Version: 1.0
Status: Draft

SunSpec Modbus Conformance Test Procedures Results Reporting

SunSpec Specification

Abstract

This document specifies the contents and format of SunSpec Modbus Test Procedures test results report format. SunSpec Modbus certification testing partners are required to deliver test results to SunSpec in this format.

License Agreement and Copyright Notice

Prepared by the SunSpec Alliance
4040 Moorpark Avenue, Suite 110
San Jose, CA 95117

Website: https://sunspec.org
Email: info@sunspec.org

Revision History

Version	Date	Comments
1.0	10-06-2021	Candidate final release

About the SunSpec Alliance

The SunSpec Alliance is a trade alliance of developers, manufacturers, operators, and service providers together pursuing open information standards for the distributed energy industry. SunSpec standards address most operational aspects of PV, storage, and other distributed energy power plants on the smart grid, including residential, commercial, and utility-scale systems, thus reducing cost, promoting innovation, and accelerating industry growth.

Over 100 organizations are members of the SunSpec Alliance, including global leaders from Asia, Europe, and North America. Membership is open to corporations, non-profits, and individuals. For more information about the SunSpec Alliance, or to download SunSpec specifications at no charge, visit sunspec.org.

Contents

1 Introduction

This document specifies the content and format of the SunSpec Modbus Test Results Report ("Test Results Report") generated by testing tools that implement the SunSpec Modbus Conformance Test Procedures. The Test Results Report is the document that SunSpec uses to determine the compliance of a DER device to the SunSpec Modbus standard and to grant a SunSpec Certified mark to the device manufacturer.

This document is subdivided into three main sections:

- Chapter 2, General Testing Considerations: information you need to set up and run a valid test.
- Chapter 3, Test Results Report (TRR) Format Specification: the specific data needed by SunSpec to make a certification evaluation of the device that was tested. These reports are posted on the SunSpec web site and made available to the public.
- Chapter 4, Detailed Test Logs: the data used by SunSpec to verify successful execution of each test procedure. This data is archived by SunSpec and used only for confirming test results. It is not shared with the public.

SunSpec may update this document on a periodic basis. To make sure you have the most current version, go to https://sunspec.org/.

2 General Testing Considerations

The following general testing considerations must be observed when performing tests and generating a Test Results Report (TRR).

2.1 Equipment Version

All tests must be completed without altering the software or hardware during the testing process. If any changes are made to the software or hardware, the testing process must be restarted.

2.2 Configuration

The complete equipment configuration must be documented such that the configuration can be applied to an unconfigured production unit for test replication.

2.2.1 Device

The device under test must be configured before the test procedures are started and the configuration may not be changed during the testing process. If any changes are made to the client configuration, the testing process must be restarted.

3 Test Results Report (TRR) Format Specification

TRR's submitted to SunSpec for the SunSpec Modbus Conformance Test Procedures are comprised of two elements: Summary Test Results and a Detailed Test Log for all tests.

3.1 Summary Test Results

The Summary Test Results are contained in a single document containing comma separated values (CSV) where each row consists of an information key and a value. The ordering of the key/value pairs is unimportant although the example below lays out the report in a manner that can be easily read and is thus strongly preferred.

- The values for many of the keys are enumerated. Enumerated values are defined in the Description column as shown in the example.
- For values that are not enumerated, the following rules apply:
    - All values must be strings
    - Keys must match the documented key values
    - Values that include characters that have a specific meaning in the CSV format must be placed in quotes.

Commented [AS1]: What are these characters? The document should provide a list of these characters.

3.1.1 Key/Value Pair Descriptions for Summary Test Results

The following table describes the key/value pair definitions for test results:

Key	**Description**	**Value Field Type**
Certificate Type	The name of the SunSpec certificate type.	Enumerated list (IEEE1815/AN2018, IEEE 2030.5/CSIP, **SunSpec Modbus**, SunSpec RSD)
Certificate Number	Number of the SunSpec Certificate. This number is assigned by SunSpec.	String
Company Name	Official name of the company that the SunSpec Certificate is issued to.	String
Company Address	Official address of the company that the SunSpec Certificate is issued to.	String
Company City	City of the company that the SunSpec Certificate is issued to.	String
Company State	The U.S. State of the company that the SunSpec Certificate is issued to (if the company is in the U.S.).	Enumerated list (see Appendix A). Mark "Not Applicable" if company is not located in U.S.
Company Province	The province of the company that the SunSpec Certificate is issued to. Only applicable to companies not located in the U.S.	String. Mark "Not Applicable if company is located in the U.S.

