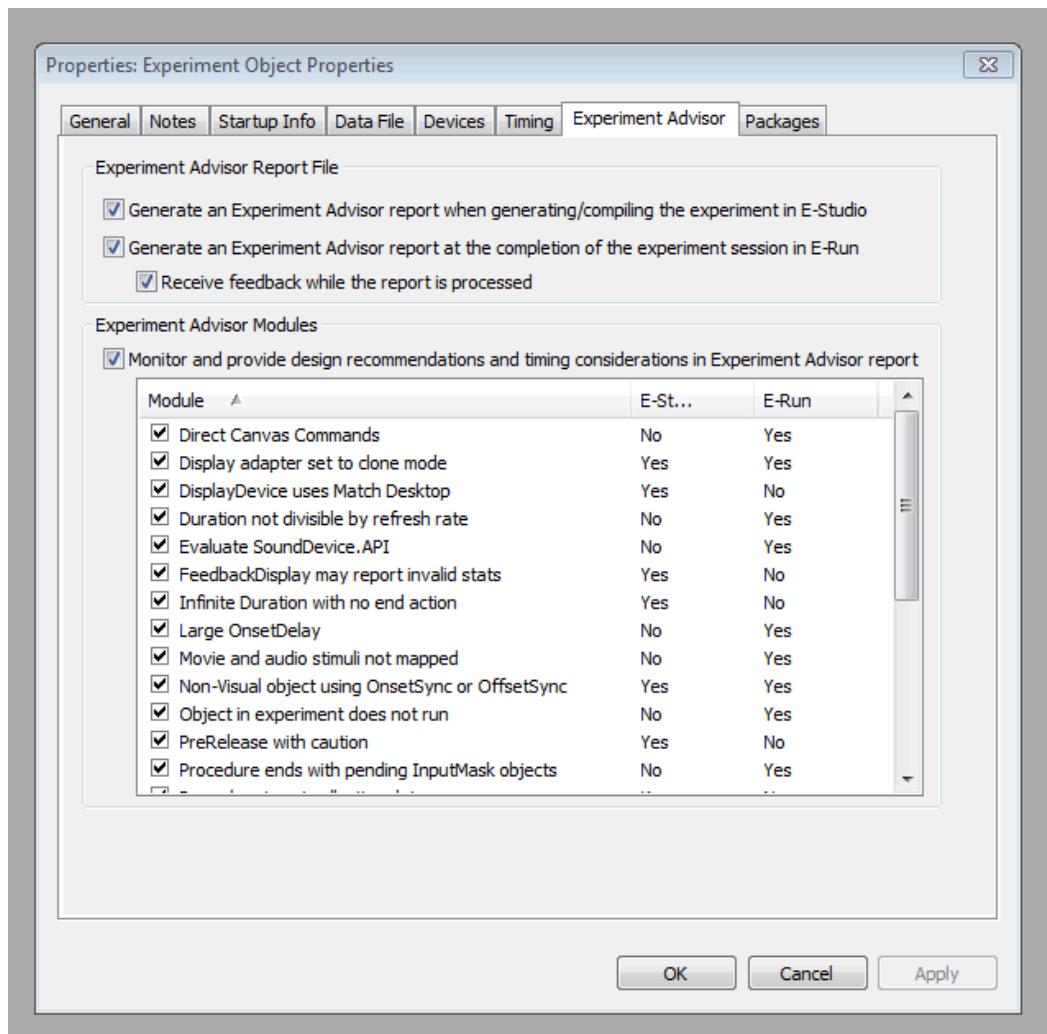
Note: E-Prime support emphasized the importance of having the API set as DirectSound

