TDACS Test Tables and Procedures

Version 4

February 14, 2024

Developed by Transportation Laboratories

Version History

Version	Date	Revision Description
1	1/25/2016	Initial publication
2	8/23/2018	Format with SGS brand
3	4/3/2020	Retrofit to template
4	2/14/2024	Rebrand to TRP Laboratories

Document Conventions

This document uses the following typographic and syntax conventions.

- Commands, command options, file names or any user-entered input appear in Courier type. Variables appear in Courier italic type.

  Example: Select the `cmdapp-`*`relVersion-buildVersion`*`.zip` file….

- User interface elements, such as field names, button names, menus, menu commands, and items in clickable dropdown lists, appear in Arial bold type.

  Example: **Type**: Click **Select Type** to display drop-down menu options.

- Cross-references are designated in Arial italics.

  Example: Refer to *Figure 1*…

- Click intra-document cross-references and page references to display the stated destination.

  Example: Refer to *Section 1* Overview on page 1.

  The clickable cross-references in the preceding example are *1*, *Overview*, and on page 1.

CyFlex Documentation

CyFlex documentation is available at https://cyflex.com/. View **Help & Docs** topics or use the **Search** facility to find topics of interest.

Table of Contents

LIST OF TABLES

1 Overview

1.1 Test Table

A test table is a matrix whose rows represent test steps and columns represent parameters that are controlled in each step. The row is processed in sequential order.

Some of the parameters (data in columns) are used by the `gp_test` script. Other parameters are just passed through to other services that act on them.

1.2 Test Procedure

If a test procedure that manages test tables is well structured, it can be used for a great many different actual tests that all share the same pattern. The structure of the test is normalized (i.e. does not refer to specific absolute values, but instead uses relative values, e.g., percent rated load vs. an actual load value as a target or other value, then the test tables themselves become reusable across a wider set of products. This can increase the efficiency and quality of testing as it allows for best practices to be reused without adaptation to specific product ratings.

1.3 Conventions

There are many ways to structure `gp_test` procedures and test tables to work with one another. A test table with a given set of parameters may require a very specific test procedure to operate on it to get the desired results. In order to help keep the association between test tables and procedures straight, a convention is called for. This convention is not strictly enforced. It requires users to be aware of and use the convention. The convention for TDACS is that `gp_test` procedures that operate on tables are named `gp_table_<type>`. The test tables used by this procedure are all located in the directory `/specs/tables/<type>/`.

For example, two procedures referenced in examples in this document, `gp_table_simple1` and `gp_table_simple2`, would have test tables located in the directories: `/specs/table/simple1` and `/specs/table/simple2` respectively. Each directory could hold as many tables as were useful to run with each type of test.

2 Simple Test Table Format

Table 1: gp_table_simple1 Format

Column Name	Purpose	Units	Notes
Used in gp_simple1 test script			
`TestStep`	index into the test	None	starts at 1, increments by 1
`StepDwell`	amount of time to spend at step	Minutes	0 = indefinite
Passed through to other services			
`LoadStep`	load step to select	None	a value from 1 to 5, typically 1 = 0 % load, 5 = 100% load
`LogActive`	indicates that logging should occur in step	Boolean	ON = Active
`LogPeriod`	specifies rate at which logging should occur	sec	minimum = 0.01, maximum = 10
`TestNotes`	annotates test setup and provides feedback to user while test is in progress	None	80 characters max

3 Test Table Examples

The following tests all use the basic test table procedure, but the data in the table drives different results even though the `genset` is sent through basically the same commands. A single test procedure can be used to good effect to achieve required results through changes in test table data.

3.1 Simple Test Table

This test will apply load step changes and log data at a fixed rate during the entire test. This test captures both transient and steady state data.

Table 2: Simple Test Table

TestStep	StepDwell	LoadStep	LogActive	LogRate	TestNotes
None	Min	None	None	Hz	None
1	3	5	OFF	1	Warmup, no logging
2	5	1	ON	1	No load, logging starts
3	5	2	ON	1	25% load
4	5	3	ON	1	50% load
5	5	4	ON	1	75% load
6	5	5	ON	1	100% load
7	5	4	ON	1	75% load
8	5	3	ON	1	50% load
9	5	2	ON	1	25% load
10	5	1	ON	1	0% load

Test tables offers a feature that can avoid redundant data being in the table which eases changing the test. This feature enables use of a dash in place of a value to use the value in the previous step for the current step. Therefore *Table 2* can also be represented as in *Table 3*:

