



System Console and Toolkit Tcl Command Reference Manual

Updated for Intel® Quartus® Prime Design Suite: **21.2**



[Subscribe](#)

[Send Feedback](#)

MNL-1110 | 2021.06.21

Latest document on the web: [PDF](#) | [HTML](#)

Contents

1. System Console Tcl Command Reference.....	8
1.1. Command Summary.....	8
1.1.1. add_help.....	15
1.1.2. add_service.....	16
1.1.3. autosweep_add_input_parameter.....	16
1.1.4. autosweep_add_output_metric.....	17
1.1.5. autosweep_apply_bestcase.....	17
1.1.6. autosweep_apply_case.....	18
1.1.7. autosweep_create_instance.....	18
1.1.8. autosweep_destroy_instance.....	19
1.1.9. autosweep_get_best_case.....	19
1.1.10. autosweep_get_case_count.....	19
1.1.11. autosweep_get_case_description.....	20
1.1.12. autosweep_get_case_result.....	20
1.1.13. autosweep_get_data.....	21
1.1.14. autosweep_get_input_parameter_range.....	21
1.1.15. autosweep_get_input_parameters.....	22
1.1.16. autosweep_get_instances.....	22
1.1.17. autosweep_get_output_metrics.....	23
1.1.18. autosweep_get_progress.....	23
1.1.19. autosweep_remove_input_paramater.....	23
1.1.20. autosweep_remove_output_metric.....	24
1.1.21. autosweep_set_input_parameter_range.....	25
1.1.22. autosweep_start.....	25
1.1.23. autosweep_stop.....	26
1.1.24. bytestream_receive.....	26
1.1.25. bytestream_send.....	26
1.1.26. claim_service.....	27
1.1.27. close_service.....	28
1.1.28. debug_get_commands.....	28
1.1.29. debug_get_legacy_service_types.....	29
1.1.30. debug_print_filesystem.....	29
1.1.31. design_extract_debug_files.....	30
1.1.32. design_extract_dotty.....	30
1.1.33. design_get_warnings.....	31
1.1.34. design_instantiate.....	31
1.1.35. design_link.....	31
1.1.36. design_load.....	32
1.1.37. design_update_debug_files.....	32
1.1.38. device_download_sof.....	33
1.1.39. device_get_board.....	33
1.1.40. device_get_connections.....	33
1.1.41. device_get_design.....	34
1.1.42. etile_eye_background_scan_done.....	34
1.1.43. etile_eye_cancel_background_scan.....	34
1.1.44. etile_eye_get_attribute.....	35
1.1.45. etile_eye_get_data.....	35

1.1.46. etile_eye_scan_and_load.....	36
1.1.47. etile_eye_unload.....	36
1.1.48. etile_get_actions.....	37
1.1.49. etile_get_parameter.....	37
1.1.50. etile_get_parameters.....	37
1.1.51. etile_read_register.....	38
1.1.52. etile_run_action.....	38
1.1.53. etile_set_parameter.....	39
1.1.54. etile_write_register.....	39
1.1.55. executor_cancel.....	40
1.1.56. executor_clean_directory.....	40
1.1.57. executor_get_directory.....	41
1.1.58. executor_get_environment.....	41
1.1.59. executor_get_exit_value.....	42
1.1.60. executor_get_stderr.....	43
1.1.61. executor_get_stdout.....	43
1.1.62. executor_is_cancelled.....	44
1.1.63. executor_is_done.....	44
1.1.64. executor_run.....	45
1.1.65. executor_set_environment.....	46
1.1.66. executor_unset_environment.....	46
1.1.67. executor_wait_for.....	47
1.1.68. eye_create_instance.....	48
1.1.69. eye_destroy_instance.....	48
1.1.70. eye_get_channel.....	49
1.1.71. eye_get_data.....	49
1.1.72. eye_get_instances.....	49
1.1.73. eye_get_progress.....	50
1.1.74. eye_get_toolkit_instance.....	50
1.1.75. eye_start.....	51
1.1.76. eye_stop.....	51
1.1.77. get_claimed_services.....	52
1.1.78. get_service_paths.....	52
1.1.79. get_service_types.....	53
1.1.80. get_services_to_add.....	53
1.1.81. get_version.....	53
1.1.82. help.....	54
1.1.83. io_bus_access.....	54
1.1.84. io_bus_get_protocol.....	54
1.1.85. is_plugin_enabled.....	55
1.1.86. is_service_open.....	55
1.1.87. is_service_path.....	56
1.1.88. issp_get_instance_info.....	56
1.1.89. issp_read_probe_data.....	56
1.1.90. issp_read_source_data.....	57
1.1.91. issp_write_source_data.....	57
1.1.92. jtag_debug_loop.....	57
1.1.93. jtag_debug_reset_system.....	58
1.1.94. jtag_debug_sample_clock.....	58
1.1.95. jtag_debug_sample_reset.....	59
1.1.96. jtag_debug_sense_clock.....	59

1.1.97. log_command_start.....	59
1.1.98. log_command_stop.....	60
1.1.99. loopback_get.....	60
1.1.100. loopback_set.....	61
1.1.101. loopback_start.....	61
1.1.102. loopback_stop.....	61
1.1.103. marker_get_assignments.....	62
1.1.104. marker_get_info.....	62
1.1.105. marker_get_type.....	63
1.1.106. marker_get_types.....	63
1.1.107. marker_node_info.....	63
1.1.108. master_get_register_names.....	64
1.1.109. master_get_slaves.....	64
1.1.110. master_get_timeout.....	64
1.1.111. master_read_16.....	65
1.1.112. master_read_32.....	65
1.1.113. master_read_8.....	66
1.1.114. master_read_memory.....	66
1.1.115. master_read_to_file.....	66
1.1.116. master_set_timeout.....	67
1.1.117. master_write_16.....	67
1.1.118. master_write_32.....	68
1.1.119. master_write_8.....	68
1.1.120. master_write_from_file.....	69
1.1.121. master_write_memory.....	69
1.1.122. module_get_children.....	70
1.1.123. module_get_keys.....	70
1.1.124. module_get_parent.....	71
1.1.125. module_get_values.....	71
1.1.126. monitor_add_range.....	72
1.1.127. monitor_get_all_read_intervals.....	73
1.1.128. monitor_get_interval.....	73
1.1.129. monitor_get_missing_event_count.....	73
1.1.130. monitor_get_read_interval.....	74
1.1.131. monitor_read_all_data.....	74
1.1.132. monitor_read_data.....	75
1.1.133. monitor_set_callback.....	75
1.1.134. monitor_set_enabled.....	76
1.1.135. monitor_set_interval.....	76
1.1.136. open_service.....	77
1.1.137. packet_send_command.....	77
1.1.138. plugin_disable.....	77
1.1.139. plugin_enable.....	78
1.1.140. processor_clear_breakpoint.....	78
1.1.141. processor_download_elf.....	79
1.1.142. processor_gdbserver.....	79
1.1.143. processor_gdbserver_start.....	80
1.1.144. processor_gdbserver_stop.....	80
1.1.145. processor_get_register.....	81
1.1.146. processor_get_register_names.....	81
1.1.147. processor_in_debug_mode.....	82

1.1.148. processor_reset.....	82
1.1.149. processor_run.....	82
1.1.150. processor_semihosting_start.....	83
1.1.151. processor_semihosting_stop.....	83
1.1.152. processor_set_breakpoint.....	84
1.1.153. processor_set_register.....	84
1.1.154. processor_step.....	85
1.1.155. processor_stop.....	85
1.1.156. processor_stop_reason.....	85
1.1.157. processor_verify_elf.....	86
1.1.158. refresh_connections.....	87
1.1.159. remove_service.....	87
1.1.160. semiConsoleNew.....	87
1.1.161. semihosting_start.....	88
1.1.162. semihosting_stop.....	88
1.1.163. send_message.....	89
1.1.164. sld_access_dr.....	89
1.1.165. sld_access_ir.....	90
1.1.166. sld_lock.....	91
1.1.167. sld_run_test_idle.....	91
1.1.168. sld_send_program.....	91
1.1.169. sld_test_logic_reset.....	92
1.1.170. sld_unlock.....	92
1.1.171. stp_run.....	93
1.1.172. system_get_keys.....	93
1.1.173. system_get_values.....	94
1.1.174. toolkit_get_toolkit_actions.....	95
1.1.175. toolkit_get_toolkit_autosweep_input_parameters.....	95
1.1.176. toolkit_get_toolkit_autosweep_output_metrics.....	96
1.1.177. toolkit_get_toolkit_channel_properties.....	96
1.1.178. toolkit_get_toolkit_channels.....	97
1.1.179. toolkit_get_toolkit_display_hint.....	97
1.1.180. toolkit_get_toolkit_display_id_to_name.....	98
1.1.181. toolkit_get_toolkit_display_item_property.....	98
1.1.182. toolkit_get_toolkit_eye_channels.....	99
1.1.183. toolkit_get_toolkit_eye_input_parameters.....	99
1.1.184. toolkit_get_toolkit_eye_output_metrics.....	100
1.1.185. toolkit_get_toolkit_matching_modules.....	100
1.1.186. toolkit_get_toolkit_matching_services.....	101
1.1.187. toolkit_get_toolkit_matching_systems.....	102
1.1.188. toolkit_get_toolkit_parameter_properties.....	103
1.1.189. toolkit_get_toolkit_parameter_property.....	103
1.1.190. toolkit_get_toolkit_parameter_value.....	104
1.1.191. toolkit_get_toolkit_parameters.....	104
1.1.192. toolkit_get_toolkit_properties.....	105
1.1.193. toolkit_get_toolkit_property.....	105
1.1.194. toolkit_get_toolkit_requirement_ids.....	106
1.1.195. toolkit_get_toolkit_requirement_properties.....	106
1.1.196. toolkit_get_toolkit_requirement_property.....	107
1.1.197. toolkit_load_toolkit_instance.....	108
1.1.198. toolkit_log_toolkit_command_start.....	109

1.1.199. toolkit_log_toolkit_command_stop.....	109
1.1.200. toolkit_run_toolkit_action.....	110
1.1.201. toolkit_set_toolkit_parameter_value.....	110
1.1.202. toolkit_unload_toolkit_instance.....	111
1.1.203. trace_db_delete_snapshot.....	111
1.1.204. trace_db_get_snapshot.....	112
1.1.205. trace_db_snapshot_get_event_data.....	112
1.1.206. trace_db_snapshot_get_event_fields.....	113
1.1.207. trace_db_snapshot_get_event_kind.....	113
1.1.208. trace_db_snapshot_get_event_timestamp.....	113
1.1.209. trace_db_snapshot_get_events.....	114
1.1.210. trace_decoder_add_key_result.....	114
1.1.211. trace_decoder_add_result.....	115
1.1.212. trace_decoder_define_key.....	115
1.1.213. trace_decoder_get_config.....	116
1.1.214. trace_decoder_get_data_16.....	116
1.1.215. trace_decoder_get_data_16be.....	116
1.1.216. trace_decoder_get_data_32.....	117
1.1.217. trace_decoder_get_data_32be.....	117
1.1.218. trace_decoder_get_data_64.....	117
1.1.219. trace_decoder_get_data_64be.....	118
1.1.220. trace_decoder_get_data_8.....	118
1.1.221. trace_decoder_get_length.....	118
1.1.222. trace_decoder_get_timestamp.....	119
1.1.223. trace_decoder_set_callback.....	119
1.1.224. trace_decoder_set_config_regs.....	119
1.1.225. trace_decoder_set_summary.....	120
1.1.226. trace_get_db_size.....	120
1.1.227. trace_get_max_db_size.....	120
1.1.228. trace_get_monitor_info.....	121
1.1.229. trace_get_monitors.....	121
1.1.230. trace_get_status.....	122
1.1.231. trace_load.....	122
1.1.232. trace_read_monitor.....	122
1.1.233. trace_save.....	123
1.1.234. trace_set_max_db_size.....	123
1.1.235. trace_set_trigger_mode.....	124
1.1.236. trace_start.....	124
1.1.237. trace_stop.....	124
1.1.238. trace_write_monitor.....	125
1.2. Properties.....	125
1.2.1. _hw.tcl Capture Mode.....	125
1.2.2. _hw.tcl Command Flag Name.....	126
1.2.3. _hw.tcl Format.....	126
1.2.4. _hw.tcl Kind.....	126
1.2.5. _hw.tcl Protocol.....	126
1.2.6. _hw.tcl Status.....	126
1.2.7. _hw.tcl Trigger Mode.....	126
2. Toolkit Tcl Command References.....	127
2.1. Command Summary.....	127

2.1.1. add_channel.....	128
2.1.2. add_display_item.....	129
2.1.3. add_parameter.....	129
2.1.4. add_requirement.....	130
2.1.5. add_timed_callback.....	131
2.1.6. get_accessible_module.....	132
2.1.7. get_accessible_modules.....	132
2.1.8. get_accessible_service.....	133
2.1.9. get_accessible_services.....	134
2.1.10. get_accessible_system.....	135
2.1.11. get_channel_display_group.....	136
2.1.12. get_channel_property.....	136
2.1.13. get_display_hint.....	137
2.1.14. get_display_item_property.....	138
2.1.15. get_eye_viewer_display_group.....	138
2.1.16. get_parameter_property.....	139
2.1.17. get_parameter_value.....	139
2.1.18. get_toolkit_property.....	140
2.1.19. remove_timed_callback.....	141
2.1.20. send_message.....	141
2.1.21. set_channel_property.....	142
2.1.22. set_current_progress.....	143
2.1.23. set_display_hint.....	143
2.1.24. set_display_item_property.....	144
2.1.25. set_eye_data.....	145
2.1.26. set_eye_property.....	145
2.1.27. set_parameter_property.....	146
2.1.28. set_parameter_update_callback.....	147
2.1.29. set_parameter_value.....	148
2.1.30. set_requirement_property.....	148
2.1.31. set_toolkit_property.....	149
2.1.32. stop_requested.....	150
2.2. Properties.....	150
2.2.1. _hw.tcl Callbacks.....	150
2.2.2. _hw.tcl Channel Properties.....	151
2.2.3. _hw.tcl Channel Type.....	151
2.2.4. _hw.tcl Display Hint.....	151
2.2.5. _hw.tcl Display Item Properties.....	152
2.2.6. _hw.tcl Display Item Type.....	153
2.2.7. _hw.tcl Eye Properties.....	153
2.2.8. _hw.tcl Parameter Properties.....	154
2.2.9. _hw.tcl Parameter Type.....	155
2.2.10. _hw.tcl Requirement Properties.....	155
2.2.11. _hw.tcl Requirement Type.....	156
2.2.12. _hw.tcl Toolkit Properties.....	156
2.3. System Console and Toolkit Tcl Command Reference Manual Revision History.....	157

1. System Console Tcl Command Reference

The System Console commands enable you to control hardware debug and testing using System Console. Use System Console commands to identify a System Console service by its path, to open and close a connection, add a service, and a variety of other System Console controls.

1.1. Command Summary

Returns	Command	Arguments
no return value	<code>add_help</code>	<code><command> <help-text></code>
<i>various</i>	<code>add_service</code>	<code><service-type> <node-name></code>
no return value	<code>autosweep_add_input_parameter</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name> <range></code>
no return value	<code>autosweep_add_output_metric</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name></code>
no return value	<code>autosweep_apply_bestcase</code>	<code><autosweep-inst-id></code>
no return value	<code>autosweep_apply_case</code>	<code><autosweep-inst-id> <case-number></code>
String	<code>autosweep_create_instance</code>	<code>[-id <autosweep-inst-id>]</code>
no return value	<code>autosweep_destroy_instance</code>	<code><autosweep-inst-id></code>
int	<code>autosweep_get_best_case</code>	<code><autosweep-inst-id></code>
int	<code>autosweep_get_case_count</code>	<code><autosweep-inst-id></code>
String	<code>autosweep_get_case_description</code>	<code><autosweep-inst-id> <case-number></code>
String	<code>autosweep_get_case_result</code>	<code><autosweep-inst-id> <toolkit-inst-id> <metric-name> <case-number></code>
String	<code>autosweep_get_data</code>	<code>[-outputfile <filename>] [-format <format>] <autosweep-inst-id></code>
String[]	<code>autosweep_get_input_parameter_range</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name></code>
String[]	<code>autosweep_get_input_parameters</code>	<code><autosweep-inst-id></code>
String[]	<code>autosweep_get_instances</code>	
String[]	<code>autosweep_get_output_metrics</code>	<code><autosweep-inst-id></code>
int	<code>autosweep_get_progress</code>	<code><autosweep-inst-id></code>
no return value	<code>autosweep_remove_input_paramater</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name></code>
no return value	<code>autosweep_remove_output_metric</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name></code>
no return value	<code>autosweep_set_input_parameter_range</code>	<code><autosweep-inst-id> <toolkit-inst-id> <param-name> <range></code>

continued...