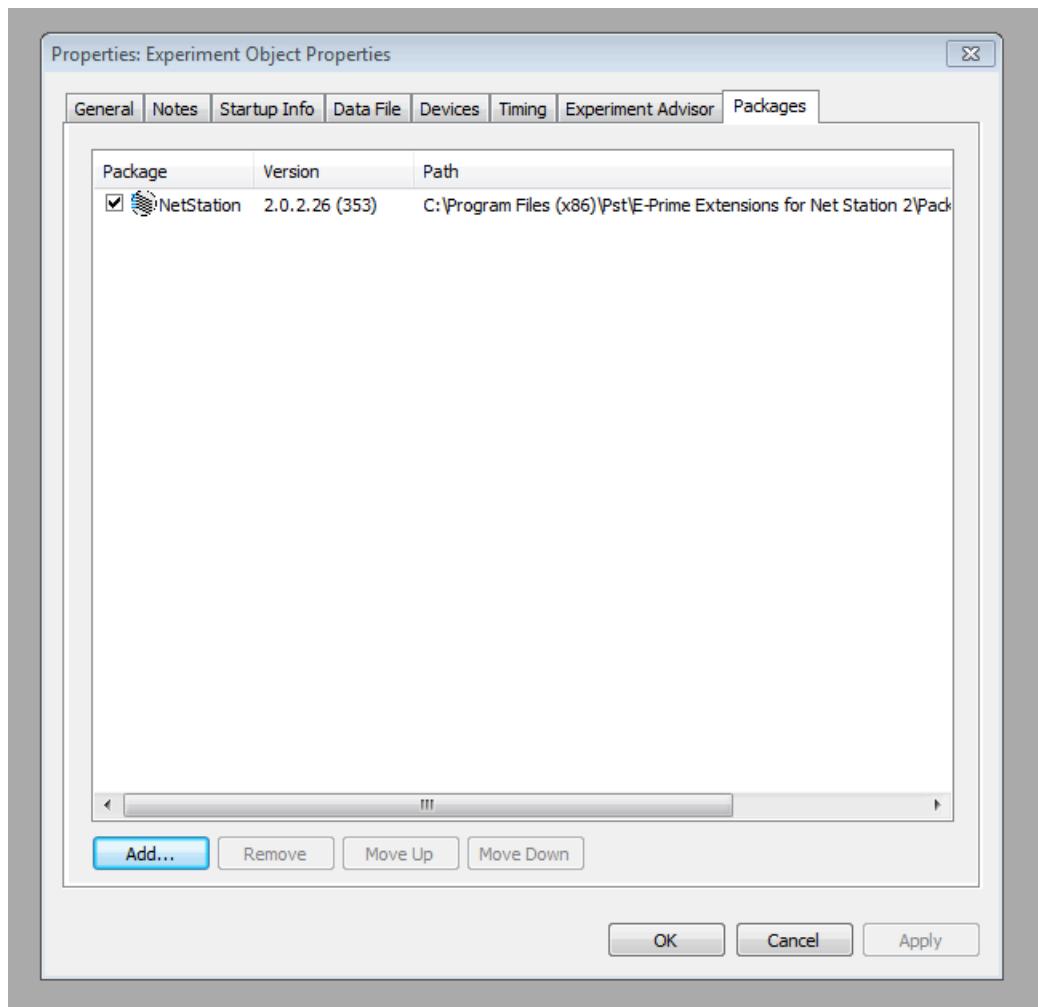












MMN Trial List

Conditions 1= Standard 2= Deviant

Trial Nu	Condition						
1	1	169	1	337	2	505	1
2	1	170	1	338	1	506	1
3	1	171	1	339	1	507	1
4	1	172	1	340	1	508	2
5	1	173	2	341	1	509	1
6	1	174	1	342	1	510	1
7	1	175	1	343	2	511	1
8	1	176	1	344	1	512	1
9	1	177	1	345	1	513	1
10	1	178	1	346	1	514	1
11	2	179	1	347	1	515	2
12	1	180	1	348	2	516	1
13	1	181	1	349	1	517	1
14	1	182	2	350	1	518	1
15	1	183	1	351	1	519	1
16	2	184	1	352	1	520	1
17	1	185	1	353	1	521	1
18	1	186	1	354	1	522	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
19	1	187	2	355	1	523	1
20	1	188	1	356	2	524	1
21	1	189	1	357	1	525	2
22	1	190	1	358	1	526	1
23	1	191	1	359	1	527	1
24	1	192	1	360	1	528	1
25	2	193	1	361	2	529	1
26	1	194	2	362	1	530	1
27	1	195	1	363	1	531	1
28	1	196	1	364	1	532	1
29	1	197	1	365	2	533	1
30	1	198	1	366	1	534	1
31	1	199	1	367	1	535	1
32	1	200	1	368	1	536	2
33	1	201	1	369	1	537	1
34	1	202	1	370	1	538	1
35	1	203	2	371	2	539	1
36	2	204	1	372	1	540	2
37	1	205	1	373	1	541	1
38	1	206	1	374	1	542	1
39	1	207	1	375	1	543	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
40	1	208	1	376	1	544	1
41	2	209	1	377	2	545	1
42	1	210	1	378	1	546	1
43	1	211	1	379	1	547	1
44	1	212	2	380	1	548	2
45	1	213	1	381	1	549	1
46	1	214	1	382	2	550	1
47	2	215	1	383	1	551	1
48	1	216	1	384	1	552	1
49	1	217	1	385	1	553	1
50	1	218	1	386	2	554	1
51	1	219	1	387	1	555	2
52	2	220	1	388	1	556	1
53	1	221	1	389	1	557	1
54	1	222	1	390	1	558	1
55	1	223	2	391	2	559	1
56	1	224	1	392	1	560	1
57	1	225	1	393	1	561	2
58	1	226	1	394	1	562	1
59	1	227	1	395	1	563	1
60	1	228	2	396	2	564	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
61	1	229	1	397	1	565	1
62	1	230	1	398	1	566	1
63	2	231	1	399	1	567	1
64	1	232	1	400	1	568	2
65	1	233	1	401	2	569	1
66	1	234	1	402	1	570	1
67	1	235	1	403	1	571	1
68	1	236	2	404	1	572	1
69	1	237	1	405	1	573	1
70	2	238	1	406	1	574	1