Table 3: Simple Test Table with Reduced Redundancies

TestStep	StepDwell	LoadStep	LogActive	LogRate	TestNotes
None	Min	None	None	Hz	None
1	3	5	OFF	1	Warmup, no logging
2	5	1	ON	1	No load, logging starts
3	-	2	-	-	25% load
4	-	3	-	-	50% load
5	-	4	-	-	75% load
6	-	5	-	-	100% load

TestStep	StepDwell	LoadStep	LogActive	LogRate	TestNotes
7	-	4	-	-	75% load
8	-	3	-	-	50% load
9	-	2	-	-	25% load
10	-	1	-	-	0% load

Table 4 shows the advantage of doing this as it allows the dwell time at each step to be changed to 15 minutes and the logging rate to 10 seconds by only changing row 2. The rows after 2 all have a dash for `StepDwell` and `LogPeriod`, and hence use the most recent values specified, i.e. those in step 2.

Table 4: Reduced Redundancy Advantages

TestStep	StepDwell	LoadStep	LogActive	LogRate	TestNotes
None	Min	None	None	Hz	None
1	3	5	OFF	1	Warmup, no logging
2	15	1	ON	0.01	No load, logging starts
3	-	2	-	-	25% load
4	-	3	-	-	50% load
5	-	4	-	-	75% load
6	-	5	-	-	100% load
7	-	4	-	-	75% load
8	-	3	-	-	50% load
9	-	2	-	-	25% load
10	-	1	-	-	0% load

3.2 Evolved Simple Test

Assume the desire to modify the test to only capture steady state data. This can be done by controlling logging during the steps. The test in *Table 5* introduces a new step between each steady state point to perform the load change during which time logging is disabled.

Table 5: Capture Steady State Data Only

TestStep	StepDwell	LoadStep	LogActive	LogRate	TestNotes
None	Min	None	None	Hz	None
1	3	5	OFF	1	Warmup, no logging
2	0.25	1	OFF	1	Transition to no load, no logging
3	15	-	ON	1	No load, logging on
4	0.25	2	OFF	-	Transition to 25% load, no logging
5	15	-	ON	-	25% load, logging on
6	0.25	3	OFF	-	Transition to 50% load, no logging
7	15	-	ON	-	50% load, logging on
8	0.25	4	OFF	-	Transition to 75% load, no logging
9	15	-	ON	-	75% load, logging on
10	0.25	5	OFF	-	Transition to 100% load, no logging
11	15	-	ON	-	100% load, logging on
12	0.25	4	OFF	-	Transition to 75% load, no logging
13	15	-	ON	-	75% load, logging on
14	0.25	3	OFF	-	Transition to 50% load, no logging
15	15	-	ON	-	50% load, logging on
16	0.25	2	OFF	-	Transition to 25% load, no logging
17	15	-	ON	-	25% load, logging on
18	0.25	1	OFF	-	Transition to no load, no logging
19	15	-	ON	-	0% load, logging on

3.3 Advanced Test

Assuming that controlling logging both during transients and steady state is a common occurrence, it is possible to have a more advanced test procedure to support that situation. This procedure could have knowledge of a transient and steady state step time and transient and steady state logging control and rates. The following test table format could support this, however a different test procedure, `gp_table_simple2`, is needed to support it due to the added features. Compare the test in *Table 6* to the preceding test to view implementation of the same features in a more compact and usable form.

Table 6: Advanced Test

TestStep	LoadStep	XientDwell	XientLogActive	XientLogRate	SSDwell	SSLogActive	SSLogRate	TestNotes
None	None	Sec	None	Sec	Min	None	Min	None
1	5	30	OFF	1	5	OFF	0.1	Warmup, no logging
2	1	-	OFF	-	-	ON	-	Transition to no load, log steady state
3	2	-	ON	-	-	-	-	Transition to 25% load, log transient and steady state
4	3	-	-	-	-	-	-	Transition to 50% load, log transient and steady state
5	4	-	-	-	-	-	-	Transition to 75% load, log transient and steady state

TestStep	LoadStep	XientDwell	XientLogActive	XientLogRate	SSDwell	SSLogActive	SSLogRate	TestNotes
6	5	-	-	-	-	-	-	Transition to 100% load, log transient and steady state
7	4	-	-	-	-	-	-	Transition to 75% load, log transient and steady state
8	3	-	-	-	-	-	-	Transition to 50% load, log transient and steady state
9	2	-	-	-	-	-	-	Transition to 25% load, log transient and steady state