Returns	Command	Arguments
no return value	autosweep_start	[-background <background>] <autosweep-inst-id>
no return value	autosweep_stop	<autosweep-inst-id>
String[]	bytestream_receive	<service-path> <length>
no return value	bytestream_send	<service-path> <values>
String	claim_service	<serviceType> <servicePath> <libraryName> [<claims>]
no return value	close_service	<service-type> <service-path>
String[]	debug_get_commands	<flags>
String[]	debug_get_legacy_service_types	
String	debug_print_filesystem	<service-path>
no return value	design_extract_debug_files	<service-path> <String>
no return value	design_extract_dotty	<service-path> <String> [<List>]
String[]	design_get_warnings	<service-path>
String	design_instantiate	<service-path> [<String>]
no return value	design_link	<String> <String>
various	design_load	[-legacy <Boolean>] <String>
no return value	design_update_debug_files	<service-path> <List>
no return value	device_download_sof	<service-path> <String>
String	device_get_board	<service-path>
String[]	device_get_connections	<service-path>
String	device_get_design	<service-path>
int	etile_eye_background_scan_done	<service-path> <channel>
no return value	etile_eye_cancel_background_scan	<service-path> <channel>
String	etile_eye_get_attribute	<service-path> <load_id> <attribute_name>
String[][]	etile_eye_get_data	<service-path> <load_id>
int	etile_eye_scan_and_load	[-timeout <channel>] <service-path>
no return value	etile_eye_unload	<service-path> <load_id>
String[]	etile_get_actions	<service-path>
String	etile_get_parameter	[-timeout <channel>] <service-path> <parameter>
String[]	etile_get_parameters	<service-path>
String	etile_read_register	<service-path> <channel> <offset>
various	etile_run_action	[-timeout <channel>] <service-path> <action>
no return value	etile_set_parameter	[-timeout <channel>] <service-path> <parameter> <value>
no return value	etile_write_register	<service-path> <channel> <offset> <value>
boolean	executor_cancel	<service-path> <int>
no return value	executor_clean_directory	[-delete <Boolean>] <service-path>

continued...

Returns	Command	Arguments
String	executor_get_directory	<service-path>
Map	executor_get_environment	[-inherited_only <Boolean>] <service-path>
int	executor_get_exit_value	<service-path> <int>
String	executor_get_stderr	<service-path> <int>
String	executor_get_stdout	<service-path> <int>
boolean	executor_is_cancelled	<service-path> <int>
boolean	executor_is_done	<service-path> <int>
int	executor_run	[-clean_directory_before <Boolean>] [-clean_directory_after <Boolean>] <service-path> <String> <List>
no return value	executor_set_environment	<service-path> <String> <String>
no return value	executor_unset_environment	<service-path> <String>
boolean	executor_wait_for	<service-path> <int> <int>
String	eye_create_instance	[-id <eye-inst-id>] <toolkit-inst-id> <channel-name>
no return value	eye_destroy_instance	<eye-inst-id>
String	eye_get_channel	<eye-inst-id>
String	eye_get_data	[-outputfile <filename>] <eye-inst-id>
String[]	eye_get_instances	
int	eye_get_progress	<eye-inst-id>
String	eye_get_toolkit_instance	<eye-inst-id>
no return value	eye_start	[-background <background>] <eye-inst-id>
no return value	eye_stop	<eye-inst-id>
String[]	get_claimed_services	<claim-group>
String[]	get_service_paths	[-device <device-path>] [-hpath <hpath-prefix>] [-type <component-type>] <service-type>
Set	get_service_types	
String[]	get_services_to_add	
String[]	get_version	
String	help	[<command>]
String[]	io_bus_access	<service-path> <address> <write> <read_count>
Protocol	io_bus_get_protocol	<service-path>
boolean	is_plugin_enabled	<plugin-path>
boolean	is_service_open	<service-type> <service-path>
Boolean	is_service_path	<service-type> <service-path>
String[]	issp_get_instance_info	<service-path>
String	issp_read_probe_data	<service-path>
String	issp_read_source_data	<service-path>
<i>continued...</i>		

Returns	Command	Arguments
no return value	issp_write_source_data	<service-path> <value>
String[]	jtag_debug_loop	<service-path> <list-of-byte-values>
no return value	jtag_debug_reset_system	<service-path>
int	jtag_debug_sample_clock	<service-path>
int	jtag_debug_sample_reset	<service-path>
boolean	jtag_debug_sense_clock	<service-path>
no return value	log_command_start	[-file <file>] [-append <append>]
no return value	log_command_stop	
String	loopback_get	<service-path> <key>
no return value	loopback_set	<service-path> <key> <value>
no return value	loopback_start	<service-path>
no return value	loopback_stop	<service-path>
Map	marker_get_assignments	<service-path>
Map	marker_get_info	<service-path>
String	marker_get_type	<service-path>
String[]	marker_get_types	<service-path>
Map	marker_node_info	<service-path>
String[]	master_get_register_names	<service-path>
String[]	master_get_slaves	<service-path>
int	master_get_timeout	<service-path>
String[]	master_read_16	<service-path> <address> <size>
String[]	master_read_32	<service-path> <address> <size>
ByteBuffer	master_read_8	<service-path> <address> <size>
various	master_read_memory	<service-path> <address> <size>
no return value	master_read_to_file	<service-path> <filename> <address> <count>
no return value	master_set_timeout	<service-path> <seconds>
no return value	master_write_16	<service-path> <address> <values>
no return value	master_write_32	<service-path> <address> <values>
no return value	master_write_8	<service-path> <address> <values>
no return value	master_write_from_file	<service-path> <filename> <address>
no return value	master_write_memory	<service-path> <address> <values>
String[]	module_get_children	<service-path>
String[]	module_get_keys	<service-path> [<key...>]
String	module_get_parent	<service-path>
String[]	module_get_values	<service-path> <key> [<key...>]
no return value	monitor_add_range	<service-path> <target path> <address> <size>

continued...

Returns	Command	Arguments
String[]	monitor_get_all_read_intervals	<service-path> <target path> <address> <size>
long	monitor_get_interval	<service-path>
int	monitor_get_missing_event_count	<service-path>
long	monitor_get_read_interval	<service-path> <target path> <address> <size>
String[]	monitor_read_all_data	<service-path> <target path> <address> <size>
ByteBuffer	monitor_read_data	<service-path> <target path> <address> <size>
no return value	monitor_set_callback	<service-path> <tcl command>
no return value	monitor_set_enabled	<service-path> <activated>
no return value	monitor_set_interval	<service-path> <interval>
no return value	open_service	<service-type> <service-path>
various	packet_send_command	[-format <format>] [-timeout <command>] <service-path>
no return value	plugin_disable	<plugin-path>
no return value	plugin_enable	<plugin_path>
no return value	processor_clear_breakpoint	<service-path> <String>
no return value	processor_download_elf	[-multiverify <multiverify>] [-noverify <noverify>] [-reset <reset>] <service-path> <elfPath> [<switches>]
int	processor_gdbserver	[-timeout <int>] [-verbosity <int>] [-port <int>] [-semihosting <BOOLEAN_OPTIONAL>] <service-path>
int	processor_gdbserver_start	[-timeout <int>] [-verbosity <int>] [-port <int>] [-semihosting <BOOLEAN_OPTIONAL>] <service-path>
no return value	processor_gdbserver_stop	<service-path>
String	processor_get_register	<service-path> <String>
String[]	processor_get_register_names	<service-path>
boolean	processor_in_debug_mode	<service-path>
no return value	processor_reset	<service-path>
no return value	processor_run	<service-path>
String	processor_semihosting_start	[-file <String>] [-stdout <Boolean>] [-noconsole <Boolean>] [-messages <Boolean>] <service-path>
no return value	processor_semihosting_stop	<service-path>
no return value	processor_set_breakpoint	[-hardware <BOOLEAN_OPTIONAL>] <service-path> <String>
no return value	processor_set_register	<service-path> <String> <int>
no return value	processor_step	<service-path>
no return value	processor_stop	<service-path>
Map	processor_stop_reason	<service-path>
no return value	processor_verify_elf	[-multiverify <Boolean>] [-nodata <BOOLEAN_OPTIONAL>] [-norodata <BOOLEAN_OPTIONAL>] [-norwdata <BOOLEAN_OPTIONAL>] [-notext <BOOLEAN_OPTIONAL>]

continued...

Returns	Command	Arguments
		[-noentry <BOOLEAN_OPTIONAL>] [-noexceptions <BOOLEAN_OPTIONAL>] [-reset <BOOLEAN_OPTIONAL>] <service-path> <String>
no return value	refresh_connections	
no return value	remove_service	<service-type> <service-path>
no return value	semiConsoleNew	[-severAddress <String>] [-name <String>] <int>
no return value	semihosting_start	[-severAddress <String>] [-name <String>] [-file <String>] [-stdout <Boolean>] [-noconsole <Boolean>] [-messages <Boolean>] <int>
no return value	semihosting_stop	<int>
no return value	send_message	<level> <message>
String[]	sld_access_dr	[-write_only <BOOLEAN_OPTIONAL>] <service-path> <length> <delay> <values>
String	sld_access_ir	[-ir <BOOLEAN_OPTIONAL>] [-length <length>] <service-path> <ir-value> <delay>
int	sld_lock	<service-path> <timeout>
no return value	sld_run_test_idle	<service-path> <clocks>
no return value	sld_send_program	<service-path> <streamPath>
no return value	sld_test_logic_reset	<service-path>
int	sld_unlock	<service-path>
no return value	stp_run	<service-path> <signal_set> <trigger> <data_log> <timeout>
String[]	system_get_keys	<service-path> [<key...>]
String[]	system_get_values	<service-path> <key> [<key...>]
String[]	toolkit_get_toolkit_actions	<inst-id> [<channel>]
String[]	toolkit_get_toolkit_autosweep_input_parameters	<inst-id> [<channel>]
String[]	toolkit_get_toolkit_autosweep_output_metrics	<inst-id> [<channel>]
String[]	toolkit_get_toolkit_channel_properties	
String[]	toolkit_get_toolkit_channels	<inst-id>
String	toolkit_get_toolkit_display_hint	<inst-id> <parameter-name> <display-hint>
String[]	toolkit_get_toolkit_display_id_to_name	<inst-id> [<channel>]
various	toolkit_get_toolkit_display_item_property	<inst-id> <display-item-id> <display-item-property>
String[]	toolkit_get_toolkit_eye_channels	<inst-id>
String[]	toolkit_get_toolkit_eye_input_parameters	<inst-id> [<channel>]
String[]	toolkit_get_toolkit_eye_output_metrics	<inst-id> [<channel>]
String[]	toolkit_get_toolkit_matching_modules	[-requirement <requirement-id>] [-recursive <BOOLEAN_OPTIONAL>] [-hpath <hpath>] [-module_name <module-name>] [-version <version>] <service-path>

continued...

Returns	Command	Arguments
String[]	toolkit_get_toolkit_matching_services	<code>[-requirement <requirement-id>] [-recursive <BOOLEAN_OPTIONAL>] [-hpath <hpath>] [-type <type>] [-agent_type_name <agent-type-name>] <service-path></code>
String[]	toolkit_get_toolkit_matching_systems	<code>[-requirement <requirement-id>] [-hpath <hpath>] [-design_id <design-id>] <service-path></code>
String[]	toolkit_get_toolkit_parameter_properties	
<i>various</i>	toolkit_get_toolkit_parameter_property	<code><inst-id> <parameter-name> <parameter-property></code>
String	toolkit_get_toolkit_parameter_value	<code><inst-id> <parameter-name></code>
String[]	toolkit_get_toolkit_parameters	<code><inst-id> [<channel>]</code>
String[]	toolkit_get_toolkit_properties	<code><service-path></code>
String	toolkit_get_toolkit_property	<code><service-path> <property></code>
String[]	toolkit_get_toolkit_requirement_ids	<code><service-path></code>
String[]	toolkit_get_toolkit_requirement_properties	<code><service-path> <requirement-id></code>
String[]	toolkit_get_toolkit_requirement_property	<code><service-path> <requirement-id> <property></code>
String	toolkit_load_toolkit_instance	<code>[-id <inst-id>] [-log <log-flag>] [-output-file-path <output-file-path>] <toolkit-name> <version> <requirements></code>
no return value	toolkit_log_toolkit_command_start	<code><id> [<file> <append>]</code>
no return value	toolkit_log_toolkit_command_stop	<code><id></code>
no return value	toolkit_run_toolkit_action	<code><inst-id> <display-item-id></code>
no return value	toolkit_set_toolkit_parameter_value	<code><inst-id></code>
no return value	toolkit_unload_toolkit_instance	<code><inst-id></code>
no return value	trace_db_delete_snapshot	<code><service-path> <snapshot-id></code>
int	trace_db_get_snapshot	<code>[-start-time <start-time>] [-end-time <end-time>] [-monitor-ids <monitor-ids>] <service-path></code>
String	trace_db_snapshot_get_event_data	<code><service-path> <snapshot-id> <event-id> <field></code>
Set	trace_db_snapshot_get_event_fields	<code><service-path> <snapshot-id> <event-id></code>
Kind	trace_db_snapshot_get_event_kind	<code><service-path> <snapshot-id> <event-id></code>
long	trace_db_snapshot_get_event_timestamp	<code><service-path> <snapshot-id> <event-id></code>
String[]	trace_db_snapshot_get_events	<code><service-path> <snapshot-id></code>
String	trace_decoder_add_key_result	<code><field> <parent> <name> <value></code>
String	trace_decoder_add_result	<code><parent> <name> <value></code>
no return value	trace_decoder_define_key	<code><name> <display_name></code>
int	trace_decoder_get_config	<code><regNum></code>
int	trace_decoder_get_data_16	<code><offset></code>
int	trace_decoder_get_data_16be	<code><offset></code>
int	trace_decoder_get_data_32	<code><offset></code>

continued...

Returns	Command	Arguments
int	trace_decoder_get_data_32be	<offset>
BigInteger	trace_decoder_get_data_64	<offset>
BigInteger	trace_decoder_get_data_64be	<offset>
int	trace_decoder_get_data_8	<offset>
int	trace_decoder_get_length	
BigInteger	trace_decoder_get_timestamp	
no return value	trace_decoder_set_callback	<callback>
no return value	trace_decoder_set_config_regs	<regs>
no return value	trace_decoder_set_summary	<value>
long	trace_get_db_size	<service-path>
long	trace_get_max_db_size	<service-path>
Map	trace_get_monitor_info	<service-path> <String>
String[]	trace_get_monitors	<service-path>
Status	trace_get_status	<service-path>
String	trace_load	<filename>
String	trace_read_monitor	<service-path> <String> <int>
no return value	trace_save	<service-path> <String>
no return value	trace_set_max_db_size	<service-path> <long>
no return value	trace_set_trigger_mode	<service-path> <TriggerMode>
no return value	trace_start	<service-path> <CaptureMode>
no return value	trace_stop	<service-path>
no return value	trace_write_monitor	<service-path> <String> <int> <String>

1.1.1.1. add_help

Description

Adds help text for a given command.

Usage

```
add_help <command> <help-text>
```

ReturnType

Nothing

Returns

no return value

Arguments

command

help-text

1.1.2. add_service

Description

Add a service of the specified service type with the given instance name. Run `get_services_to_add` to retrieve a list of instantiable services. This command returns the path where the service was added. Run `help add_service <service-type>` to get specific help about that service-type, including any parameters that might be required for that service.