71	1	239	1	407	1	575	2
72	1	240	1	408	1	576	1
73	1	241	1	409	2	577	1
74	1	242	1	410	1	578	1
75	1	243	1	411	1	579	1
76	1	244	2	412	1	580	1
77	1	245	1	413	1	581	1
78	2	246	1	414	2	582	2
79	1	247	1	415	1	583	1
80	1	248	2	416	1	584	1
81	1	249	1	417	1	585	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
82	1	250	1	418	1	586	1
83	1	251	1	419	1	587	1
84	2	252	1	420	1	588	1
85	1	253	1	421	2	589	1
86	1	254	2	422	1	590	1
87	1	255	1	423	1	591	2
88	1	256	1	424	1	592	1
89	1	257	2	425	1	593	1
90	2	258	1	426	2	594	1
91	1	259	1	427	1	595	1
92	1	260	1	428	1	596	1
93	1	261	1	429	1	597	1
94	1	262	2	430	1	598	1
95	1	263	1	431	1	599	1
96	1	264	1	432	1	600	1
97	1	265	1	433	1	601	2
98	2	266	1	434	2	602	1
99	1	267	2	435	1	603	1
100	1	268	1	436	1	604	1
101	1	269	1	437	1	605	1
102	1	270	1	438	2	606	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
103	1	271	1	439	1	607	2
104	1	272	1	440	1	608	1
105	2	273	1	441	1	609	1
106	1	274	2	442	1	610	1
107	1	275	1	443	1	611	1
108	1	276	1	444	1	612	1
109	1	277	1	445	2	613	1
110	2	278	1	446	1	614	1
111	1	279	2	447	1	615	2
112	1	280	1	448	1	616	1
113	1	281	1	449	1	617	1
114	1	282	1	450	2	618	1
115	1	283	1	451	1	619	1
116	1	284	2	452	1	620	1
117	1	285	1	453	1	621	1
118	2	286	1	454	1	622	1
119	1	287	1	455	1	623	2
120	1	288	2	456	1	624	1
121	1	289	1	457	1	625	1
122	1	290	1	458	1	626	1
123	2	291	1	459	1	627	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
124	1	292	1	460	2	628	1
125	1	293	1	461	1	629	2
126	1	294	1	462	1	630	1
127	2	295	2	463	1	631	1
128	1	296	1	464	1	632	1
129	1	297	1	465	1	633	1
130	1	298	1	466	2	634	1
131	1	299	1	467	1	635	2
132	1	300	1	468	1	636	1
133	1	301	2	469	1	637	1
134	1	302	1	470	1	638	1
135	2	303	1	471	1	639	1
136	1	304	1	472	2	640	1
137	1	305	1	473	1	641	1
138	1	306	1	474	1	642	1
139	1	307	1	475	1	643	1
140	1	308	2	476	1	644	1
141	1	309	1	477	1	645	2
142	1	310	1	478	2	646	1
143	2	311	2	479	1	647	1
144	1	312	1	480	1	648	1

