Procedure 8

PTCS Running TCL Tests

Version 1.0

|  |  |
| --- | --- |
|  | |
| Author | Date |
| Owen Shriver | August 13, 2019 |
|  |  |
|  | |

|  |  |  |
| --- | --- | --- |
|  | | |
| Revision | Date | Description |
| 1.0 | 8/13/19 | Initial release |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | | |

[1.0 Introduction 2](#_Toc505773994)

[2.0 INSTRUCTIONS 3](#_Toc505773995)

2.1 Before creating a test……………………………………………………………………………….3

2.2 Creating the TCL files……………………………………………………………………………….3

2.3 Creating the JSON Configs…………………………………………………………………………3

2.4 Running a TCL Test Series………………………………………………………………………….3

PTCS Running TCL Tests

# Introduction

As of June 2019, the functionality exists to run tests through TCL. Although the test interface is the same, the internal procedure is somewhat different, and therefore some precautions must be taken to make sure that the TCL tests run correctly. This document details what the user must do to create and run a TCL test.

# Instructions

This section will explain what the user needs to do to create and run a TCL test.

## Before creating a test

The hardware that the test is to be run on must have a Vivado project associated with it. Furthermore, the user must have access to the Vivado TCL shell. The location of the TCL shell should be listed in the CONSTANTS file; the ability to edit this location through the GUI is not yet implemented.

## Creating the TCL files

The TCL test system concatenates a series of partial TCL scripts into a single test, so rather than creating a single script that does everything necessary, the user should break a script up into modular pieces that can be assembled in any order—for instance, it is recommended to have separate TCL files for opening and closing the associated Vivado project. In addition, the following conventions should be used to ensure that the script can read and store data correctly:

* All test parameters must be set using the TCL “set” command on the first few lines of the script, before any other commands or blank lines.
* The variable “$output” must be used to represent the directory where test results are to be saved to.

## Creating the JSON Configs

Each TCL file needs an associated JSON config. The JSON config must have the following attributes:

* name: “Test Name (Tcl)”, where Test Name is the name of the procedure.
* tcl: the name of the TCL file associated with it.
* experiment: any other scripts associated with this procedure, using the normal experiment syntax. The order must be greater than 1.
* data: a dictionary containing all the parameters used in the test and their default values.

## Running a TCL Test Series

To run a TCL test series, all that is needed is to add the TCL tests to the queue sequentially. PTCS will convert them into a single test called “Tcl Experiment” and their results will be found in subdirectories within a single results directory. As an example, an Eyescan can be run with the following test series:

* Initialize VCU 108 (Tcl)
* Program IBERT (Tcl)
* Eyescan (Tcl)
* Close VCU 108 (Tcl)