Usage

```
add_service <service-type> <node-name>
```

ReturnType

various

Returns

Arguments

service-type

node-name

1.1.3. autosweep_add_input_parameter

Description

Add an input parameter with range for autosweep

Usage

```
autosweep_add_input_parameter <autosweep-inst-id> <toolkit-inst-id>  
<param-name> <range>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

range Range

Related Information

- [autosweep_get_input_parameters](#) on page 22
- [autosweep_remove_input_paramater](#) on page 23

1.1.4. autosweep_add_output_metric

Description

Add an autosweep output metric

Usage

```
autosweep_add_output_metric <autosweep-inst-id> <toolkit-inst-id> <param-name>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

Related Information

- [autosweep_get_output_metrics](#) on page 23
- [autosweep_remove_output_metric](#) on page 24

1.1.5. autosweep_apply_bestcase

Description

Apply the parameter value from best case

Usage

```
autosweep_apply_bestcase <autosweep-inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

1.1.6. autosweep_apply_case

Description

Apply the parameter value from the specified case

Usage

```
autosweep_apply_case <autosweep-inst-id> <case-number>
```

Return Type

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

case-number Case number

1.1.7. autosweep_create_instance

Description

Create a new autosweep instance

Usage

```
autosweep_create_instance [-id <autosweep-inst-id>]
```

Return Type

String

Returns

Autosweep instance ID

Arguments

autosweep-inst-id (optional) ID of the autosweep instance, auto-assigned if not specified

Related Information

[autosweep_destroy_instance](#) on page 19

1.1.8. autosweep_destroy_instance

Description

Remove an autosweep instance

Usage

```
autosweep_destroy_instance <autosweep-inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_create_instance](#) on page 18

1.1.9. autosweep_get_best_case

Description

Get the best case of the most recent autosweep run

Usage

```
autosweep_get_best_case <autosweep-inst-id>
```

ReturnType

int

Returns

Best case

Arguments

autosweep-inst-id Autosweep instance ID

1.1.10. autosweep_get_case_count

Description

Get the total number of autosweep cases

Usage

```
autosweep_get_case_count <autosweep-inst-id>
```

ReturnType

int

Returns

Total number of cases

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_get_progress](#) on page 23

1.1.11. autosweep_get_case_description

Description

Get the description of the parameter set used for a specified case number

Usage

```
autosweep_get_case_description <autosweep-inst-id> <case-number>
```

ReturnType

String

Returns

Description of the parameter set used for a specified case number

Arguments

autosweep-inst-id Autosweep instance ID

case-number Case number

1.1.12. autosweep_get_case_result

Description

Get the result of the specified autosweep case

Usage

```
autosweep_get_case_result <autosweep-inst-id> <toolkit-inst-id> <metric-name> <case-number>
```

ReturnType

String

Returns

Autosweep result

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

metric-name Output metric name

case-number Case number

1.1.13. autosweep_get_data

Description

Get data of the specified autosweep case

Usage

```
autosweep_get_data [-outputfile <filename>] [-format <format>] <autosweep-  
inst-id>
```

ReturnType

String

Returns

Autosweep data

Arguments

filename (optional) Output file name

format (optional) File output format, either XML or HTML

autosweep-inst-id Autosweep instance ID

1.1.14. autosweep_get_input_parameter_range

Description

Get the range of the specified autosweep input parameter

Usage

```
autosweep_get_input_parameter_range <autosweep-inst-id> <toolkit-inst-id>  
<param-name>
```

ReturnType

String[]

Returns

Range

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

Related Information

[autosweep_set_input_parameter_range](#) on page 25

1.1.15. autosweep_get_input_parameters

Description

Retrieve a list of all added autosweep input parameters

Usage

```
autosweep_get_input_parameters <autosweep-inst-id>
```

Return Type

String[]

Returns

A list of all added autosweep input parameters

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_add_input_parameter](#) on page 16

1.1.16. autosweep_get_instances

Description

Retrieve all autosweep instances

Usage

```
autosweep_get_instances
```

Return Type

String[]

Returns

A list of all autosweep instances

Arguments

no arguments

Example

```
autosweep_get_instances
```

Related Information

[autosweep_create_instance](#) on page 18

1.1.17. autosweep_get_output_metrics

Description

Retrieve a list of added autosweep output metrics

Usage

```
autosweep_get_output_metrics <autosweep-inst-id>
```

Return Type

String[]

Returns

A list of added autosweep output metrics

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_add_output_metric](#) on page 17

1.1.18. autosweep_get_progress

Description

Get the autosweep progress in percentage

Usage

```
autosweep_get_progress <autosweep-inst-id>
```

Return Type

int

Returns

Progress in percentage

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_get_case_count](#) on page 19

1.1.19. autosweep_remove_input_paramater

Description

Remove an autosweep input parameter

Usage

```
autosweep_remove_input_paramater <autosweep-inst-id> <toolkit-inst-id>  
<param-name>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

Related Information

[autosweep_add_input_parameter](#) on page 16

1.1.20. autosweep_remove_output_metric

Description

Remove an autosweep output metric

Usage

```
autosweep_remove_output_metric <autosweep-inst-id> <toolkit-inst-id>  
<param-name>
```

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

Related Information

[autosweep_add_output_metric](#) on page 17

1.1.21. autosweep_set_input_parameter_range

Description

Set the range of the specified autosweep input parameter

Usage

```
autosweep_set_input_parameter_range <autosweep-inst-id> <toolkit-inst-id>  
<param-name> <range>
```

Return Type

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

toolkit-inst-id Toolkit instance ID

param-name Parameter name

range Range

Related Information

[autosweep_get_input_parameter_range](#) on page 21

1.1.22. autosweep_start

Description

Start autosweep

Usage

```
autosweep_start [-background <background>] <autosweep-inst-id>
```

Return Type

Nothing

Returns

no return value

Arguments

background (optional) TRUE to run in the background, else FALSE. Default is FALSE

autosweep-inst-id Autosweep instance ID

Related Information

- [autosweep_get_progress](#) on page 23
- [autosweep_stop](#) on page 26

1.1.23. autosweep_stop**Description**

Stop autosweep

Usage

`autosweep_stop <autosweep-inst-id>`

ReturnType

Nothing

Returns

no return value

Arguments

autosweep-inst-id Autosweep instance ID

Related Information

[autosweep_start](#) on page 25

1.1.24. bytestream_receive**Description**

null

Usage

`bytestream_receive <service-path> <length>`

ReturnType

String[]

Returns**Arguments**

service-path

length

1.1.25. bytestream_send**Description**

null

Usage

```
bytestream_send <service-path> <values>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

values

1.1.26. claim_service**Description**

Claim service provides finer grain control of which portion of a service the user wants to use. This is an enhanced mode of `open_service`. The claim-group argument is required; library authors are advised to create their unique claim-group string to keep track of their claimed services, and to prevent other users from accessing their claimed services. For default scripting use, "" is an acceptable default for the claim-group argument. The claims argument is optional, and is used to specify what specific portion of a service the user wants to use. If that specific portion cannot be safely claimed - for example, if another user has claimed that portion for exclusive access - the `claim_service` command will fail. When specifying claims, the claim strings must be entered as a list of lists. Please see the examples below. Leaving the claims argument empty means the user wants to claim the entire channel for exclusive use. This is equivalent to using `open_service` to access the service. This call returns a new service-path where the claimed service will be placed. Use this new service-path to access the claimed service.

Usage

```
claim_service <serviceType> <servicePath> <libraryName> [<claims>]
```

ReturnType

String

Returns**Arguments**

serviceType

servicePath

libraryName

claims (optional)

Example

```
To claim a service for exclusive use with "" as the claim-group name, run:
    claim_service <service-type> <service-path> ""
To claim a service with one claim, run:
    claim_service <service-type> <service-path> claimGroup1 {{<claim
contents>}} , or
    claim_service <service-type> <service-path> claimGroup1 [list [list
<claim contents> ] ]
To claim a service with two claims, run:
    claim_service <service-type> <service-path> claimGroup1 {{<claim1
contents>} {<claim2 contents>}} , or
    claim_service <service-type> <service-path> claimGroup1 [list [list
<claim1 contents> ] [list <claim2 contents>]]
```

1.1.27. close_service

Description

Closes the specified service type at the specified path.

Usage

```
close_service <service-type> <service-path>
```

Return Type

Nothing

Returns

no return value

Arguments

service-type

service-path

1.1.28. debug_get_commands

Description

null

Altera Internal

This is for internal Altera use only.

Usage

```
debug_get_commands <flags>
```

Return Type

String[]

Returns

Arguments

flags See [Command Flag Name](#).

1.1.29. debug_get_legacy_service_types

Description

null

Altera Internal

This is for internal Altera use only.

Usage

```
debug_get_legacy_service_types
```

ReturnType

String[]

Returns

Arguments

no arguments

1.1.30. debug_print_filesystem

Description

Debug command to print a text representation of the System Console Virtual Filesystem. An optional path can be supplied - which will only print the subtree from that path and down. Not giving a path it will print the entire filesystem. 'o' is a node '+' are interfaces registered on the node '->' is the nodes parent resolver.

Altera Internal

This is for internal Altera use only.

Usage

```
debug_print_filesystem <service-path>
```

ReturnType

String

Returns

Arguments

service-path

1.1.31. design_extract_debug_files

Description

null

Usage

design_extract_debug_files <service-path> <String>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

1.1.32. design_extract_dotty

Description

null

Altera Internal

This is for internal Altera use only.

Usage

design_extract_dotty <service-path> <String> [<List>]

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

List (optional)

1.1.33. design_get_warnings

Description

null

Usage

design_get_warnings <service-path>

ReturnType

String[]

Returns

Arguments

service-path

1.1.34. design_instantiate

Description

null

Usage

design_instantiate <service-path> [<String>]

ReturnType

String

Returns

Arguments

service-path

String (optional)

1.1.35. design_link

Description

null

Usage

design_link <String> <String>

ReturnType

Nothing

Returns

no return value

Arguments

String

String

1.1.36. design_load**Description**

null

Usage

`design_load [-legacy <Boolean>] <String>`

ReturnType

various

Returns**Arguments**

Boolean (optional)

String

1.1.37. design_update_debug_files**Description**

null

Altera Internal

This is for internal Altera use only.

Usage

`design_update_debug_files <service-path> <List>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

List

1.1.38. device_download_sof

Description

null

Usage

device_download_sof <service-path> <String>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

1.1.39. device_get_board

Description

null

Usage

device_get_board <service-path>

ReturnType

String

Returns

Arguments

service-path

1.1.40. device_get_connections

Description

null

Usage

device_get_connections <service-path>

ReturnType

String[]

Returns**Arguments**

service-path

1.1.41. device_get_design**Description**

null

Usage

device_get_design <*service-path*>

ReturnType

String

Returns**Arguments**

service-path

1.1.42. etile_eye_background_scan_done**Description**

null

Usage

etile_eye_background_scan_done <*service-path*> <*channel*>

ReturnType

int

Returns**Arguments**

service-path

channel

1.1.43. etile_eye_cancel_background_scan**Description**

null

Usage

etile_eye_cancel_background_scan <*service-path*> <*channel*>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

channel

1.1.44. etile_eye_get_attribute

Description

null

Usage

`etile_eye_get_attribute <service-path> <load_id> <attribute_name>`

ReturnType

String

Returns

Arguments

service-path

load_id

attribute_name

1.1.45. etile_eye_get_data

Description

null

Usage

`etile_eye_get_data <service-path> <load_id>`

ReturnType

String[][]

Returns**Arguments**

service-path

load_id

1.1.46. etile_eye_scan_and_load**Description**

null

Usage

`etile_eye_scan_and_load [-timeout <channel>] <service-path>`

ReturnType

int

Returns**Arguments**

channel (optional)

service-path

1.1.47. etile_eye_unload**Description**

null

Usage

`etile_eye_unload <service-path> <load_id>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

load_id

1.1.48. etile_get_actions

Description

null

Altera Internal

This is for internal Altera use only.

Usage

```
etile_get_actions <service-path>
```

ReturnType

String[]

Returns

Arguments

service-path

1.1.49. etile_get_parameter

Description

null

Usage

```
etile_get_parameter [-timeout <channel>] <service-path> <parameter>
```

ReturnType

String

Returns

Arguments

channel (optional)

service-path

parameter

1.1.50. etile_get_parameters

Description

null

Altera Internal

This is for internal Altera use only.

Usage

`etile_get_parameters <service-path>`

ReturnType

String[]

Returns**Arguments**

service-path

1.1.51. etile_read_register

Description

null

Usage

`etile_read_register <service-path> <channel> <offset>`

ReturnType

String

Returns**Arguments**

service-path

channel

offset

1.1.52. etile_run_action

Description

null

Usage

`etile_run_action [-timeout <channel>] <service-path> <action>`

ReturnType

various

Returns**Arguments**

channel (optional)

service-path

action

1.1.53. etile_set_parameter

Description

null

Usage

```
etile_set_parameter [-timeout <channel>] <service-path> <parameter>  
<value>
```

ReturnType

Nothing

Returns

no return value

Arguments

channel (optional)

service-path

parameter

value

1.1.54. etile_write_register

Description

null

Usage

```
etile_write_register <service-path> <channel> <offset> <value>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

channel

offset

value

1.1.55. `executor_cancel`

Description

Attempts to cancel process execution. This attempt will fail if the process has already completed, has already been cancelled, or could not be cancelled for some other reason. If successful, and this process not started when cancel is called, the process should never run.

Usage

```
executor_cancel <service-path> <int>
```

ReturnType

boolean

Returns

False if the process could not be cancelled, typically because it has already completed normally; true otherwise

Arguments

service-path

int

Example

```
set cancelled [executor_cancel $executor_path $handle]
```

Related Information

- [executor_is_cancelled](#) on page 44
- [executor_run](#) on page 45

1.1.56. `executor_clean_directory`

Description

Clean the working directory by recursively deleting all files and folders, will take some time if the directory is huge

Usage

```
executor_clean_directory [-delete <Boolean>] <service-path>
```


ReturnType

Nothing

Returns

no return value

Arguments

Boolean (optional)

service-path

Example

```
executor_clean_directory $executor_path
```

1.1.57. executor_get_directory

Description

Return the working directory allocated to the executor when the service was created through "add_service"

Usage

```
executor_get_directory <service-path>
```

ReturnType

String

Returns

Working directory of executor service

Arguments

service-path

Example

```
executor_get_directory $executor_path
```

Related Information

[add_service](#) on page 16

1.1.58. executor_get_environment

Description

Return all environment variables in Tcl array

Usage

```
executor_get_environment [-inherited_only <Boolean>] <service-path>
```

ReturnType

Map

Returns

All environment variables

Arguments

Boolean (optional)

service-path

Example

```
array set env_var [executor_get_environment $executor_path]
```

Related Information

- [executor_set_environment](#) on page 46
- [executor_unset_environment](#) on page 46

1.1.59. executor_get_exit_value

Description

Return the exit value of the process. Prerequisite: Must call `executor_is_done` or `executor_wait_for` and wait until completion before retrieving exit value, except if the process was cancelled.

Usage

```
executor_get_exit_value <service-path> <int>
```

ReturnType

int

Returns

Exit value of the process, -1 if process has been cancelled. By convention, value 0 indicates normal termination.

Arguments

service-path

int

Example

```
executor_get_exit_value $executor_path $handle
```

Related Information

- [executor_is_done](#) on page 44

- [executor_wait_for](#) on page 47

1.1.60. executor_get_stderr

Description

Return the stderr text produced by the process

Usage

```
executor_get_stderr <service-path> <int>
```

ReturnType

String

Returns

Stderr text produced by the process

Arguments

service-path

int

Example

```
executor_get_stderr $executor_path $handle
```

Related Information

- [executor_is_done](#) on page 44
- [executor_wait_for](#) on page 47

1.1.61. executor_get_stdout

Description

Return the stdout text produced by the process

Usage

```
executor_get_stdout <service-path> <int>
```

ReturnType

String

Returns

Stdout text produced by the process

Arguments

service-path

int

Example

```
executor_get_stdout $executor_path $handle
```

Related Information

- [executor_is_done](#) on page 44
- [executor_wait_for](#) on page 47

1.1.62. executor_is_cancelled**Description**

Query if the process was cancelled before it completed normally

Usage

```
executor_is_cancelled <service-path> <int>
```

Return Type

boolean

Returns

True if process cancelled, false otherwise

Arguments

service-path

int

Example

```
if {[executor_is_cancelled $executor_path $handle]} { # do something }
```

Related Information

[executor_cancel](#) on page 40

1.1.63. executor_is_done**Description**

Query if the process completed. Completion may be due to normal termination, cancellation or failure to start, in all of these cases, this will return true

Usage

```
executor_is_done <service-path> <int>
```

Return Type

boolean

Returns

True if the process completed, false otherwise

Arguments

service-path

int

Example

```
if {[executor_is_done $executor_path $handle]} { # do something }
```

Related Information

- [executor_run](#) on page 45
- [executor_wait_for](#) on page 47

1.1.64. executor_run

Description

Schedule the process to run some time in the future in this single-threaded executor. Previous processes have to finish execution before this process is started by executor.

Usage

```
executor_run [-clean_directory_before <Boolean>] [-clean_directory_after <Boolean>] <service-path> <String> <List>
```

ReturnType

int

Returns

Process handle

Arguments

Boolean (optional)

Boolean (optional)

service-path

String

List

Example

```
set handle [executor_run $executor_path $handle "quartus_sh" "--version"]
```

Related Information

- [executor_cancel](#) on page 40

- [executor_is_done](#) on page 44
- [executor_wait_for](#) on page 47

1.1.65. executor_set_environment

Description

Set environment variable

Usage

```
executor_set_environment <service-path> <String> <String>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

String

Example

```
executor_set_environment $executor_path "MY_VARIABLE" "my_value"
```

Related Information

- [executor_get_environment](#) on page 41
- [executor_unset_environment](#) on page 46

1.1.66. executor_unset_environment

Description

Unset environment variable

Usage

```
executor_unset_environment <service-path> <String>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

Example

```
executor_set_environment $executor_path "MY_VARIABLE"
```

Related Information

- [executor_get_environment](#) on page 41
- [executor_set_environment](#) on page 46

1.1.67. executor_wait_for

Description

Blocks until process completed or timeout occurred, whichever comes first. Timeout cannot be more than 10 seconds. To wait longer, run the command multiple times.

Usage

```
executor_wait_for <service-path> <int> <int>
```

ReturnType

boolean

Returns

True if the process completed before timeout, false otherwise

Arguments

service-path

int

int

Example

```
executor_wait_for $executor_path $handle 1
```

Related Information

- [executor_is_done](#) on page 44
- [executor_run](#) on page 45

1.1.68. eye_create_instance

Description

Create a new eye instance

Usage

```
eye_create_instance [-id <eye-inst-id>] <toolkit-inst-id> <channel-name>
```

ReturnType

String

Returns

Eye instance ID

Arguments

eye-inst-id (optional) Eye instance ID

toolkit-inst-id Toolkit instance ID

channel-name Channel name

Related Information

[eye_destroy_instance](#) on page 48

1.1.69. eye_destroy_instance

Description

Remove an eye instance

Usage

```
eye_destroy_instance <eye-inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

eye-inst-id Eye instance ID

Related Information

[eye_create_instance](#) on page 48

1.1.70. eye_get_channel

Description

Get the channel associated with the eye instance

Usage

```
eye_get_channel <eye-inst-id>
```

ReturnType

String

Returns

Channel name

Arguments

eye-inst-id Eye instance ID

1.1.71. eye_get_data

Description

Get data from eye scan

Usage

```
eye_get_data [-outputfile <filename>] <eye-inst-id>
```

ReturnType

String

Returns

Eye scan data in XML if no output file is specified

Arguments

filename (optional) Output file name

eye-inst-id Eye instance ID

1.1.72. eye_get_instances

Description

Retrieve all eye instances

Usage

```
eye_get_instances
```

ReturnType

String[]

Returns

A list of all eye instances

Arguments

no arguments

Example

```
eye_get_instances
```

Related Information

[eye_create_instance](#) on page 48

1.1.73. eye_get_progress

Description

Get the eye scanning progress in percentage

Usage

```
eye_get_progress <eye-inst-id>
```

Return Type

int

Returns

Progress in percentage

Arguments

eye-inst-id Eye instance ID

1.1.74. eye_get_toolkit_instance

Description

Get toolkit instance associated with the specified eye instance

Usage

```
eye_get_toolkit_instance <eye-inst-id>
```

Return Type

String

Returns

Toolkit instance ID

Arguments

eye-inst-id Eye instance ID

1.1.75. eye_start

Description

Start scanning eye

Usage

```
eye_start [-background <background>] <eye-inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

background (optional) TRUE to run in the background, else FALSE. Default is FALSE

eye-inst-id Eye instance ID

Related Information

[eye_stop](#) on page 51

1.1.76. eye_stop

Description

Stop scanning eye

Usage

```
eye_stop <eye-inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

eye-inst-id Eye instance ID

Related Information

[eye_start](#) on page 51

1.1.77. get_claimed_services

Description

For the given claim-group, returns a list services claimed via claim_service with that claim-group name. The returned list will consists of pairs of paths and service types, each pair denoting one claimed service.