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
145	1	313	1	481	1	649	1
146	1	314	1	482	1	650	1
147	2	315	1	483	2	651	1
148	1	316	2	484	1	652	1
149	1	317	1	485	1	653	1
150	1	318	1	486	1	654	1
151	1	319	1	487	1	655	1
152	1	320	1	488	1	656	2
153	1	321	1	489	1	657	1
154	2	322	2	490	1	658	1
155	1	323	1	491	1	659	1
156	1	324	1	492	2	660	1
157	1	325	1	493	1	661	1
158	1	326	1	494	1	662	1
159	1	327	1	495	1	663	2
160	1	328	1	496	1	664	1
161	1	329	2	497	1	665	1
162	1	330	1	498	1	666	1
163	2	331	1	499	2	667	1
164	1	332	1	500	1		
165	1	333	1	501	1		

Trial	Condition	Trial	Condition	Trial	Condition	Trial	Condition
Nu		Nu		Nu		Nu	
m		m		m		m	
166	1	334	1	502	1		
167	1	335	1	503	1		
168	1	336	1	504	1		

Cedrus Polarity Blog Post

Link here : <https://www.cedrus.com/blog/update-for-egi-users.htm>

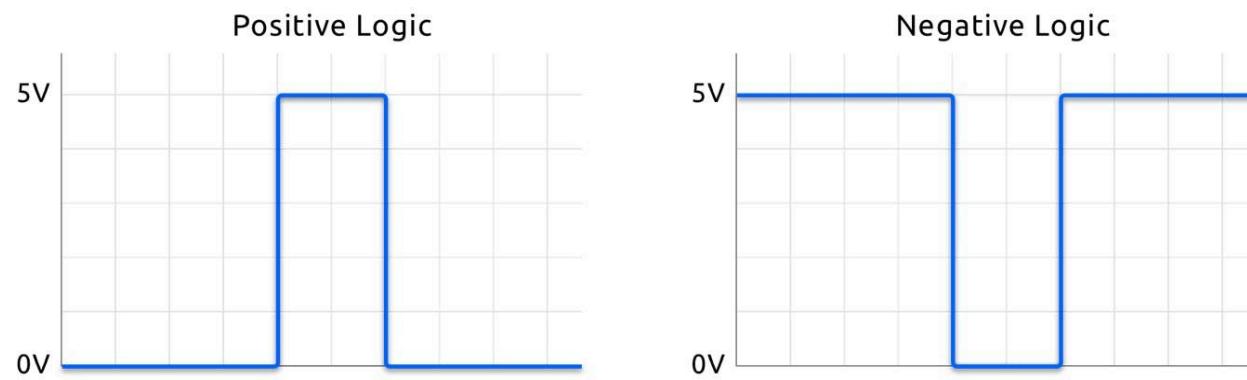
Blog An Important Update for EGI Users

Jun 13, 2022

Recently, while providing support for a lab that is using EGI, two issues came to our attention. If you have used any Cedrus equipment to send event markers, e.g. [StimTracker](#), [c-pod](#), or [m-pod](#), this blog post applies to you; read on. Even if you used other software and hardware such as E-Prime, Chronos, or PsychoPy, you might still find this blog post relevant.

First Issue – It's a Matter of Logic

Positive logic is what most users are familiar with. The output voltage from m-pod is normally zero volts (0V). When sending an event marker for, say, 10ms, the output voltage rises to +5V and drops back to zero after 10ms.



