

## FonaDyn 3.1.2 script syntax

Everything in a FonaDyn script file is case sensitive.

Command	Argument	
//		Anything after a double slash is a comment. Always use <i>forward</i> slashes / in path names.
LOAD	"full/path/to/file.csv"	Load a *_cEGG.csv, *_cPhon.csv, or *_VRP.csv file (include the "quotes" and the file type extension)
LOAD	<str-expr>	As above; <str-expr> is any SC code that evaluates to the pathname of an existing file
HOLD		Pause the reading of the script and wait for the user to press START When the input file has been processed, STOP analysing and continue reading the script
RUN		Pause the reading of the script, and START the analysis When the input file has been processed, STOP analysing and continue reading the script
SAVE	"full/path/to/file_<type>.csv"	Save the NOW map to a *_VRP.csv file (include the "quotes" and the file type extension) If the file name ends in *_S_VRP.csv, the NOW map will be smoothed before it is saved. or, save the cluster centroids to a *_cEGG.csv or *_cPhon.csv file. Include the "quotes" and the file type extension. The <type> ending of the file name controls which data to save.
SAVE	<str-expr>	As above; <str-expr> is any SC code that evaluates to a valid pathname of a file
EVAL	<single-line expression>	<expression> is any SC code that evaluates correctly in the command window (Ctrl-E) when FonaDyn is running. This can be used to get or set global variables (a..z, ~...). Line breaks are not allowed. Use EVAL only if you know what you are doing.

(no space before the .period)

Class	Method	=value of type	Meaning, example	Notes
<b>Input and output</b>				
io	.filePathInput	"string"	The file to analyse: "drive:/full/pathname/to/the/input/file/<*>_Voice_EGG.wav"	(1)
io	.keepInputName	false   true	false: create a time-stamped filename for output files; true: keep the <*> part of the input file name	
io	.keepData	false   true	false: clear the current voice map and the cluster data before starting; true: don't	
io	.enabledWriteLog	false   true	true: while running, save a _Log.aiff file in the Output directory	
io	.writeLogFrameRate	number: 0   50   100   300	0: log on every EGG cycle; otherwise log at one of the rates given, in Hz.	
io	.arrayRecordInputs	[inputVoice , inputEGG , ...]	Specifies from which inputs to record to "<*>_Voice_EGG.wav"	
io	.arrayRecordExtraInputs	[n , ...]	Specifies from which inputs to record to "<*>_Extra.wav"	
io	.rateExtraInputs	1...500	Slow sampling rate for the Extra channels, in Hz. Use only integer divisors of 44100	
io	.enabledEcho	false   true	True if audio should be played back on the speakers	
io	.enabledEGGlisten	false   true	True to play the EGG signal on the second output; and to display it in the signal window	

**Cycle-rate sample entropy estimation**

sampen .amplitudeWindowSize	1...20	
sampen .amplitudeHarmonics	1...20	(2)
sampen .amplitudeSequenceLength	1...20	
sampen .amplitudeTolerance	0.0...10	
sampen .phaseWindowSize	1...20	
sampen .phaseHarmonics	1...20	(2)
sampen .phaseSequenceLength	1...20	
sampen .phaseTolerance	0.0...10	

**The check boxes that turn on time plots**

sampen .bDrawQci	false   true	true: checks the corresponding check box in the Plots panel; false: unchecks
sampen .bDrawDEGGmax	false   true	(same as Qdelta)
sampen .bDrawCPP	false   true	
sampen .bDrawSpecBal	false   true	
sampen .bDrawSampEn	false   true	
sampen .isVisible	false   true	hide/show the Plots panel

**EGG clustering settings**

cluster .nHarmonics	2...20		(3)
cluster .nClusters	2...20		(3)
cluster .initialize	false   true	false: Relearn; true: Pre-learned	(3)
cluster .learn	false   true	false: classify; true: perform clustering	(3)
cluster .reset	false   true	false: disallow resetting the clusters while running	
cluster .autoReset	false   true	true: automatically reset the clustering after first onset of phonation	
cluster .iFramesToReset	integer (default=5)	# of frames (at 24 Hz) to wait after phonation onset before resetting the cluster data	
cluster .suppressGibbs	false   true	true: hide the ripple in resynthesized EGG wave shapes	
cluster .isVisible	false   true	hide/show the EGG clusters panel	

**Phonation type clustering settings**

clusterPhon .nClusters	2...10		(4)
clusterPhon .initialize	false   true	false: Relearn; true: Pre-learned	(4)
clusterPhon .learn	false   true	false: classify; true: perform clustering	(4)
clusterPhon .reset	false   true	false: disallow resetting the clusters while running	
clusterPhon .autoReset	false   true	true: automatically reset the clustering after first onset of phonation	
clusterPhon .iFramesToReset	integer (default=5)	# of frames (at 24 Hz) to wait after phonation onset before resetting the cluster data	
clusterPhon .isVisible	false   true	hide/show the phonation clusters panel	

Other phonation-type settings (metrics, labels, ranges) are set by loading a prepared \_cPhon.csv file

**Sundry**

vrp .clarityThreshold	0.0 ... 1.0	default: 0.96 , maybe reduce to 0.9 or less for running speech or pathological voices
vrp .bHzGrid	false   true	true: plot voice map x axes in Hz (default: MIDI semitones)
vrp .isVisible	false   true	hide/show the voice map(s) panel
vrp .wantsContextSave	false   true	enable auto-saving of context with Save Map
scope .duration	1.0 ... 10.0	seconds on the time axis of plots
scope .normalize	false   true	amplitude-normalize the moving EGG scope and the clustered EGG waveforms
scope .noiseThreshold	0.0 ... 5.0	set a spectral threshold for suppressing wide-band noise in the EGG
scope .isVisible	false   true	hide/show the moving EGG scope
general .output_directory	"/full/path"	where to store recordings and other output files
checkClusterFileMods("/full/path/to/*_VRP.csv")		Checks that the loaded _cEGG.csv and _cPhon.csv files are older than the given map file; warns if not

**NOTES**

- (1) In scripts, only one input signal file at a time can be given - batch mode can not be invoked from here
- (2) This number must not exceed cluster.nHarmonics
- (3) These values are also set when you LOAD a \_cEGG.csv file
- (4) These values are also set when you LOAD a \_cPhon.csv file