Usage

```
get_claimed_services <claim-group>
```

ReturnType

String[]

Returns

Arguments

claim-group

1.1.78. get_service_paths

Description

Returns a list of paths to nodes that implement the requested service type. Service paths can be filtered based on a number of different optional parameters.

Usage

```
get_service_paths [-device <device-path>] [-hpath <hpath-prefix>] [-type  
<component-type>] <service-type>
```

ReturnType

String[]

Returns

Arguments

device-path (optional)

hpath-prefix (optional)

component-type (optional)

service-type

Example

```
To list all paths to nodes that implement a supported service type:  
    get_service_paths <service-type>  
To list all paths to nodes that implement the requested service type under a  
specific hpath:  
    get_service_paths -hpath <hpath-prefix> <service-type>
```

```
To list all paths to nodes attached to a specific component type that implement the requested service:
```

```
get_service_paths -type <component-type> <service-type>
```

```
To list all paths to nodes on a specific device that implement the requested service type:
```

```
get_service_paths -device <device-path> <service-type>
```

1.1.79. get_service_types

Description

Returns a list of all services that the console manages.

Usage

```
get_service_types
```

ReturnType

Set

Returns

Arguments

no arguments

1.1.80. get_services_to_add

Description

Returns a list of all services that are instantiable with the add_service command.

Usage

```
get_services_to_add
```

ReturnType

String[]

Returns

Arguments

no arguments

1.1.81. get_version

Description

Returns the current System Console version and build number.

Usage

```
get_version
```

ReturnType

String[]

Returns**Arguments**

no arguments

1.1.82. help**Description**

Get help on any of the following commands: %s by typing help <command name>

Usage

help [<command>]

ReturnType

String

Returns**Arguments**

command (optional)

1.1.83. io_bus_access**Description**

null

Usage

io_bus_access <service-path> <address> <write> <read_count>

ReturnType

String[]

Returns**Arguments**

service-path

address

write

read_count

1.1.84. io_bus_get_protocol**Description**

null

Usage

`io_bus_get_protocol <service-path>`

ReturnType

Protocol

Returns

See Protocol.

Arguments

service-path

1.1.85. is_plugin_enabled

Description

Returns non-zero when the plugin at the specified path is enabled.

Usage

`is_plugin_enabled <plugin-path>`

ReturnType

boolean

Returns

Arguments

plugin-path

1.1.86. is_service_open

Description

Returns non-zero when the service of the specified type at the specified path is open.

Usage

`is_service_open <service-type> <service-path>`

ReturnType

boolean

Returns

Arguments

service-type

service-path

1.1.87. is_service_path

Description

Returns non-zero when the specified path belongs to the specified service type.

Usage

```
is_service_path <service-type> <service-path>
```

ReturnType

Boolean

Returns**Arguments**

service-type

service-path

1.1.88. issp_get_instance_info

Description

null

Usage

```
issp_get_instance_info <service-path>
```

ReturnType

String[]

Returns**Arguments**

service-path

1.1.89. issp_read_probe_data

Description

null

Usage

```
issp_read_probe_data <service-path>
```

ReturnType

String

Returns

Arguments

service-path

1.1.90. `issp_read_source_data`

Description

null

Usage

`issp_read_source_data <service-path>`

ReturnType

String

Returns

Arguments

service-path

1.1.91. `issp_write_source_data`

Description

null

Usage

`issp_write_source_data <service-path> <value>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

value

1.1.92. `jtag_debug_loop`

Description

Loops the specified list of byte values through a loopback at tdi and tdo of a system level debug (SLD) node. Returns the values in the order that they arrived through the loop. Blocks until all bytes are received.

Usage

```
jtag_debug_loop <service-path> <list-of-byte-values>
```

ReturnType

String[]

Returns**Arguments**

service-path

list-of-byte-values

1.1.93. jtag_debug_reset_system

Description

Issues a reset request to the system.

Usage

```
jtag_debug_reset_system <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.94. jtag_debug_sample_clock

Description

Returns an instantaneous sample of the clock signal that drives the component's system interface.

Usage

```
jtag_debug_sample_clock <service-path>
```

ReturnType

int

Returns**Arguments**

service-path

1.1.95. jtag_debug_sample_reset

Description

Returns an instantaneous sample of the reset signal that drives the component's system interface.

Usage

```
jtag_debug_sample_reset <service-path>
```

ReturnType

int

Returns

Arguments

service-path

1.1.96. jtag_debug_sense_clock

Description

Returns the value of a sticky bit that monitors system clock activity. The sticky bit is reset (to 0) on read. 1 indicates clock is active. 0 indicates no clock activity.

Usage

```
jtag_debug_sense_clock <service-path>
```

ReturnType

boolean

Returns

Arguments

service-path

1.1.97. log_command_start

Description

Starts command logging. This command must be enabled with the command line argument "enable_command_logging". These logged commands can be displayed in the messages window of system console or can be captured in a file. The append flag can be used to append logged commands to the end of a file.

Usage

```
log_command_start [-file <file>] [-append <append>]
```

ReturnType

Nothing

Returns

no return value

Arguments

file (optional) Captured commands file

append (optional) Append flag

Example

```
log_command_start "logged_commands_file.txt" append
```

1.1.98. log_command_stop

Description

Stops logging commands.

Usage

```
log_command_stop
```

ReturnType

Nothing

Returns

no return value

Arguments

no arguments

Example

```
log_command_stop
```

1.1.99. loopback_get

Description

null

Usage

```
loopback_get <service-path> <key>
```

ReturnType

String

Returns**Arguments**

service-path

key

1.1.100. loopback_set

Description

null

Usage

loopback_set <service-path> <key> <value>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

key

value

1.1.101. loopback_start

Description

null

Usage

loopback_start <service-path>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.102. loopback_stop

Description

null

Usage

loopback_stop <service-path>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.103. marker_get_assignments**Description**

null

Usage

marker_get_assignments <service-path>

ReturnType

Map

Returns**Arguments**

service-path

1.1.104. marker_get_info**Description**

null

Usage

marker_get_info <service-path>

ReturnType

Map

Returns**Arguments**

service-path

1.1.105. marker_get_type

Description

null

Usage

marker_get_type <service-path>

ReturnType

String

Returns

Arguments

service-path

1.1.106. marker_get_types

Description

null

Usage

marker_get_types <service-path>

ReturnType

String[]

Returns

Arguments

service-path

1.1.107. marker_node_info

Description

null

Usage

marker_node_info <service-path>

ReturnType

Map

Returns

Arguments

service-path

1.1.108. master_get_register_names

Description

null

Usage

```
master_get_register_names <service-path>
```

ReturnType

String[]

Returns**Arguments**

service-path

1.1.109. master_get_slaves

Description

null

Usage

```
master_get_slaves <service-path>
```

ReturnType

String[]

Returns**Arguments**

service-path

1.1.110. master_get_timeout

Description

null

Altera Internal

This is for internal Altera use only.

Usage

```
master_get_timeout <service-path>
```

ReturnType

int

Returns

Arguments

service-path

1.1.111. master_read_16

Description

null

Usage

master_read_16 <*service-path*> <*address*> <*size*>

ReturnType

String[]

Returns

Arguments

service-path

address

size

1.1.112. master_read_32

Description

null

Usage

master_read_32 <*service-path*> <*address*> <*size*>

ReturnType

String[]

Returns

Arguments

service-path

address

size

1.1.113. master_read_8

Description

null

Usage

```
master_read_8 <service-path> <address> <size>
```

ReturnType

ByteBuffer

Returns**Arguments**

service-path

address

size

1.1.114. master_read_memory

Description

null

Usage

```
master_read_memory <service-path> <address> <size>
```

ReturnType

various

Returns**Arguments**

service-path

address

size

1.1.115. master_read_to_file

Description

null

Usage

`master_read_to_file <service-path> <filename> <address> <count>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

filename

address

count

1.1.116. master_set_timeout

Description

null

Altera Internal

This is for internal Altera use only.

Usage

`master_set_timeout <service-path> <seconds>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

seconds

1.1.117. master_write_16

Description

null

Usage

master_write_16 <service-path> <address> <values>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

address

values

1.1.118. master_write_32**Description**

null

Usage

master_write_32 <service-path> <address> <values>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

address

values

1.1.119. master_write_8**Description**

null

Usage

master_write_8 <service-path> <address> <values>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

address

values

1.1.120. master_write_from_file

Description

null

Usage

master_write_from_file <*service-path*> <*filename*> <*address*>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

filename

address

1.1.121. master_write_memory

Description

null

Usage

master_write_memory <*service-path*> <*address*> <*values*>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

address

values

1.1.122. module_get_children

Description

Retrieve a list of child module paths of the specified module

Usage

```
module_get_children <service-path>
```

ReturnType

String[]

Returns

A list children module paths of the specified module

Arguments

service-path Module service path (from "get_service_paths module")

Example

```
module_get_children /designs/top.sof/my_ip.sopcinfo/foo
```

1.1.123. module_get_keys

Description

Retrieve the keys available to query the module's SOPCINFO file content, for use in the command `module_get_values`. Keys are mapped to either the attribute or sub-element of the corresponding `<module>` in SOPCINFO file

Usage

```
module_get_keys <service-path> [<key...>]
```

ReturnType

String[]

Returns

A list of keys for the specified module path

Arguments

service-path Module service path (from "get_service_paths module")

key... (optional) Key(s) mapped to attribute(s) or sub-element(s)

Example

```
module_get_keys /designs/top.sof/my_ip.sopcinfo/foo
```

1.1.124. module_get_parent

Description

Retrieve parent module path of the specified module

Usage

```
module_get_parent <service-path>
```

Return Type

String

Returns

Parent module path of the specified module

Arguments

service-path Module service path (from "get_service_paths module")

Example

```
module_get_parent /designs/top.sof/my_ip.sopcinfo/foo/bar
```

1.1.125. module_get_values

Description

Retrieve a list of children module paths of the specified module

Usage

```
module_get_values <service-path> <key> [<key...>]
```

Return Type

String[]

Returns

A list of values for the specified key of module path

Arguments

service-path Module service path (from "get_service_paths module")

key Key mapped to attribute or sub-element

key... (optional) Key(s) mapped to attribute(s) or sub-element(s)

Example

```
Given a SOPCINFO file with the following <module> element:
<module>
  <parameter name="param_a">
    <type>java.lang.String</type>
    <value>value_a</value>
    <derived>false</derived>
    <enabled>true</enabled>
    <visible>true</visible>
    <valid>true</valid>
  </parameter>
  <parameter name="param_b">
    <type>java.lang.String</type>
    <value>value_b</value>
    <derived>false</derived>
    <enabled>true</enabled>
    <visible>true</visible>
    <valid>true</valid>
  </parameter>
  ...
</module>
Command:
  module_get_values /designs/top.sof/my_ip.sopcinfo/foo parameter
Result:
  param_a param_b
Command:
  module_get_values /designs/top.sof/my_ip.sopcinfo/foo parameter:param_a
Result:
  value_a
```

1.1.126. monitor_add_range

Description

null

Usage

`monitor_add_range <service-path> <target path> <address> <size>`

Return Type

Nothing

Returns

no return value

Arguments

service-path

target path

address

size

1.1.127. monitor_get_all_read_intervals

Description

null

Usage

```
monitor_get_all_read_intervals <service-path> <target path> <address>  
<size>
```

ReturnType

String[]

Returns

Arguments

service-path

target path

address

size

1.1.128. monitor_get_interval

Description

null

Usage

```
monitor_get_interval <service-path>
```

ReturnType

long

Returns

Arguments

service-path

1.1.129. monitor_get_missing_event_count

Description

null

Usage

monitor_get_missing_event_count <service-path>

ReturnType

int

Returns**Arguments**

service-path

1.1.130. monitor_get_read_interval

Description

null

Usage

monitor_get_read_interval <service-path> <target path> <address> <size>

ReturnType

long

Returns**Arguments**

service-path

target path

address

size

1.1.131. monitor_read_all_data

Description

null

Usage

monitor_read_all_data <service-path> <target path> <address> <size>

ReturnType

String[]

Returns

Arguments

service-path

target path

address

size

1.1.132. monitor_read_data

Description

null

Usage

monitor_read_data <*service-path*> <*target path*> <*address*> <*size*>

Return Type

ByteBuffer

Returns

Arguments

service-path

target path

address

size

1.1.133. monitor_set_callback

Description

null

Usage

monitor_set_callback <*service-path*> <*tcl command*>

Return Type

Nothing

Returns

no return value

Arguments

service-path

tcl command

1.1.134. monitor_set_enabled**Description**

null

Usage

`monitor_set_enabled <service-path> <activated>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

activated

1.1.135. monitor_set_interval**Description**

null

Usage

`monitor_set_interval <service-path> <interval>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

interval

1.1.136. open_service

Description

Opens the specified service type at the specified path. An open_service call is equivalent to a claim_service call with no claims specified.

Usage

```
open_service <service-type> <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-type

service-path

1.1.137. packet_send_command

Description

null

Usage

```
packet_send_command [-format <format>] [-timeout <command>] <service-path>
```

ReturnType

various

Returns

Arguments

format (optional) See [Format](#).

command (optional)

service-path

1.1.138. plugin_disable

Description

Disables the plugin at the given path.

Usage

`plugin_disable <plugin-path>`

ReturnType

Nothing

Returns

no return value

Arguments

plugin-path

1.1.139. plugin_enable**Description**

Enables the plugin at the given path.