With negative logic, it's the reverse: the output voltage is normally +5V. When sending an event marker, the voltage drops to zero and then goes back up to +5V after 10ms:

While positive logic is more popular, it is not unusual for electronic devices to adopt negative logic.

If you were wondering aloud “who cares?”, you would not be wrong. After all, we want our hardware to just work and we do not care which logic is used inside. The trouble arises when two devices want to communicate with each other and they use different logics.

What Happened?

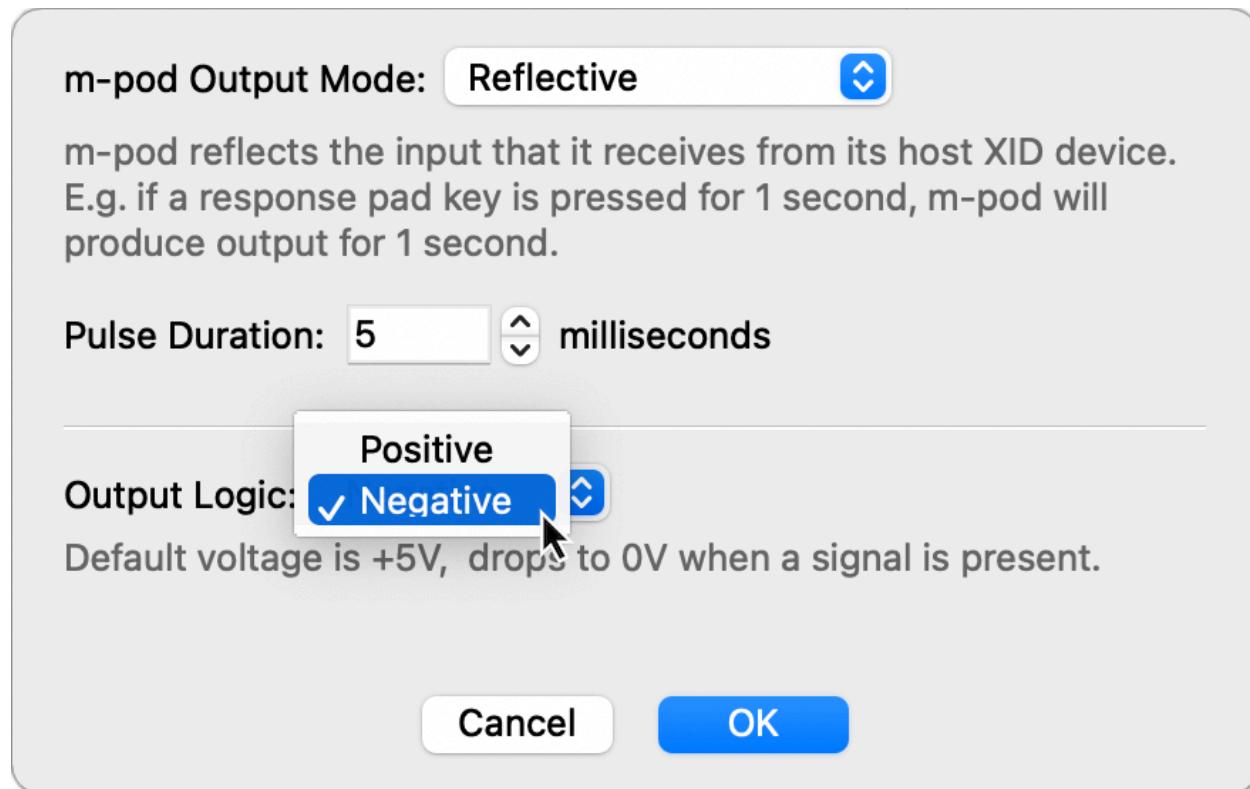
After developing the interface between the original StimTracker ST-100 model and Net Amps, Cedrus worked closely with EGI to test compatibility and create a [support page](#) on how to setup Net Station for accepting event markers. This was done using Net Station version 4.5.5.

During this joint effort, the fact the Cedrus and EGI are using different logics was somehow missed.

