**Overview of Function\_Maker\_G4**

This set of scripts and GUIs can be used to design analog output functions and position functions to be used in conjunction with displaying patterns on a G4 display. Position functions control what frame of the selected pattern is displayed for every refresh cycle (when the display system is operating in position function mode), operating at a rate of either 500 or 1000 Hz (1-bit or 4-bit patterns, respectively). Analog output functions control the voltage of the analog output channels of the G4 system (accessed easily with the optional breakout box) in a way that is synchronized to the display refresh cycle, operating at 1000 Hz regardless of the pattern refresh rate. Similar to Motion\_Maker\_G4, functions are created using the Function\_Maker\_G4.m script based on input parameters that describe the desired function. These scripts output two types of files: The first type is a .mat file which contains the created function array and all the function parameters so that it can be easily read back into MATLAB. The second type is either a .afn (for analog output functions) or .pfn (for position functions) file containing a binary vector of the function that can be quickly accessed by the Display Controller.

**Creating and visualizing functions in the GUI.** Running ‘Function\_Maker\_G4\_gui’ opens a window where functions can be quickly generated and viewed. Functions are displayed as a time-series plot, with time on the x-axis and either the voltage (for analog output functions) or frame # (for position functions) on the y-axis. Different types of waveforms can be generated in this GUI, such as sawtooth, square waves, or sine waves. A single function can be composed of different sections of waveforms to accommodate different use cases. Many parameters of these waveforms (such as their frequency or intensity) can be adjusted and immediately viewed in the GUI. At the end of this document, each parameter is listed and described for your reference.

**Creating functions using scripts.** The Function\_Maker GUI is recommended for learning about the parameters of functions, testing out new functions, and debugging previously generated functions. For creating many functions at once, it is recommended to use the MATLAB script form of this tool. In addition, while the GUI only supports functions with a maximum of 5 waveform sections, the script version can accommodate any number of sections. The ‘create\_experiment\_G4\_example’ script located in G4\_Display\_Tools\G4\_Example\_Experiment\_Scripts shows one example of creating many functions in a single script, and clicking “create script” in the Function\_Maker\_G4 GUI will generate and open a script in MATLAB based on the current GUI parameters so that you can see how the current function was generated.

**Using functions in an experiment.** After functions have been created, they can be used with the G4 display system by incorporating them into an experiment folder and sending the appropriate commands to the display system. This can be achieved in multiple ways, such as 1) using the script examples located in G4\_Display\_Tools\G4\_Example\_Experiment\_Scripts, 2) using the Protocol Designer located in G4\_Display\_Tools\G4\_Protocol\_Designer, or 3) by using PControl\_G4 located in G4\_Display\_Tools\PControl\_Matlab.

**Description of Parameters**

**Note on units.** The rate of change of each repeating waveform type (e.g. sawtooth, triangle) is controlled by the frequency parameter, in Hz. When creating position functions, it may instead be useful to understand the speed of the moving pattern in terms of degrees-per-second (dps). Use the dps2freq function (which requires the step\_size of the pattern) to convert between these units.

**type**: (string) sets the type of function to create. Options are:

* '**pfn**' – position function (controls what frame is displayed during each display refresh cycle)
* ‘**afn**’ – analog output function (controls voltage of AO channels during each display refresh cycle)

**section**: (1xN cell array of strings) Defines the number of waveform sections in a function, and what type of waveform each section is. Options are:

* '**static**' – holds a single value (set by the ‘val’ parameter) for the duration of the section
* '**sawtooth**' – creates a waveform in a sawtooth pattern, starting at the ‘low’ parameter value and linearly increasing to the ‘high’ parameter value, repeating at a frequency set by the ‘freq’ parameter. (useful for sweeping through all frames of a pattern and repeating)
* ‘**triangle**’ – creates a triangle waveform, starting at ‘low’, linearly increasing to ‘high’, then linear back down to ‘low’
* ‘**sine**’ – creates a sine waveform within the bounds set by ‘low’ and ‘high’
* ‘**cosine**’ – creates cosine waveform
* ‘**square**’ – creates a square wave of 50% duty cycle, starting at ‘high’ and dropping down to ‘low’
* ‘**loom**’ – creates a waveform that models the apparent size of an object approaching at a constant velocity (useful for creating a looming position function for an expansion-contraction pattern)

**dur**: (1xN array of doubles) duration of each section in seconds

**val**: (1xN array of doubles) value for static sections (values aligning with non-static sections will be ignored)

**high**: (1xN array of doubles) high end of waveform range of non-static sections

**low**: (1xN array of doubles) low end of waveform range for non-static sections

**freq**: (1xN array of doubles) frequency of waveform for non-static sections

**size\_speed\_ratio**: (1xN array of doubles) sets the speed of the loom, for loom functions only

**flip**: (1xN array of logicals) sets whether to vertically flip the waveform of each non-static section

*The following parameters are only used for position functions*

**frames**: (integer) specifies the number of frames in the pattern which this position function will be used for. This parameter is optional; it may be useful to set this so that any frame values used in parameters such as ‘val’, ‘high’, or ‘low’ that are set to a value of ‘0’ will be replaced with the last frame in the pattern. In this way, entering a frame value of ‘0’ represents the ‘end’ frame of the pattern.

**gs\_val**: (integer) sets the number of bits for LED brightness intensity in the pattern which this position function will be used for (either 1 or 4). Since the display operates at 500 Hz for 4-bit patterns and 1 kHz for 1-bit patterns, the position function rate has to be consistent with the actual display refresh rate. For analog output functions, the rate will always be 1 kHz.