Usage

`plugin_enable <plugin_path>`

ReturnType

Nothing

Returns

no return value

Arguments

plugin_path

1.1.140. processor_clear_breakpoint**Description**

null

Usage

`processor_clear_breakpoint <service-path> <String>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

1.1.141. processor_download_elf

Description

null

Usage

```
processor_download_elf [-multiverify <multiverify>] [-noverify <noverify>] [-  
reset <reset>] <service-path> <elfPath> [<switches>]
```

ReturnType

Nothing

Returns

no return value

Arguments

multiverify (optional)

noverify (optional)

reset (optional)

service-path

elfPath

switches (optional)

1.1.142. processor_gdbserver

Description

null

Usage

```
processor_gdbserver [-timeout <int>] [-verbosity <int>] [-port <int>] [-  
semihosting <BOOLEAN_OPTIONAL>] <service-path>
```

ReturnType

int

Returns**Arguments**

int (optional)

int (optional)

int (optional)

BOOLEAN_OPTIONAL (optional)

service-path

1.1.143. processor_gdbserver_start**Description**

null

Usage

```
processor_gdbserver_start [-timeout <int>] [-verbosity <int>] [-port <int>] [-semihosting <BOOLEAN_OPTIONAL>] <service-path>
```

Return Type

int

Returns**Arguments**

int (optional)

int (optional)

int (optional)

BOOLEAN_OPTIONAL (optional)

service-path

1.1.144. processor_gdbserver_stop**Description**

null

Usage

`processor_gdbserver_stop <service-path>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.145. processor_get_register

Description

null

Usage

`processor_get_register <service-path> <String>`

ReturnType

String

Returns

Arguments

service-path

String

1.1.146. processor_get_register_names

Description

null

Usage

`processor_get_register_names <service-path>`

ReturnType

String[]

Returns

Arguments

service-path

1.1.147. processor_in_debug_mode

Description

null

Usage

```
processor_in_debug_mode <service-path>
```

ReturnType

boolean

Returns**Arguments**

service-path

1.1.148. processor_reset

Description

null

Usage

```
processor_reset <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.149. processor_run

Description

null

Usage

```
processor_run <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.150. processor_semihosting_start

Description

null

Usage

```
processor_semihosting_start [-file <String>] [-stdout <Boolean>] [-noconsole  
<Boolean>] [-messages <Boolean>] <service-path>
```

ReturnType

String

Returns

Arguments

String (optional)

Boolean (optional)

Boolean (optional)

Boolean (optional)

service-path

1.1.151. processor_semihosting_stop

Description

null

Usage

```
processor_semihosting_stop <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.152. processor_set_breakpoint

Description

null

Usage

```
processor_set_breakpoint [-hardware <BOOLEAN_OPTIONAL>] <service-path>  
<String>
```

ReturnType

Nothing

Returns

no return value

Arguments

BOOLEAN_OPTIONAL (optional)

service-path

String

1.1.153. processor_set_register

Description

null

Usage

```
processor_set_register <service-path> <String> <int>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

int

1.1.154. processor_step

Description

null

Usage

processor_step <service-path>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.155. processor_stop

Description

null

Usage

processor_stop <service-path>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.156. processor_stop_reason

Description

null

Usage

processor_stop_reason <service-path>

ReturnType

Map

Returns**Arguments**

service-path

1.1.157. processor_verify_elf**Description**

null

Usage

```
processor_verify_elf [-multiverify <Boolean>] [-nodata  
<BOOLEAN_OPTIONAL>] [-norodata <BOOLEAN_OPTIONAL>] [-norodata  
<BOOLEAN_OPTIONAL>] [-notext <BOOLEAN_OPTIONAL>] [-noentry  
<BOOLEAN_OPTIONAL>] [-noexceptions <BOOLEAN_OPTIONAL>] [-reset  
<BOOLEAN_OPTIONAL>] <service-path> <String>
```

Return Type

Nothing

Returns

no return value

Arguments

Boolean (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

BOOLEAN_OPTIONAL (optional)

service-path

String

1.1.158. refresh_connections

Description

Scans for available hardware and updates the available services if there have been any changes.

Usage

```
refresh_connections
```

ReturnType

Nothing

Returns

no return value

Arguments

no arguments

1.1.159. remove_service

Description

null

Usage

```
remove_service <service-type> <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-type

service-path

1.1.160. semiConsoleNew

Description

null

Usage

```
semiConsoleNew [-severAddress <String>] [-name <String>] <int>
```

ReturnType

Nothing

Returns

no return value

Arguments

String (optional)

String (optional)

int

1.1.161. semihosting_start

Description

null

Usage

```
semihosting_start [-severAddress <String>] [-name <String>] [-file <String>] [-  
stdout <Boolean>] [-noconsole <Boolean>] [-messages <Boolean>] <int>
```

Return Type

Nothing

Returns

no return value

Arguments

String (optional)

String (optional)

String (optional)

Boolean (optional)

Boolean (optional)

Boolean (optional)

int

1.1.162. semihosting_stop

Description

null

Usage

`semihosting_stop <int>`

ReturnType

Nothing

Returns

no return value

Arguments

int

1.1.163. send_message

Description

Sends a message of the given level to the message window. Available levels are info, warning, error, and debug.

Usage

`send_message <level/> <message>`

ReturnType

Nothing

Returns

no return value

Arguments

level

message

1.1.164. sld_access_dr

Description

Shifts the specified number of bits into the data register (DR), taking data from the list of bytes provided. If a positive delay is specified then the JTAG chain will pause for that long before the next action. Returns the same number of bits, or nothing if write-only

Usage

`sld_access_dr [-write_only <BOOLEAN_OPTIONAL>] <service-path> <length> <delay> <values>`

ReturnType

String[]

Returns

The same number of bits, or nothing if write-only

Arguments

BOOLEAN_OPTIONAL (optional)

service-path Service path

length Data size in number of bits

delay Delay in microseconds

values List of byte values. If empty list is provided, operation is read-only.

1.1.165. sld_access_ir**Description**

Loads the given value into the instruction register (IR) of the specified System Level Debug (SLD) node. If a positive delay is specified then the JTAG chain will pause for that long before the next action. Returns the value captured from the instruction register.

Usage

```
sld_access_ir [-ir <BOOLEAN_OPTIONAL>] [-length <length>] <service-path>  
<ir-value> <delay>
```

Return Type

String

Returns

The value captured from the instruction register.

Arguments

BOOLEAN_OPTIONAL (optional)

length (optional) ir value length

service-path Service path

ir-value ir value

delay Delay in milliseconds

1.1.166. sld_lock

Description

Locks the device that this SLD node belongs to for exclusive access. If the device is locked by another client, it will try for <timeout> milliseconds before giving up. Multiple locks by the same client are permitted but must each have a matching unlock. Returns 0 for success, -1 for any errors.

Usage

```
sld_lock <service-path> <timeout>
```

ReturnType

int

Returns

0 for success, -1 for any errors.

Arguments

service-path Service path

timeout Timeout in milliseconds

1.1.167. sld_run_test_idle

Description

Put the chain containing this device into RUN-TEST-IDLE state and remain in that state for the specified number of clocks.

Usage

```
sld_run_test_idle <service-path> <clocks>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path Service path

clocks Number of clocks

1.1.168. sld_send_program

Description

null

Usage

```
sld_send_program <service-path> <streamPath>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

streamPath

1.1.169. sld_test_logic_reset**Description**

Put the chain containing this device into TEST-LOGIC-RESET state. This operation is only permitted if this is the only device in use on the JTAG chain.

Usage

```
sld_test_logic_reset <service-path>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path Service path

1.1.170. sld_unlock**Description**

Unlocks the device that this SLD node belongs to. Returns 0 for success, -1 for any errors.

Usage

```
sld_unlock <service-path>
```

ReturnType

int

Returns

0 for success, -1 for any errors.

Arguments

service-path Service path

1.1.171. stp_run

Description

null

Usage

```
stp_run <service-path> <signal_set> <trigger> <data_log> <timeout>
```

Return Type

Nothing

Returns

no return value

Arguments

service-path

signal_set

trigger

data_log

timeout

1.1.172. system_get_keys

Description

Retrieve the keys available to query the system's SOPCINFO file content, for use in the command `system_get_values`. Keys are mapped to either the attribute or sub-element under `<EnsembleReport>` in SOPCINFO file

Usage

```
system_get_keys <service-path> [<key...>]
```

Return Type

String[]

Returns

A list of keys for the specified system path

Arguments

service-path System service path (from "get_service_paths system")

key... (optional) Key(s) mapped to attribute(s) or sub-element(s)

Example

```
system_get_keys /designs/top.sof/system.sopcinfo
```

Related Information

[system_get_values](#) on page 94

1.1.173. system_get_values

Description

Retrieve the values of the specified key in the system's SOPCINFO file

Usage

```
system_get_values <service-path> <key> [<key...>]
```

ReturnType

String[]

Returns

Arguments

service-path System service path (from "get_service_paths system")

key Key mapped to attribute or sub-element

key... (optional) Key(s) mapped to attribute(s) or sub-element(s)

Example

Given a SOPCINFO file with the following content:

```
<EnsembleReport>
...
<parameter name="param_a">
  <type>java.lang.String</type>
  <value>value_a</value>
  <derived>>false</derived>
  <enabled>>true</enabled>
  <visible>>true</visible>
  <valid>>true</valid>
</parameter>
<parameter name="param_b">
  <type>java.lang.String</type>
  <value>value_b</value>
  <derived>>false</derived>
  <enabled>>true</enabled>
  <visible>>true</visible>
  <valid>>true</valid>
</parameter>
```

```
...
</EnsembleReport>
Command:
  system_get_values /designs/top.sof/system.sopcinfo parameter
Result:
  param_a param_b
Command:
  system_get_values /designs/top.sof/system.sopcinfo parameter:param_a
Result:
  value_a
```

1.1.174. toolkit_get_toolkit_actions

Description

Get a list of action IDs for the specified instance and channel

Usage

```
toolkit_get_toolkit_actions <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of action IDs

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_actions inst1 ch1
```

1.1.175. toolkit_get_toolkit_autosweep_input_parameters

Description

Get a list of input parameters for autosweep of the specified instance and channel

Usage

```
toolkit_get_toolkit_autosweep_input_parameters <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of input parameters for autosweep

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_autosweep_input_parameters inst1 chl
```

Related Information

[toolkit_get_toolkit_autosweep_output_metrics](#) on page 96

1.1.176. toolkit_get_toolkit_autosweep_output_metrics

Description

Get a list of parameters with quality comparison callback defined for autosweep of the specified instance and channel

Usage

```
toolkit_get_toolkit_autosweep_output_metrics <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of parameters with quality comparison callback

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_autosweep_output_metrics inst1 chl
```

Related Information

[toolkit_get_toolkit_autosweep_input_parameters](#) on page 95

1.1.177. toolkit_get_toolkit_channel_properties

Description

Get a list of channel properties

Usage

```
toolkit_get_toolkit_channel_properties
```

ReturnType

String[]

Returns

A list of channel properties

Arguments

no arguments

Example

```
toolkit_get_toolkit_channel_properties
```

1.1.178. toolkit_get_toolkit_channels

Description

Get a list of toolkit channel names for the specified instance

Usage

```
toolkit_get_toolkit_channels <inst-id>
```

ReturnType

String[]

Returns

A list of toolkit channel names

Arguments

inst-id Toolkit instance ID

Example

```
toolkit_get_toolkit_channels inst1
```

1.1.179. toolkit_get_toolkit_display_hint

Description

Get the display hint value for the specified instance

Usage

```
toolkit_get_toolkit_display_hint <inst-id> <parameter-name> <display-hint>
```

ReturnType

String

Returns

Display hint value

Arguments

inst-id Toolkit instance ID

parameter-name Parameter name

display-hint Display hint

Example

```
toolkit_get_toolkit_display_hint inst1 param1 TEXT
```

1.1.180. toolkit_get_toolkit_display_id_to_name

Description

Get a Tcl map of display item's ID to display name for the specified instance and channel

Usage

```
toolkit_get_toolkit_display_id_to_name <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A Tcl map of display ID to display name

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_display_id_to_name inst1 chl
```

1.1.181. toolkit_get_toolkit_display_item_property

Description

Get the display item property by name for the specified instance

Usage

```
toolkit_get_toolkit_display_item_property <inst-id> <display-item-id>  
<display-item-property>
```

ReturnType

various

Returns

Display item property name

Arguments

inst-id Toolkit instance ID

display-item-id Display item ID

display-item-property Display item property

Example

```
toolkit_get_toolkit_display_item_property inst1 param1 DISPLAY_NAME
```

1.1.182. toolkit_get_toolkit_eye_channels

Description

Get a list of channels that support eye viewer for the specified instance

Usage

```
toolkit_get_toolkit_eye_channels <inst-id>
```

ReturnType

String[]

Returns

A list of channels that support eye viewer

Arguments

inst-id Toolkit instance ID

Example

```
toolkit_get_toolkit_eye_channels inst1
```

1.1.183. toolkit_get_toolkit_eye_input_parameters

Description

Get a list of eye input parameters for the specified instance and channel

Usage

```
toolkit_get_toolkit_eye_input_parameters <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of eye input parameters

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_eye_input_parameters inst1 ch1
```

Related Information

[toolkit_get_toolkit_eye_output_metrics](#) on page 100

1.1.184. toolkit_get_toolkit_eye_output_metrics

Description

Get a list of eye statistic parameters for the specified instance and channel

Usage

```
toolkit_get_toolkit_eye_output_metrics <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of eye statistic parameters

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_eye_output_metrics inst1 ch1
```

Related Information

[toolkit_get_toolkit_eye_input_parameters](#) on page 99

1.1.185. toolkit_get_toolkit_matching_modules

Description

Retrieve the path for IPs (modules) that match the specified arguments

Usage

```
toolkit_get_toolkit_matching_modules [-requirement <requirement-id>] [-recursive <BOOLEAN_OPTIONAL>] [-hpath <hpath>] [-module_name <module-name>] [-version <version>] <service-path>
```

ReturnType

String[]

Returns

A list of module paths matching the specified arguments

Arguments

requirement-id (optional) Requirement ID

BOOLEAN_OPTIONAL (optional)

hpath (optional) Full hierarchy path of the IP

module-name (optional) Module name (a.k.a. entity name) of the IP

version (optional) IP version

service-path Toolkit service path (from "get_service_paths toolkit")

Example

```
toolkit_get_toolkit_matching_modules -requirement req1 /toolkits/my_toolkit_1.0
```

Related Information

- [toolkit_get_toolkit_matching_services](#) on page 101
- [toolkit_get_toolkit_matching_systems](#) on page 102
- [toolkit_get_toolkit_requirement_ids](#) on page 106

1.1.186. toolkit_get_toolkit_matching_services

Description

Retrieve the path for services that match the specified arguments

Usage

```
toolkit_get_toolkit_matching_services [-requirement <requirement-id>] [-recursive <BOOLEAN_OPTIONAL>] [-hpath <hpath>] [-type <type>] [-agent_type_name <agent-type-name>] <service-path>
```

ReturnType

String[]

Returns

A list of service paths matching the specified arguments

Arguments

requirement-id (optional) Requirement ID

BOOLEAN_OPTIONAL (optional)

hpath (optional) Full hierarchy path of the service

type (optional) Service type (e.g. master, slave, etile)

agent-type-name (optional) Debug agent type name (e.g. jtag_phy.jtag)

service-path Toolkit service path (from "get_service_paths toolkit")

Example

```
toolkit_get_toolkit_matching_services -requirement req1 /toolkits/my_toolkit_1.0
```

Related Information

- [toolkit_get_toolkit_matching_modules](#) on page 100
- [toolkit_get_toolkit_matching_systems](#) on page 102
- [toolkit_get_toolkit_requirement_ids](#) on page 106

1.1.187. toolkit_get_toolkit_matching_systems

Description

Retrieve the path for systems that match the specified arguments

Usage

```
toolkit_get_toolkit_matching_systems [-requirement <requirement-id>] [-hpath <hpath>] [-design_id <design-id>] <service-path>
```

ReturnType

String[]

Returns

A list of system paths matching the specified requirement

Arguments

requirement-id (optional) Requirement ID

hpath (optional) Full hierarchy path of the system

design-id (optional) Design ID of the system

service-path Toolkit service path (from "get_service_paths toolkit")

Example

```
toolkit_get_toolkit_matching_systems -requirement req1 /toolkits/my_toolkit_1.0
```

Related Information

- [toolkit_get_toolkit_matching_modules](#) on page 100
- [toolkit_get_toolkit_matching_services](#) on page 101
- [toolkit_get_toolkit_requirement_ids](#) on page 106

1.1.188. toolkit_get_toolkit_parameter_properties

Description

Get a list of parameter properties

Usage

```
toolkit_get_toolkit_parameter_properties
```

ReturnType

String[]

Returns

A list of parameter properties

Arguments

no arguments

Example

```
toolkit_get_toolkit_parameter_properties
```

1.1.189. toolkit_get_toolkit_parameter_property

Description

Get the parameter property by name for the specified instance

Usage

```
toolkit_get_toolkit_parameter_property <inst-id> <parameter-name>  
<parameter-property>
```

ReturnType

various

Returns

Parameter property name

Arguments

inst-id Toolkit instance ID

parameter-name Parameter name

parameter-property Parameter property

Example

```
toolkit_get_toolkit_parameter_property inst1 param1 ENABLED
```

1.1.190. toolkit_get_toolkit_parameter_value

Description

Get the parameter value by name for the specified instance

Usage

```
toolkit_get_toolkit_parameter_value <inst-id> <parameter-name>
```

ReturnType

String

Returns

Parameter value

Arguments

inst-id Toolkit instance ID

parameter-name Parameter name

Example

```
toolkit_get_toolkit_parameter_value inst1 param1
```

Related Information

[toolkit_set_toolkit_parameter_value](#) on page 110

1.1.191. toolkit_get_toolkit_parameters

Description

Get a list of parameter IDs for the specified instance and channel

Usage

```
toolkit_get_toolkit_parameters <inst-id> [<channel>]
```

ReturnType

String[]

Returns

A list of parameter IDs

Arguments

inst-id Toolkit instance ID

channel (optional) Channel name

Example

```
toolkit_get_toolkit_parameters inst1 chl
```

1.1.192. toolkit_get_toolkit_properties

Description

Return a list of toolkit property names (e.g. name, display_name, version, description)

Usage

```
toolkit_get_toolkit_properties <service-path>
```

Return Type

String[]

Returns

A list of toolkit property names

Arguments

service-path Toolkit service path (from "get_service_paths toolkit")

Example

```
toolkit_get_toolkit_properties /toolkits/my_toolkit_1.0
```

Related Information

[toolkit_get_toolkit_property](#) on page 105

1.1.193. toolkit_get_toolkit_property

Description

Return the value of the specified property

Usage

```
toolkit_get_toolkit_property <service-path> <property>
```

Return Type

String

Returns

The value of toolkit property

Arguments

service-path Toolkit service path (from "get_service_paths toolkit")

property property to get value

Example

```
toolkit_get_toolkit_property /toolkits/my_toolkit_1.0 description
```