In practice, this means that Net Station was recording the offset of a signal instead of its onset. For example, if you were playing a sound for 100ms and StimTracker was marking this sound and generating a corresponding 100ms TTL pulse, Net Station was recording when the sound has finished playing instead of when it started to play.

The Fix

After we became aware of this difference, we updated the firmware for StimTracker Duo and Quad, m-pod, and c-pod. Version 2.2.7 provides support for negative logic; see [Changing m-pod's Output Mode and Logic](#):



The RB Series response pads and Lumina do not require a firmware update. The hardware in the original StimTracker ST-100 model is hardwired for positive logic and cannot be changed via a firmware update.

Second Issue – Net Station 5 Does Not Record Offsets

Net Station 4.5 offered the option to record the onset of an event marker or its offset (but not both). Starting with Net Station version 5, some software options were removed. When using negative logic with Cedrus products, Net Station 5 no longer recorded the offset of a signal – only its onset. This is not unusual either: once in a while, companies will remove some features in order to tame software complexity. Cedrus is currently [doing something similar](#).

If you are using an m-pod, a workaround is to change its output mode from the default “Reflective” to “Double Pulse” mode. Using the example above of playing a 100ms audio, Reflective mode would have m-pod generate a 100ms pulse but Double Pulse mode would have it generate two short pulses instead, allowing Net Station to record both the onset and the offset; see [Changing m-pod’s Output Mode and Logic](#). A pulse duration of 1ms is recommended.

There is no built-in workaround if you are using c-pod.

How You Can Correct Your Data

This is undoubtedly the harder part. If you were not aware of negative logic being used, then your data analysis is off. For example:

- If you were sending a 20ms event marker via USB at time T, Net Station 5 was recording the event at time T+20ms.
- If you were having StimTracker mark the onset of audio or visual stimuli, Net Station 5 was recording the offset instead.

The best way to correct, we think, depends on the types of event markers that you were collecting:

- If you were sending only event markers via USB, the correction is easy: subtract the pulse duration from your recorded offsets in order to obtain the onset.
- If you were having StimTracker mark the onset of visual and auditory stimuli, make sure to update the firmwares first and then re-run the same experiment, preferably a few times. You will then be obtaining the onset of events and can compare to the offsets and subtract the difference in order to correct.

In Summary

- EGI uses negative logic. It always has.
- Cedrus products used positive logic until recently. Firmware version 2.2.7 or later lets you use either.
- Net Station 4 was able to record the onset or the offset of an event marker, but not both.
- Net Station 5 records only the onset.
- When the logic does not match, i.e. Cedrus products using positive logic, then what you thought was the onset was actually the offset.

Other Devices

As of this writing, the only other device that we know of that uses negative logic is SAGA by TMSi.

See Also

- [Updating firmware for StimTracker Duo and Quad](#)
- [Updating firmware for m-pod](#)
- [Updating firmware for c-pod](#)

E-Objects and Corresponding Event Flags

Task	E-Object	Corresponding Event Flag
Resting State	TextDisplay1	phan
	BaseVideo	bas+
Face	Stim	stm+
	Fixation	1x+
	Jitter	ITI+
	Attention	dist
MMN	SoundStimulus	stms
	PauseScreen	paus
VEP	chk1in	Ch1+
	chk2in	Ch2+

EEG Parameters Graphic

Electroencephalography (EEG) Parameters

Primary Equipment	Net Station Software 5.4.1*	E-Prime 3.03.80	Net Amps 400*	HydroCel Geodesic Sensor Nets 130: 128ch*	Axis M1065-L PTZ Camera
Additional Equipment	Cedrus StimTracker	Hypertronics- to-DB9-Cable	Behringer Personal Monitor Speakers (MS16)	R8050 Sound Level Meter	Coleman 1000 Lumens LED Lantern (lowest light level)

*Items from Electrical Geodesics, Inc

Sampling Rate	Participant Monitor Model	Participant Monitor Distance	Sound Level	Lantern Distance
1000 Hz	Lenovo ThinkVision T2054p 19.5 inches	60 cm from middle of participant chair	75 decibels (72-78 acceptable range)	4 ft behind participant chair