Related Information

[toolkit_get_toolkit_properties](#) on page 105

1.1.194. toolkit_get_toolkit_requirement_ids

Description

Retrieve requirement IDs defined in the specified toolkit

Usage

```
toolkit_get_toolkit_requirement_ids <service-path>
```

ReturnType

String[]

Returns

A list of requirement IDs

Arguments

service-path Toolkit service path (from "get_service_paths toolkit")

Example

```
toolkit_get_toolkit_requirement_ids /toolkits/my_toolkit_1.0
```

1.1.195. toolkit_get_toolkit_requirement_properties

Description

Return all property names defined in the specified requirement

Usage

```
toolkit_get_toolkit_requirement_properties <service-path>  
<requirement-id>
```

ReturnType

String[]

Returns

A list of property names

Arguments

service-path Toolkit service path (from "get_service_paths toolkit")

requirement-id Requirement ID

Example

```
toolkit_get_toolkit_requirement_properties /toolkits/my_toolkit_1.0 req1
```

Related Information

- [toolkit_get_toolkit_requirement_ids](#) on page 106
- [toolkit_get_toolkit_requirement_property](#) on page 107

1.1.196. toolkit_get_toolkit_requirement_property

Description

Return value(s) of the property in the specified requirement

Usage

```
toolkit_get_toolkit_requirement_property <service-path> <requirement-id> <property>
```

ReturnType

String[]

Returns

A list of property values

Arguments

service-path Toolkit service path (from "get_service_paths toolkit")

requirement-id Requirement ID

property Property name (e.g. hpath)

Example

```
toolkit_get_toolkit_requirement_property /toolkits/my_toolkit_1.0 req1 hpath
```

Related Information

- [toolkit_get_toolkit_requirement_ids](#) on page 106
- [toolkit_get_toolkit_requirement_properties](#) on page 106

1.1.197. toolkit_load_toolkit_instance

Description

Load a toolkit (equivalent to launching a toolkit in GUI but headlessly)

Usage

```
toolkit_load_toolkit_instance [-id <inst-id>] [-log <log-flag>] [-output-file-path <output-file-path>] <toolkit-name> <version> <requirements>
```

ReturnType

String

Returns

Instance ID of the loaded toolkit

Arguments

inst-id (optional) ID of the loaded toolkit instance, auto-assigned and returned if not specified

log-flag (optional) Flag used to start a toolkit with command logging enabled. System console must have been started with the "enable-command-logging" command line argument to enable option.

output-file-path (optional) Name of file that logged commands will be captured in.

toolkit-name Toolkit name

version Toolkit version

requirements List of requirements and paths of the selected elements in key-value pairs separated by colon

Example

```
toolkit_load_toolkit_instance my_toolkit 1.0 {req0:/designs/a/b/c req1:/
devices/x/y/z}
  where:
  1. req0 is the requirement ID of type IP defined by "add_requirement" in
  Toolkit Tcl
  2. req1 is the requirement ID of type SERVICE defined by "add_requirement"
  in Toolkit Tcl
  3. req0 and req1 can be enumerated using toolkit_get_toolkit_requirement_ids
  4. /designs/a/b/c is the service service-path obtained from
  toolkit_get_toolkit_matching_services
  5. /devices/x/y/z is the module service-path obtained from
  toolkit_get_toolkit_matching_modules
```

Related Information

- [toolkit_get_toolkit_matching_modules](#) on page 100
- [toolkit_get_toolkit_matching_services](#) on page 101

- [toolkit_get_toolkit_matching_systems](#) on page 102
- [toolkit_unload_toolkit_instance](#) on page 111

1.1.198. toolkit_log_toolkit_command_start

Description

Starts command logging. This command must be enabled with the command line argument "enable_command_logging". These logged commands can be displayed in the messages window of system console or can be captured in a file. The append flag can be used to append logged commands to the end of a file.

Usage

```
toolkit_log_toolkit_command_start <id> [<file> <append>]
```

ReturnType

Nothing

Returns

no return value

Arguments

id Toolkit instance ID

file (optional) Captured commands file

append (optional) Append flag

Example

```
log_toolkit_command_start inst1 "logged_commands_file.txt" append
```

1.1.199. toolkit_log_toolkit_command_stop

Description

Stops logging commands.

Usage

```
toolkit_log_toolkit_command_stop <id>
```

ReturnType

Nothing

Returns

no return value

Arguments

id Toolkit instance ID

Example

```
log_toolkit_command_stop inst1
```

1.1.200. toolkit_run_toolkit_action**Description**

Invoke the callback associated with the given ACTION display item for the specified instance

Usage

```
toolkit_run_toolkit_action <inst-id> <display-item-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

inst-id Toolkit instance ID

display-item-id Display item ID

Example

```
toolkit_run_toolkit_action inst1 button1
```

1.1.201. toolkit_set_toolkit_parameter_value**Description**

Set the parameter value by name for the specified instance

Usage

```
toolkit_set_toolkit_parameter_value <inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

inst-id Toolkit instance ID

Example

```
toolkit_set_toolkit_parameter_value inst1 param1 666
```

Related Information

[toolkit_get_toolkit_parameter_value](#) on page 104

1.1.202. toolkit_unload_toolkit_instance

Description

Unload a toolkit (equivalent to closing a toolkit in GUI but headlessly)

Usage

```
toolkit_unload_toolkit_instance <inst-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

inst-id ID of the toolkit instance to unload

Example

```
toolkit_unload_toolkit_instance my_instance_name
```

Related Information

[toolkit_load_toolkit_instance](#) on page 108

1.1.203. trace_db_delete_snapshot

Description

null

Usage

```
trace_db_delete_snapshot <service-path> <snapshot-id>
```

ReturnType

Nothing

Returns

no return value

Arguments

service-path

snapshot-id

1.1.204. trace_db_get_snapshot

Description

null

Usage

```
trace_db_get_snapshot [-start-time <start-time>] [-end-time <end-time>] [-monitor-ids <monitor-ids>] <service-path>
```

Return Type

int

Returns**Arguments**

start-time (optional)

end-time (optional)

monitor-ids (optional)

service-path

1.1.205. trace_db_snapshot_get_event_data

Description

null

Usage

```
trace_db_snapshot_get_event_data <service-path> <snapshot-id> <event-id> <field>
```

Return Type

String

Returns**Arguments**

service-path

snapshot-id

event-id

field

1.1.206. trace_db_snapshot_get_event_fields

Description

null

Usage

```
trace_db_snapshot_get_event_fields <service-path> <snapshot-id> <event-id>
```

Return Type

Set

Returns

Arguments

service-path

snapshot-id

event-id

1.1.207. trace_db_snapshot_get_event_kind

Description

null

Usage

```
trace_db_snapshot_get_event_kind <service-path> <snapshot-id> <event-id>
```

Return Type

Kind

Returns

See [Kind](#).

Arguments

service-path

snapshot-id

event-id

1.1.208. trace_db_snapshot_get_event_timestamp

Description

null

Usage

```
trace_db_snapshot_get_event_timestamp <service-path> <snapshot-id> <event-id>
```

ReturnType

long

Returns**Arguments**

service-path

snapshot-id

event-id

1.1.209. trace_db_snapshot_get_events

Description

null

Usage

```
trace_db_snapshot_get_events <service-path> <snapshot-id>
```

ReturnType

String[]

Returns**Arguments**

service-path

snapshot-id

1.1.210. trace_decoder_add_key_result

Description

null

Usage

```
trace_decoder_add_key_result <field> <parent> <name> <value>
```

ReturnType

String

Returns

Arguments

field

parent

name

value

1.1.211. trace_decoder_add_result

Description

null

Usage

```
trace_decoder_add_result <parent> <name> <value>
```

ReturnType

String

Returns

Arguments

parent

name

value

1.1.212. trace_decoder_define_key

Description

null

Usage

```
trace_decoder_define_key <name> <display_name>
```

ReturnType

Nothing

Returns

no return value

Arguments

name

display_name

1.1.213. trace_decoder_get_config**Description**

null

Usage

`trace_decoder_get_config <regNum>`

ReturnType

int

Returns**Arguments**

regNum

1.1.214. trace_decoder_get_data_16**Description**

null

Usage

`trace_decoder_get_data_16 <offset>`

ReturnType

int

Returns**Arguments**

offset

1.1.215. trace_decoder_get_data_16be**Description**

null

Usage

`trace_decoder_get_data_16be <offset>`

ReturnType

int

Returns

Arguments

offset

1.1.216. trace_decoder_get_data_32

Description

null

Usage

trace_decoder_get_data_32 <*offset*>

ReturnType

int

Returns

Arguments

offset

1.1.217. trace_decoder_get_data_32be

Description

null

Usage

trace_decoder_get_data_32be <*offset*>

ReturnType

int

Returns

Arguments

offset

1.1.218. trace_decoder_get_data_64

Description

null

Usage

trace_decoder_get_data_64 <*offset*>

ReturnType

BigInteger

Returns**Arguments**

offset

1.1.219. trace_decoder_get_data_64be**Description**

null

Usage

trace_decoder_get_data_64be <*offset*>

ReturnType

BigInteger

Returns**Arguments**

offset

1.1.220. trace_decoder_get_data_8**Description**

null

Usage

trace_decoder_get_data_8 <*offset*>

ReturnType

int

Returns**Arguments**

offset

1.1.221. trace_decoder_get_length**Description**

null

Usage

trace_decoder_get_length

ReturnType

int

Returns

Arguments

no arguments

1.1.222. trace_decoder_get_timestamp

Description

null

Usage

trace_decoder_get_timestamp

ReturnType

BigInteger

Returns

Arguments

no arguments

1.1.223. trace_decoder_set_callback

Description

null

Usage

trace_decoder_set_callback <callback>

ReturnType

Nothing

Returns

no return value

Arguments

callback

1.1.224. trace_decoder_set_config_regs

Description

null

Usage

trace_decoder_set_config_regs <regs>

ReturnType

Nothing

Returns

no return value

Arguments

regs

1.1.225. trace_decoder_set_summary**Description**

null

Usage

`trace_decoder_set_summary <value>`

ReturnType

Nothing

Returns

no return value

Arguments

value

1.1.226. trace_get_db_size**Description**

null

Usage

`trace_get_db_size <service-path>`

ReturnType

long

Returns**Arguments**

service-path

1.1.227. trace_get_max_db_size**Description**

null

Usage

`trace_get_max_db_size <service-path>`

ReturnType

long

Returns

Arguments

service-path

1.1.228. trace_get_monitor_info

Description

null

Usage

`trace_get_monitor_info <service-path> <String>`

ReturnType

Map

Returns

Arguments

service-path

String

1.1.229. trace_get_monitors

Description

null

Usage

`trace_get_monitors <service-path>`

ReturnType

String[]

Returns

Arguments

service-path

1.1.230. trace_get_status

Description

null

Usage

trace_get_status <service-path>

ReturnType

Status

Returns

See [Status](#).

Arguments

service-path

1.1.231. trace_load

Description

null

Usage

trace_load <filename>

ReturnType

String

Returns**Arguments**

filename

1.1.232. trace_read_monitor

Description

null

Usage

trace_read_monitor <service-path> <String> <int>

ReturnType

String

Returns

Arguments

service-path

String

int

1.1.233. trace_save

Description

null

Usage

trace_save <*service-path*> <*String*>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

1.1.234. trace_set_max_db_size

Description

null

Usage

trace_set_max_db_size <*service-path*> <*long*>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

long

1.1.235. trace_set_trigger_mode

Description

null

Usage

trace_set_trigger_mode <service-path> <TriggerMode>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

TriggerMode See [Trigger Mode](#).

1.1.236. trace_start

Description

null

Usage

trace_start <service-path> <CaptureMode>

ReturnType

Nothing

Returns

no return value

Arguments

service-path

CaptureMode See [Capture Mode](#).

1.1.237. trace_stop

Description

null

Usage

`trace_stop <service-path>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

1.1.238. trace_write_monitor

Description

null

Usage

`trace_write_monitor <service-path> <String> <int> <String>`

ReturnType

Nothing

Returns

no return value

Arguments

service-path

String

int

String

1.2. Properties

1.2.1. _hw.tcl Capture Mode

Name	Description
FIFO	Events are transmitted to host as they are captured via a FIFO buffer.
STORE_AND_FORWARD	Events are stored in memory and only transmitted to host when capturing is stopped either explicitly or because a trigger condition is met.

1.2.2. _hw.tcl Command Flag Name

Name	Description
NOT_REMOTE_SAFE	Describes that this command is not safe for remote use.
NOT_THREAD_SAFE	Describes that this command is not thread safe.
REMOTE_SAFE	Describes that this command is safe for remote use.
THREAD_SAFE	Describes that this command is thread safe.

1.2.3. _hw.tcl Format

Name	Description
_32	Word list format (32 bits).
_8	Byte list format (8 bits).

1.2.4. _hw.tcl Kind

Name	Description
DATA	A normal event containing data.
DATA_CORRUPTION	An event containing corrupted or truncated data.
EVENT_LOSS	An event indicating that one or more trace events have been lost between this event and previously received event.

1.2.5. _hw.tcl Protocol

Name	Description
I2C	Specifies the use of I2C as the communication protocol used by the IO Bus.
SPI	Specifies the use of SPI as the communication protocol used by the IO Bus.

1.2.6. _hw.tcl Status

Name	Description
IDLE	The capture controller is not accepting trace events for transmission or storage.
RUNNING	The capture controller is accepting trace events for transmission or storage.

1.2.7. _hw.tcl Trigger Mode

Name	Description
CENTER	Events around the trigger will be captured.
NONE	Triggering disabled.
POST	Events after the trigger will be captured.
PRE	Events before the trigger will be captured.

2. Toolkit Tcl Command References

For advanced users, System Console supports Tcl commands that allow you to define and operate your own toolkits. You can use the Toolkit Tcl commands to add and set the toolkit requirements and properties, and to retrieve accessible toolkit modules, systems, and services.

2.1. Command Summary

Returns	Command	Arguments
no return value	add_channel	<channel-name> <channel-type>
no return value	add_display_item	<parent-group> <id> <type> [<args>]
no return value	add_parameter	<name> <type> [<value> <description>]
no return value	add_requirement	<req-id> <req-type>
no return value	add_timed_callback	<proc> <interval>
String	get_accessible_module	[-hpath <hpath>] [-module_name <module-name>] [-version <version>] <req-id>
String[]	get_accessible_modules	[-hpath <hpath>] [-module_name <module-name>] [-version <version>] <req-id>
String	get_accessible_service	[-hpath <hpath>] [-type <type>] [-agent_type_name <agent-type-name>] <req-id>
String[]	get_accessible_services	[-hpath <hpath>] [-type <type>] [-agent_type_name <agent-type-name>] <req-id>
String	get_accessible_system	[-hpath <hpath>] [-design_id <design-id>] <req-id>
String	get_channel_display_group	<channel>
various	get_channel_property	<channel-name> <property>
String	get_display_hint	<id> <display-hint>
various	get_display_item_property	<id> <display-item-property>
String	get_eye_viewer_display_group	<channel>
various	get_parameter_property	<name> <property>
String	get_parameter_value	<name>
String	get_toolkit_property	<prop>
no return value	remove_timed_callback	<proc>
no return value	send_message	<level> <message>
no return value	set_channel_property	<channel-name> <property> <property-value>
no return value	set_current_progress	<progress>

continued...

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

*Other names and brands may be claimed as the property of others.

Returns	Command	Arguments
no return value	set_display_hint	<id> <display-hint> <value>
no return value	set_display_item_property	<id> <display-property> <display-property-value>
no return value	set_eye_data	<rows>
no return value	set_eye_property	<property> <value>
no return value	set_parameter_property	<name> <property> <value>
no return value	set_parameter_update_callback	<name> <callback> [<opt-arg>]
no return value	set_parameter_value	<name> <value>
no return value	set_requirement_property	<req-id> <property-name> <property-values>
no return value	set_toolkit_property	<prop> <value>
boolean	stop_requested	

2.1.1.1. add_channel

Description

This adds a channel to your toolkit.

Availability

CHANNEL

Usage

```
add_channel <channel-name> <channel-type>
```

Return Type

Nothing

Returns

no return value

Arguments

channel-name The name of the channel.

channel-type The data type of the channel. See [Channel Type](#).

Example

```
add_channel "Channel 0" OTHER
```

Related Information

- [get_channel_property](#) on page 136
- [set_channel_property](#) on page 142

2.1.2. add_display_item

Description

This command adds a display item to the toolkit.

Availability

GLOBAL

Usage

```
add_display_item <parent-group> <id> <type> [<args>]
```

ReturnType

Nothing

Returns

no return value

Arguments

parent-group Parent group of the display item

id ID of this display item

type Type of this display item this can be ACTION, GROUP, ICON, PARAMETER, TEXT, DIAL, or LED See [Display Item Type](#).

args (optional) Optional argument that will be supplied to the callback

Example

```
add_display_item "Timing" read_latency PARAMETER  
add_display_item "Sounds" speaker_image_id ICON speaker.jpg
```

Related Information

- [get_display_hint](#) on page 137
- [get_display_item_property](#) on page 138
- [set_display_hint](#) on page 143
- [set_display_item_property](#) on page 144

2.1.3. add_parameter

Description

This command adds a new parameter to this toolkit.

Availability

GLOBAL

Usage

`add_parameter <name> <type> [<value> <description>]`

ReturnType

Nothing

Returns

no return value

Arguments

name Parameter name

type Parameter type See [Parameter Type](#).

value (optional) Parameter default value

description (optional) Parameter description

Example

```
add_parameter myparam INTEGER 1
```

Related Information

- [get_parameter_value](#) on page 139
- [set_parameter_property](#) on page 146
- [set_parameter_value](#) on page 148

2.1.4. add_requirement

Description

Define a requirement of a specific type.

Availability

DISCOVER, GLOBAL

Usage

`add_requirement <req-id> <req-type>`

ReturnType

Nothing

Returns

no return value

Arguments

req-id Requirement ID which is a string of alpha-numerical characters.

req-type Requirement type See [Requirement Type](#).

Example

```
add_requirement req1 SYSTEM
```

Related Information

[set_requirement_property](#) on page 148

2.1.5. add_timed_callback

Description

Used to add a callback procedure that is called at a fixed timed interval (in milliseconds). Interval has an allowed range of 500 - 60000. These callbacks execute sequentially, and when added, are scheduled to execute at the next available time slot.

Availability

ACTION, AUTOSWEEP_FINALIZATION, AUTOSWEEP_INITIALIZATION, AUTOSWEEP_QUALITY_METRIC, CHART, EYE_VIEWER, PARAMETER_UPDATE

Usage

```
add_timed_callback <proc> <interval>
```

Return Type

Nothing

Returns

no return value

Arguments

proc The name of the callback tcl proc that will be called when the timed callback triggers.

interval The length of time between callback triggers. This time is in milliseconds.

Example

```
add_timed_callback my_timed_callback 500
```

Related Information

[remove_timed_callback](#) on page 141

2.1.6. get_accessible_module

Description

Retrieve the IP targeted by the specified requirement. There is always only one such IP. Nothing is returned if the requirement is not an IP or OPTIONAL_IP requirement. To retrieve IP(s) that matches under a SYSTEM requirement, see `get_accessible_modules`.

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_accessible_module [-hpath <hpath>] [-module_name <module-name>] [-version <version>] <req-id>
```

ReturnType

String

Returns

Module path

Arguments

hpath (optional) Full hierarchy path of the IP

module-name (optional) Module name (a.k.a. entity name) of the IP

version (optional) IP version

req-id Requirement ID

Example

```
get_accessible_module req1
```

Related Information

- [get_accessible_modules](#) on page 132
- [get_accessible_service](#) on page 133
- [get_accessible_services](#) on page 134
- [get_accessible_system](#) on page 135

2.1.7. get_accessible_modules

Description

For IP/OPTIONAL_IP requirement, retrieve the only IP targeted by the specified requirement. For SYSTEM requirement, retrieve the IPs that match IPS/OPTIONAL_IPS sub-requirement.

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_accessible_modules [-hpath <hpath>] [-module_name <module-name>] [-  
version <version>] <req-id>
```

Return Type

String[]

Returns

A list of module paths

Arguments

hpath (optional) Full hierarchy path of the IP

module-name (optional) Module name (a.k.a. entity name) of the IP

version (optional) IP version

req-id Requirement ID

Example

```
get_accessible_modules req1
```

Related Information

- [get_accessible_module](#) on page 132
- [get_accessible_service](#) on page 133
- [get_accessible_services](#) on page 134
- [get_accessible_system](#) on page 135

2.1.8. get_accessible_service

Description

Retrieve the service targeted by the specified requirement. There is always only one such service. Nothing is returned if the requirement is not a SERVICE requirement. To retrieve service(s) that matches under a SYSTEM/IP/OPTIONAL_IP requirement or IPS/OPTIONAL_IPS sub-requirement, see [get_accessible_services](#).

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_accessible_service [-hpath <hpath>] [-type <type>] [-agent_type_name  
<agent-type-name>] <req-id>
```

ReturnType

String

Returns

Service path

Arguments

hpath (optional) Full hierarchy path of the service

type (optional) Service type (e.g. master, slave, etile)

agent-type-name (optional) Debug agent type name

req-id Requirement ID

Example

```
get_accessible_service req1
```

Related Information

- [get_accessible_module](#) on page 132
- [get_accessible_modules](#) on page 132
- [get_accessible_services](#) on page 134
- [get_accessible_system](#) on page 135

2.1.9. get_accessible_services

Description

For SERVICE requirement, retrieve the only service targeted by the specified requirement. For IP/OPTIONAL_IP requirement, retrieve the services that match the SERVICES/OPTIONAL_SERVICES sub-requirement. For SYSTEM requirement, retrieve the services that match the SERVICES/OPTIONAL_SERVICES sub-requirement or SERVICES/OPTIONAL_SERVICES sub-sub-requirement under the IPS/OPTIONAL_IPS sub-requirement.

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_accessible_services [-hpath <hpath>] [-type <type>] [-agent_type_name  
<agent-type-name>] <req-id>
```

ReturnType

String[]

Returns

A list of service paths

Arguments

hpath (optional) Full hierarchy path of the service

type (optional) Service type(e.g. master, slave, etile)

agent-type-name (optional) Debug agent type name

req-id Requirement ID

Example

```
get_accessible_services req1
```

Related Information

- [get_accessible_module](#) on page 132
- [get_accessible_modules](#) on page 132
- [get_accessible_service](#) on page 133
- [get_accessible_system](#) on page 135

2.1.10. get_accessible_system

Description

Retrieve the system targeted by the specified requirement. There is always only one such system. Nothing is returned if the requirement is not a SYSTEM requirement.

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_accessible_system [-hpath <hpath>] [-design_id <design-id>] <req-id>
```

Return Type

String

Returns

System path

Arguments

hpath (optional) Full hierarchy path of the system.

design-id (optional) Design ID of the system

req-id Requirement ID

Example

```
get_accessible_system req1
```

Related Information

- [get_accessible_module](#) on page 132
- [get_accessible_modules](#) on page 132
- [get_accessible_service](#) on page 133
- [get_accessible_services](#) on page 134

2.1.11. get_channel_display_group**Description**

This command returns the associated display group for a given channel's display area.

Availability

GLOBAL

Usage

```
get_channel_display_group <channel>
```

ReturnType

String

Returns

Associated channel display group name.

Arguments

channel The name of the targeted display group.

Example

```
set my_chan_display_group [ get_channel_display_group "Channel 0" ]
```

2.1.12. get_channel_property**Description**

Retrieves a property of the channel.

Availability

All contexts except: DISCOVER, GLOBAL

Usage

```
get_channel_property <channel-name> <property>
```

ReturnType

various

Returns

The channel property matching the specified channel property type.

Arguments

channel-name The name of the channel.

property The name of the channel property.

Example

```
get_channel_property "Channel_0" SUPPORTS_EYE
```

Related Information

- [add_channel](#) on page 128
- [set_channel_property](#) on page 142

2.1.13. get_display_hint

Description

Retrieve the value of a display hint from a specified display item.

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER

Usage

```
get_display_hint <id> <display-hint>
```

ReturnType

String

Returns

Returns the value of the display hint.

Arguments

id ID of the display hint's display item.

display-hint Name of the display hint.

Example

```
set file_loc [ get_display_hint my_disp_item FILE ]
```

Related Information

- [add_display_item](#) on page 129
- [get_display_item_property](#) on page 138
- [set_display_hint](#) on page 143

- [set_display_item_property](#) on page 144

2.1.14. [get_display_item_property](#)

Description

Retrieves a property of a display item.

Availability

All contexts except: CHANNEL, DISCOVER

Usage

```
get_display_item_property <id> <display-item-property>
```

ReturnType

various

Returns

Returns the value of the display item's property.

Arguments

id ID of the property' display item.

display-item-property The name of the property.

Example

```
set my_label [get_display_item_property my_action DISPLAY_NAME]
```

Related Information

- [add_display_item](#) on page 129
- [get_display_hint](#) on page 137
- [set_display_hint](#) on page 143
- [set_display_item_property](#) on page 144

2.1.15. [get_eye_viewer_display_group](#)

Description

This command returns the associated display group for a given channel's eye viewer display area.

Availability

GLOBAL

Usage

```
get_eye_viewer_display_group <channel>
```

ReturnType

String

Returns

Associated eye view display group name.

Arguments

channel The name of the targeted channel.

Example

```
set my_eye_display_group [ get_eye_viewer_display_group "Channel 0" ]
```

2.1.16. get_parameter_property

Description

This command returns the property value of a specific parameter.

Availability

All contexts except: CHANNEL, DISCOVER

Usage

```
get_parameter_property <name> <property>
```

Return Type

various

Returns

The value of the property.

Arguments

name Parameter name

property Parameter type

Example

```
set param_default [ get_parameter_property myparam DEFAULT_VALUE ]
```

Related Information

- [add_parameter](#) on page 129
- [get_parameter_value](#) on page 139
- [set_parameter_property](#) on page 146
- [set_parameter_value](#) on page 148

2.1.17. get_parameter_value

Description

This command returns the current value of a parameter defined previously with the `add_parameter` command.

Availability

All contexts except: CHANNEL, DISCOVER

Usage

```
get_parameter_value <name>
```

ReturnType

String

Returns

The value of the parameter.

Arguments

name Parameter name

Example

```
set myvar [ get_parameter_value myparam ]
```

Related Information

- [add_parameter](#) on page 129
- [get_parameter_property](#) on page 139
- [set_parameter_property](#) on page 146
- [set_parameter_value](#) on page 148

2.1.18. get_toolkit_property

Description

This command retrieves the value of a single toolkit property.

Availability

All contexts except: DISCOVER

Usage

```
get_toolkit_property <prop>
```

ReturnType

String

Returns

Toolkit property value.

Arguments

prop Name of the toolkit property that will be retrieved.

Example

```
set tk_display_name [ get_toolkit_property DISPLAY_NAME ]
```

Related Information

[set_toolkit_property](#) on page 149

2.1.19. remove_timed_callback

Description

This command removes previously added timed callbacks. The removal of these timed callbacks removes them from being scheduled.

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER, ELABORATION, FINALIZATION, GLOBAL, INITIALIZATION

Usage

```
remove_timed_callback <proc>
```

ReturnType

Nothing

Returns

no return value

Arguments

proc The name of the previously added Tcl callback procedure.

Example

```
add_timed_callback my_timed_callback 500  
remove_timed_callback my_timed_callback
```

Related Information

[add_timed_callback](#) on page 131

2.1.20. send_message

Description

Send a message that will be displayed in the message window.

Availability

All contexts except: DISCOVER

Usage

```
send_message <level> <message>
```

ReturnType

Nothing

Returns

no return value

Arguments

level The following message levels are supported: * ERROR: Provides an error message * WARNING: Provides a warning message * INFO: Provides an informational message * DEBUG: Provides a debug message when debug mode is enabled

message The string that will be displayed by the message window.

Example

```
send_message ERROR "The system is down!"
```

2.1.21. set_channel_property

Description

This command sets a channel's property.

Availability

CHANNEL

Usage

```
set_channel_property <channel-name> <property> <property-value>
```

Return Type

Nothing

Returns

no return value

Arguments

channel-name The name of the channel.

property The property of the channel that will be changed. See [Channel Properties](#).

property-value New value of the property.

Example

```
set_channel_property "Channel 0" SUPPORTS_EYE true
```

Related Information

- [add_channel](#) on page 128
- [get_channel_property](#) on page 136

2.1.22. set_current_progress

Description

This command is used to set the current progress bar value. By default, this value is set to 0 for 0%.

Availability

EYE_VIEWER

Usage

```
set_current_progress <progress>
```

Return Type

Nothing

Returns

no return value

Arguments

progress The new value of the progress bar.

Example

```
set_current_progress 0
set prog_max 100

for {set curr_prog 1} {$curr_prog <= $prog_max} {incr curr_prog 10} {
    after 250;

    # perform some action ...

    # update current progress after action completes

    set_current_progress $curr_prog
}
```

2.1.23. set_display_hint

Description

This command configures a display hint value for a specified display item.

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER

Usage

```
set_display_hint <id> <display-hint> <value>
```

Return Type

Nothing

Returns

no return value

Arguments

id ID of the display hint's display item.

display-hint Name of the display hint. See [Display Hint](#).

value The new display hint value.

Example

```
set_display_hint blocking_action NON_BLOCKING false
```

Related Information

- [add_display_item](#) on page 129
- [get_display_hint](#) on page 137
- [get_display_item_property](#) on page 138
- [set_display_item_property](#) on page 144

2.1.24. set_display_item_property

Description

Set a property of a display item

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER

Usage

```
set_display_item_property <id> <display-property> <display-property-value>
```

Return Type

Nothing

Returns

no return value

Arguments

id ID of the display item that will have a property set.

display-property The property of the display item that will be set. See [Display Item Properties](#).

display-property-value The new value of the display item property.

Example

```
set_display_item_property my_action DISPLAY_NAME "Click Me"
```


Related Information

- [add_display_item](#) on page 129
- [get_display_hint](#) on page 137
- [get_display_item_property](#) on page 138
- [set_display_hint](#) on page 143

2.1.25. set_eye_data

Description

This command sets the eye data to plot in the eye viewer's heat map graph diagram.

Availability

EYE_VIEWER

Usage

```
set_eye_data <rows>
```

Return Type

Nothing

Returns

no return value

Arguments

rows Eye data expressed as a list of lists. Each inner list corresponds to a displayed row in the heat map. The number of elements inside each inner list define the number of columns in the map. The value of these elements define the color intensity of each cell in the heat map.

Example

```
# The following example draws a dummy eye
set eye_data_str {}
lappend eye_data_str [list 1 1 1 1 1 1 1 1 1 1 ]
lappend eye_data_str [list 1 0.9 0.03 0.004 0.0002 0.0002 0.004 0.03 0.9 1 ]
lappend eye_data_str [list 1 0.05 0.0003 0.000004 0.00000002 0.00000002
0.000004 0.0003 0.05 1 ]
lappend eye_data_str [list 1 0.0005 0.000003 0.00000002 0 0 0.00000002 0.000003
0.0005 1 ]
lappend eye_data_str [list 1 0.05 0.0003 0.000004 0.00000002 0.00000002
0.000004 0.0003 0.9 1 ]
lappend eye_data_str [list 1 0.9 0.03 0.004 0.0002 0.0002 0.004 0.03 0.9 1 ]
lappend eye_data_str [list 1 1 1 1 1 1 1 1 1 1 ]
set_eye_data $eye_data_str
```

2.1.26. set_eye_property

Description

This command sets a single property that controls the eye viewer layout.

Availability

EYE_VIEWER

Usage

`set_eye_property <property> <value>`

ReturnType

Nothing

Returns

no return value

Arguments

property The name of the eye property. See [Eye Properties](#).

value The new value of the property.

Example

```
set_eye_property x_step 2
```

2.1.27. set_parameter_property

Description

This command sets a single parameter property.

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER

Usage

`set_parameter_property <name> <property> <value>`

ReturnType

Nothing

Returns

no return value

Arguments

name Parameter name

property Parameter property See [Parameter Properties](#).

value Parameter value

Example

```
set_parameter_value myparam 10
```

Related Information

- [add_parameter](#) on page 129
- [get_parameter_property](#) on page 139
- [get_parameter_value](#) on page 139
- [set_parameter_value](#) on page 148

2.1.28. set_parameter_update_callback

Description

This command sets a tcl callback procedure for a parameter. This callback is called whenever the associated parameter value changes. The callback_proc will receive a single argument. If <optional_argument> is specified, the value of the argument to the callback <optional_argument> is used. Otherwise, the argument value will be the name of the parameter that was changed. This command allows one callback procedure to handle updates for multiple parameters.

Availability

GLOBAL

Usage

```
set_parameter_update_callback <name> <callback> [<opt-arg>]
```

Return Type

Nothing

Returns

no return value

Arguments

name Parameter name

callback Update callback

opt-arg (optional) Optional Arguments

Example

```
set_parameter_update_callback p1 p1_ui_tweak
proc p1_ui_tweak {arg} {
    set p1 [get_parameter_value $arg]
    set_parameter_value p2 [ expr $p1 / 4 ]
}
set_parameter_update_callback p2 ui_tweak Context1
set_parameter_update_callback p3 ui_tweak Context2
set_parameter_update_callback p4 ui_tweak Context2
proc ui_tweak {arg} {
    if {$arg eq "Context1"} {
        set_parameter_value p1 0
    }
    if {$arg eq "Context2"} {
```

```
        set_parameter_value pl 1  
    }  
}
```

2.1.29. set_parameter_value

Description

This command sets a parameters value.

Availability

All contexts except: AUTOSWEEP_CASE_VALIDITY, CHANNEL, DISCOVER, GLOBAL

Usage

```
set_parameter_value <name> <value>
```

ReturnType

Nothing

Returns

no return value

Arguments

name Parameter name

value Parameter value

Example

```
set_parameter_value myparam 10
```

Related Information

- [add_parameter](#) on page 129
- [get_parameter_property](#) on page 139
- [get_parameter_value](#) on page 139
- [set_parameter_property](#) on page 146

2.1.30. set_requirement_property

Description

Set the property of a specific requirement.