Tasks	Acquisition Time (mins)	Conditions	# of Trials	Stimulus Duration (ms)	Inter-Stimulus Interval (ms)	Total Trial Length (ms)
Faces Task	4:30	Upright & Inverted, Upright & Object	200	500	500-700	1000-1200
Mismatch Negativity/ Auditory Oddball Task	11:30	Standard, Deviant	667	200	820	1020
Video Resting State	3:00	N/A	N/A	N/A	N/A	N/A
Visual Evoked Potential Task	1:00	Checkerboard A, Checkerboard B	120	500	0	500

HEALthy Brain and Child Development

Babies • Brains • Bright Futures

HBCDStudy.org



References

- Conte, S., Richards, J. E., Guy, M. W., Xie, W., & Roberts, J. E. (2020). Face-sensitive brain responses in the first year of life. *NeuroImage*, 211, 116602. <https://doi.org/10.1016/j.neuroimage.2020.116602>
- Deco, G., Jirsa, V. K., & McIntosh, A. R. (2011). Emerging concepts for the dynamical organization of resting-state activity in the brain. *Nature Reviews Neuroscience*, 12(1), 43–56. <https://doi.org/10.1038/nrn2961>
- Deco, G., Ponce-Alvarez, A., Mantini, D., Romani, G. L., Hagmann, P., & Corbetta, M. (2013). Resting-State Functional Connectivity Emerges from Structurally and Dynamically Shaped Slow Linear Fluctuations. *Journal of Neuroscience*, 33(27), 11239–11252. <https://doi.org/10.1523/JNEUROSCI.1091-13.2013>
- Gabard-Durnam, L. J., Wilkinson, C., Kapur, K., Tager-Flusberg, H., Levin, A. R., & Nelson, C. A. (2019). Longitudinal EEG power in the first postnatal year differentiates autism outcomes. *Nature Communications*, 10(1), Article 1. <https://doi.org/10.1038/s41467-019-12202-9>
- Jones, E. J. H., Goodwin, A., Orekhova, E., Charman, T., Dawson, G., Webb, S. J., & Johnson, M. H. (2020). Infant EEG theta modulation predicts childhood intelligence. *Scientific Reports*, 10(1), 11232. <https://doi.org/10.1038/s41598-020-67687-y>

Lachmann, T., Berti, S., Kujala, T., & Schröger, E. (2005). Diagnostic subgroups of developmental dyslexia have different deficits in neural processing of tones and phonemes. *International Journal of Psychophysiology*, 55(2), 105–120.

<https://doi.org/10.1016/j.ijpsycho.2004.11.005>

LeBlanc, J. J., DeGregorio, G., Centofante, E., Vogel-Farley, V. K., Barnes, K., Kaufmann, W. E., Fagiolini, M., & Nelson, C. A. (2015). Visual evoked potentials detect cortical processing deficits in Rett syndrome: VEP in Rett Syndrome. *Annals of Neurology*, 78(5), 775–786.

<https://doi.org/10.1002/ana.24513>

Markant, J., & Scott, L. S. (2018). Attention and Perceptual Learning Interact in the Development of the Other-Race Effect. *Current Directions in Psychological Science*, 27(3), 163–169. <https://doi.org/10.1177/0963721418769884>

Scherf, K. S., & Scott, L. S. (2012). Connecting developmental trajectories: Biases in face processing from infancy to adulthood. *Developmental Psychobiology*, 54(6), 643–663. <https://doi.org/10.1002/dev.21013>

Schulte-Körne, G., Deimel, W., Bartling, J., & Remschmidt, H. (1998). Role of Auditory Temporal Processing for Reading and Spelling Disability. *Perceptual and Motor Skills*, 85(3), 1043–1047. <https://doi.org/10.2466/pms.1998.86.3.1043>

Whedon, M., Perry, N. B., & Bell, M. A. (2020). Relations between frontal EEG maturation and inhibitory control in preschool in the prediction of children's early academic skills. *Brain and Cognition*, 145, 105636.

<https://doi.org/10.1016/j.bandc.2020.105636>