Availability

DISCOVER, GLOBAL

Usage

```
set_requirement_property <req-id> <property-name> <property-values>
```

ReturnType

Nothing

Returns

no return value

Arguments

req-id Requirement ID

property-name Requirement property type See [Requirement Properties](#).

property-values Requirement property value

Example

```
set_requirement_property req1 MODULE_NAME altera_xcvr_native_s10_etile
```

Related Information

[add_requirement](#) on page 130

2.1.31. set_toolkit_property

Description

This command set set one of the toolkit's properties.

Availability

DISCOVER, GLOBAL

Usage

```
set_toolkit_property <prop> <value>
```

ReturnType

Nothing

Returns

no return value

Arguments

prop The name of the toolkit property. See [Toolkit Properties](#).

value The new value of the property.

Example

```
set_toolkit_property DISPLAY_NAME { My Toolkit }
```

Related Information

[get_toolkit_property](#) on page 140

2.1.32. stop_requested

Description

This command is used in defined callbacks to detect and respond to an early exit.

Altera Internal

This is for internal Altera use only.

Availability

AUTOSWEEP_QUALITY_METRIC, EYE_VIEWER

Usage

`stop_requested`

Return Type

boolean

Returns

No return value.

Arguments

no arguments

Example

```
set early_exit_status [ stop_requested ]
```

2.2. Properties

2.2.1. _hw.tcl Callbacks

Name	Description
ACTION_CALLBACK	A callback procedure that is called when an ACTION display item's action is performed.
AUTOSWEEP_CASE_VALIDITY_CALLBACK	A callback procedure that is called at the start of each autosweep scan iteration and takes in a set of parameters to be set to determine whether a given configuration case is valid by returning a 'true' or 'false' value.
AUTOSWEEP_FINALIZATION_CALLBACK	A callback procedure that is called at the end of an autosweep scan (not per autosweep scan case iteration).
AUTOSWEEP_INITIALIZATION_CALLBACK	A callback procedure that is called at the start of an autosweep scan (not per autosweep scan case iteration).
AUTOSWEEP_QUALITY_METRIC_CALLBACK	A callback procedure that is called in each autosweep scan iteration to update specified quality metric values.
CHANNEL_CALLBACK	A callback procedure that is called before the toolkit initialization stage to define associated toolkit channels.

Name	Description
CHART_CALLBACK	The update procedure for chartable parameters. A callback procedure that is called at a fixed time interval as specified by the user's update frequency. These callbacks execute sequentially and, when added, are scheduled to execute at the next available time slot.
ELABORATION_CALLBACK	A callback procedure that is called in response to parameter value changes.
EYE_VIEWER_CALLBACK	A callback procedure that is called to configure the Eye Viewer tool to display an eye based on a given set of eye data.
FINALIZATION_CALLBACK	A callback procedure that is called once when a toolkit is either closed by System Console or is no longer referenced in a System Console toolkit view, whichever occurs last.
INITIALIZATION_CALLBACK	A callback procedure that is called once when the toolkit is either first launched or referenced in a System Console toolkit view, whichever occurs first.
PARAMETER_UPDATE_CALLBACK	A callback procedure that is called when the associated parameter is changed in the parameter editor UI.
QUALITY_COMPARISON_CALLBACK	A callback procedure that is used for captured metric comparison to sort in provided result tables.
TIMED_CALLBACK	A callback procedure that is called at a fixed timed interval (in milliseconds). These callbacks execute sequentially and, when added, are scheduled to execute at the next available time slot.

2.2.2. `_hw.tcl` Channel Properties

Name	Description
SUPPORTS_EYE	Available for RECEIVER type channels. USED to specify whether the channel is configured to support an eye diagram.

2.2.3. `_hw.tcl` Channel Type

Name	Description
GLOBAL	A global channel.
OTHER	A channel that is neither a transceiver, transmitter, nor receiver channel.
RECEIVER	A transceiver transmitter channel.
TRANSMITTER	A transceiver receiver channel.

2.2.4. `_hw.tcl` Display Hint

Name	Description
BACKGROUND_COLOR_LIST	Used to configure the background color of table columns or LED icons.
BIT_WIDTH	Bit width of a number.
BOOLEAN	Integer value of either 0 or 1.
COLLAPSED	Indicates whether a group is collapsed when initially displayed.
COLUMN	If present on a group, the group should be displayed in a top-level column.

Name	Description
COLUMNS	Number of columns in text field, for example, "columns:N".
DIAL_MAX	Specifies the maximum value displayed by the dial.
DIAL_MIN	Specifies the minimum value displayed by the dial.
DIAL_TICK_SIZE	Specifies the distance between tick values.
EDITABLE	Indicates whether a list of strings allows free-form text entry (editable combo box).
FILE	Indicates that the string is an optional file path, for example, "file:jpg,png,gif".
FIXED_SIZE	Indicates a fixed size for a table or list.
FOREGROUND_COLOR_LIST	Used to configure the foreground color of table columns or LED icons.
GROUP_FINALIZE	Group finalizer to explicitly stop adding items to the current tab group or column group to enable adding multiple column groups to the same parent.
GROW	If set, the widget can grow when the component is resized.
HAS_LABEL	Indicates whether or the parameter has a label .
HEXADECIMAL	Indicates that the long integer is hexadecimal.
MONOSPACED_FONT	Display text with a fixed width font.
NON_BLOCKING	(Deprecated) Boolean to turn off non_blocking behavior.
RADIO	Indicates that the range displays as radio buttons.
READ_ONLY_TEXT	Indicates if a parameter is read-only. Read-only parameters are displayed as plain text labels.
ROWS	Number of rows in text field or visible rows in a table, for example, "rows:N".
SHOW_IN_MENU	Determines if the item is displayed in the action menu.
SLIDER	Range displays as slider.
TAB	If present for a group, the group displays in a tab.
TABLE	If present on a group, the group must contain all list-type parameters, which should be shown collectively in a single table.
TEXT	String is a text field with a limited character set, for example, "text:A-Za-z0-9_".
VISIBLE	(Deprecated) Boolean to determine that the display item will appear in the view.
WIDTH	Width of a table column.

2.2.5. _hw.tcl Display Item Properties

Name	Description
ACTION_PRIORITY	The priority of the associated action.
ADDITIONAL_ARGS	Additional arguments to the display item.
CHANNEL	Associated channel information for a given display item. This property is automatically set by the framework and may only be read using get_display_item_property. set_display_item_property may not be used to set this property.
DESCRIPTION	A description of this display item, which may be used as a tooltip.

Name	Description
DISPLAY_GROUP	Defines the hierarchical grouping of parameters and ACTION type display items.
DISPLAY_HINT	A hint that can affect how this display item will be displayed in the parameter editor.
DISPLAY_NAME	The label for the display item in the parameter editor. All display items and parameters in a given channel must have unique display names.
ENABLED	Whether this display item is enabled or not.
PATH	The path to a file. Only applies to display items of type ICON.
TEXT	Text associated with a display item. Only applies to display items of type TEXT.
TYPE	Type of the display item. This property is read-only and cannot be set with set_display_item_property.
VISIBLE	Whether this display item is visible or not.

2.2.6. _hw.tcl Display Item Type

Name	Description
ACTION	An action will be displayed as a button in the GUI. When the button is clicked, it will call the callback procedure. The button label will be the display item id.
DIAL	A dial widget.
GROUP	A group of display items. This group will be a child of the parent_group group. If the parent_group is an empty string, this will be a top-level group.
ICON	A .gif, .jpg, or .png file.
LED	An LED icon. Contains configurable display hints BACKGROUND_COLOR_LIST and FOREGROUND_COLOR_LIST. The background color defines the LED outline color, and the foreground color defines the LED color.
PARAMETER	A parameter in the instance.
TEXT	A block of text.

2.2.7. _hw.tcl Eye Properties

Name	Description
X_AXIS_OFFSET	This property shifts the x number axis by this property's value. Note: this offset is not an offset of the eye data array. It is the offset of the range of the axis. Also note: if a specific offset is desired that is independent of the step value, then use the equation: $Axis_Offset - (Axis_Offset \% Step_Value)$.
X_STEP	The step amount for x_values displayed on the x_axis number line. This is the distance between samples on the eye. If this value is x, then the distance between samples will be x.
X_UNIT	Label for the x_unit. Must be from a set defined in EEyeHorizontalUnits.
Y_AXIS_OFFSET	This property shifts the y number axis by this property's value. Note: this offset is not an offset of the eye data array. It is the offset of the range of the axis. Also note: if a specific offset is desired that is independent of the step value, then use the equation: $Axis_Offset - (Axis_Offset \% Step_Value)$.
Y_STEP	The step amount for y_values displayed on the y_axis number line. This is the distance between samples on the eye. If this value is x, then the distance between samples will be x.
Y_UNIT	Label for the y_unit. Must be from a set defined in EEyeVerticalUnits.

Name	Description
Z_LEGEND	This is a string that is used to define the z legend. The format is <code>{value0 label0} color0 {value1 label1} color1 {value2 label2}</code> . This allows for any colors, labels, or values to be used to build the legend.
Z_UNIT	Label for the z_unit. Can be an arbitrary string.
Z_UNIT_TYPE	This is the type of the values in the z_legend. Currently, must be an INTEGER or DOUBLE.

2.2.8. _hw.tcl Parameter Properties

Name	Description
AFFECTS_ELABORATION	Specifies whether the elaborate callback is triggered whenever the parameter's value has changed.
ALLOWED_RANGES	Indicates the range or ranges that the parameter value can have. For integers, the ALLOWED_RANGES property is a list of ranges that the parameter can take on, where each range is a single value, or a range of values defined by a start and end value separated by a colon, such as 11:15. This property can also specify legal values and display strings for integers, such as {0:None 1:Monophonic 2:Stereo 4:Quadrophonic} meaning 0, 1, 2, and 4 are the legal values. You can also assign display strings to be displayed in the parameter editor for string variables. For example, ALLOWED_RANGES {"dev1:Cyclone IV GX"dev2:Stratix V GT"}.
ALLOWS_AUTOSWEEP	This property, when set to true, marks a parameter as "autosweepable" and enables a parameter to be scanned/swept in the autosweep view. By default, this value is set to true. Parameters with this property set must define the ALLOWED_RANGES property.
ALLOWS_CHARTING	This property, when set to true, marks a parameter as chartable and enables it to be used in the dashboard view. By default, this value is set to true.
CHANNEL	Associated channel information for a given channel. This property is automatically set by the framework and may only be read using <code>get_parameter_property</code> . <code>set_parameter_property</code> may not be used to set this property.
DEFAULT_VALUE	The default value of the parameter.
DESCRIPTION	A short user-visible description of the parameter, suitable for a tooltip description in the parameter editor.
DISPLAY_GROUP	Defines the hierarchical grouping of parameters displayed in a "Select Parameter" menu. The hierarchy of these is represented by a "/" separated string. Parameters without a defined DISPLAY_GROUP are listed out as a flat list by their DISPLAY_NAME.
DISPLAY_HINT	Provides a hint about how to display a property.
DISPLAY_NAME	This is the GUI label that appears to the left of this parameter. All display items and parameters in a given channel must have unique display names.
ENABLED	When false, the parameter is disabled, meaning that it is displayed but greyed out, indicating that it is not editable on the parameter editor.
EYE_STATISTIC	Enables a parameter to be an eye statistic. Eye statistics are captured and displayed in the eye viewer's result table. By default, this property is disabled (false).
GROUP	Controls the layout of parameters in the GUI.

Name	Description
QUALITY_COMPARISON_CALLBACK	Defines how values of this parameter should be compared, i.e., what is considered as a better or worse value. This property takes in the name of a tcl procedure. When set, this property marks a parameter as a quality metric parameter.
TABLE_COLUMN_ORDER	Defines the default column order of parameters. This property takes in a display order value as its argument, which is an integer value greater than or equal to -1. The table column order values of 0 or greater are sorted and arranged in numerical order, starting with the smallest value on the left and larger values to the right. Parameters with matching column order values are then sorted alphabetically with respect to each other. By default, the column order value of a parameter is -1. When parameters with both order values of -1 and ≥ 0 are present in the status table, parameters with an order value of -1 are arranged and sorted alphabetically after all parameters with a non-default column order value.
TABLE_COLUMN_VISIBILITY	Indicates whether a parameter column is displayed as part of the status table at the launch of a toolkit. This property is set to true by default. To not display this parameter in the status table by default, configure this property to false.
TYPE	The type of this parameter.
UNITS	The unit used for this parameter.
VISIBLE	Indicates whether or not to display the parameter in the parameterization GUI.

2.2.9. _hw.tcl Parameter Type

Name	Description
BOOLEAN	A boolean parameter whose value can be true or false.
FLOAT	A signed 32-bit floating point parameter.
FLOAT_LIST	A parameter containing a list of signed 32-bit floating point values.
INTEGER	A signed 32-bit integer parameter.
INTEGER_LIST	A parameter containing a list of 32-bit integers.
LONG	A signed 64-bit integer parameter.
NATURAL	A 32-bit number that can contain values 0 to 2147483647 (0x7fffffff).
POSITIVE	A 32-bit number that can contain values 1 to 2147483647 (0x7fffffff).
STRING	A string parameter.
STRING_LIST	A parameter containing a list of strings.

2.2.10. _hw.tcl Requirement Properties

Name	Description
AGENT_TYPE_NAME	The service's debug typename.
DESIGN_ID	A string that identifies the system / example design.
HPATH	A hierarchical path used to specify the IP or Service within a design.
IPS	Specifies an IP requirement.

Name	Description
MODULE_NAME	Required IP module name.
OPTIONAL_IPS	Specifies an optional IP requirement.
OPTIONAL_SERVICES	Specifies an optional Service requirement.
RELATIVE_HPATH	The relative HPATH associated with the specified IP requirement. This is used to further pin-point the targeted service within a design.
SERVICES	Specifies a Service requirement.
TYPE	Defines the Service type.
VERSIONS	Required IP version(s).

2.2.11. _hw.tcl Requirement Type

Name	Description
IP	An IP requirement.
OPTIONAL_IP	An optional IP requirement.
OPTIONAL_SERVICE	An optional Service requirement.
SERVICE	A Service requirement.
SYSTEM	A System requirement.

2.2.12. _hw.tcl Toolkit Properties

Name	Description
AUTHOR	The author of the toolkit.
AUTOSWEEP_CASE_VALIDITY_CALLBACK	A callback procedure that is called at the start of each autosweep scan iteration and takes in a set of parameters to be set to determine whether a given configuration case is valid by returning a 'true' or 'false' value.
AUTOSWEEP_FINALIZATION_CALLBACK	A callback procedure that is called at the end of an autosweep scan (not per autosweep scan case iteration). This callback is used to configure the device at the end of a scan, e.g., restore the device to a previous state.
AUTOSWEEP_INITIALIZATION_CALLBACK	A callback procedure that is called at the start of an autosweep scan (not per autosweep scan case iteration). This callback is used to configure the device in preparation for an autosweep scan.
AUTOSWEEP_QUALITY_METRIC_CALLBACK	A callback procedure that is called in each autosweep scan iteration to update specified quality metric values. The argument of this callback procedure is a list of quality metric parameters that are expected to be updated after every scan. Note: quality metric parameters must have an associated quality comparison callback defined.
CHANNEL_CALLBACK	A callback procedure that is called to declare toolkit channels. This can be used to add channels with the "add_channel" command and can be used to set the properties of that channel using the "set_channel_property".
CHART_CALLBACK	The update procedure for chartable parameters. The argument of the callback procedure is a list of chartable parameters that are expected to be updated. This callback is called at a fixed

Name	Description
	time interval as specified by the user's update frequency. These callbacks execute sequentially and, when added, are scheduled to execute at the next available time slot.
DATASHEET_URL	The URL for the corresponding datasheet.
DESCRIPTION	Description of the toolkit.
DISPLAY_NAME	Name of the toolkit.
ELABORATION_CALLBACK	The name of the elaboration callback. When defined, the toolkit's elaboration callback is called to respond to parameter value changes, i.e., this callback may be used to push updated parameter values to hardware.
EYE_VIEWER_CALLBACK	A callback procedure that is called to configure the Eye Viewer tool to display an eye based on a given set of eye data. The argument of this callback is the channel that is going to be scanned. Use "set_eye_data" command to set the data used to create the eye diagram.
FINALIZATION_CALLBACK	When defined, this callback is executed once when a toolkit is either closed by System Console or is no longer referenced in a System Console toolkit view, whichever occurs last.
GROUP	The group that this toolkit belongs to.
INITIALIZATION_CALLBACK	When defined, this callback is executed once when the toolkit is either first launched or referenced in a System Console toolkit view, whichever occurs first.
INTERNAL	When set to true, this property causes the toolkit to only be displayed when using the "--debug" flag.
NAME	A unique identifier for the toolkit.
VERSION	Toolkit version, e.g., 1.0

2.3. System Console and Toolkit Tcl Command Reference Manual Revision History

Document Version	Intel® Quartus® Prime Version	Changes
2021.06.21	21.2	Initial release of the document.