Company Country	Country of the company that the SunSpec Certificate is issued to.	Enumerated list. See Appendix A for details.
Company Postal Code	Postal code of the company that the SunSpec Certificate is issued to.	String
Date Issued	Date the SunSpec Certificate was issued.	Date: yyyy-mm-dd
Test Laboratory	The SunSpec Authorized Test Laboratory that performed the Test.	Enumerated list. See Appendix A for details.
Supervising Test Engineer	The name(s) of the employee of the SunSpec Authorized Test Laboratory who performed the Tests.	String
Certificate Signer Name	The name of the person at SunSpec Alliance who approved the SunSpec Certificate.	String
Software Name <n (Software)>	The file name of the SunSpec Certified Software including file type suffix. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String
Software Version <n (Software)>	The version number of the SunSpec Certified software. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String
Software Checksum <n (Software)>	Checksum of the SunSpec Certified software. Multiple fields of this type may be present if the software is structured as a set of software modules. Individual fields are distinguished by adding a number at the end of the key label.	String
Operating System <n (OS)>	The operating system of the SunSpec Certified software	String
Operating System Version <n (OS)>	Major version number plus first digit of minor version number of the Operating System that supports SunSpec Certified software, expressed as "X.X". Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String
Software Operating Environment	The compute environment where the Certified Software operates.	Enumerated list. Choices are "Cloud" or "Hardware Device"
Protocol Implementation Conformance Statement	The Protocol Implementation Conformance Statement of the Certified Software.	String (properly formed URL indicating the location of the document.

		SunSpec is responsible for filling in this field.)
Cloud Provider	The name of cloud platform provider where the Certified Software operates.	String. Mark "Not Applicable" if software is hosted in a device.
Cloud Provider Version	The version of cloud platform where the Certified Software operates.	String. Mark "Not Applicable" if software is hosted in a device.
Product Manufacturer <n (Prod Mfr)>	The manufacturer name of the Product that incorporates the SunSpec Certified software and the Hardware. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label. The manufacturer name may be the same as the Company Name or may be a third-party Original Equipment Manufacturer (OEM)	String. Mark "Not Applicable" is software is hosted in cloud.
Product Model <n (Prod Model)>	The model name of the Product that incorporates the SunSpec Certified software and the Hardware. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String. Mark "Not Applicable" is software is hosted in cloud.
Hardware Manufacturer <n (HW Mfr)>	The manufacturer name of the Hardware device that supports the SunSpec Certified software. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String. Mark "Not Applicable" is software is hosted in cloud.
Hardware Model <n (HW Model)>	The model name of the Hardware device that supports the SunSpec Certified software. Multiple fields of this type may be present. Individual fields are distinguished by adding a number at the end of the key label.	String. Mark "Not Applicable" is software is hosted in cloud.
Test Completion Date	The date when the test was completed.	Date: yyyy-mm-dd
Test Description	The description of the test performed.	Enumerated list. See Appendix A for details.
Additional Test Comments	Comments associated with most recent Test as indicated by Test Date.	String
Test <Test ID>	The result of the test performed. Multiple fields of this type may be present. Individual fields are distinguished by adding a Test ID at the end of the key label.	Enumerated list. Choices are "PASS" or "FAIL" or “NOT SUPPORTED”

3.1.2 Example Test Results Report

The following table is an example of a Test Results Report generated by a SunSpec Authorized Test lab following the evaluation of a SunSpec Modbus compliant product:

SunSpec Modbus Test Example Report

Key	**Value**
Certificate Type	SunSpec Modbus
Certificate Type Version	2021
Certificate Number	ICS183884882
Company Name	ACME Incorporated
Company Address	569 Regents Park Avenue
Company City	Boston
Company State/Province	MA
Company Country	USA
Company Postal Code	02123
Date Issued	01/12/2019
Test Laboratory	Official Lab LLC
Supervising Test Engineer	Peter A Smith
Certificate Signer Name	Daniela T Jones
Software Name 1	ABC2030client.exe
Software Version 1	11.01
Software Checksum 1	423EC0E4
Operating System 1	Ubuntu
Operating System Version 1	18.04
Software Operating Environment	Device
Protocol Implementation Conformance Statement	https://pics.sunspec.org/abccopics12.xlsx
Cloud Provider	Not Applicable
Cloud Provider Version	Not Applicable
Hardware Manufacturer 1	Maxima
Hardware Model 1	DER MB31-M-SC
Test Completion Date	10/03/2021
Test Description	Test performed in compliance with SunSpec Modbus test procedures.
Additional Test Comments	

Test DEV-1	PASS
Test DEV-2	PASS
Test MOD-1.1	PASS
Test MOD-2.1	PASS
Test MOD-3.1	PASS
Test MOD-1.701	PASS
Test MOD-2.701	PASS
Test MOD-3.701	PASS
Test MOD-1.705	PASS
Test MOD-2.705	PASS
Test MOD-3.705	PASS
Test CRV-1.705	PASS
Test CRV-2.705	PASS
Test CRV-3.705	PASS
Test REV-1.705	PASS
Test REV-2.705	PASS
Test REV-3.705	PASS
Test TCP-1	PASS
Test TCP-2	PASS
Test TCP-3	PASS

4 Detailed Test Logs

The detailed test logs consist of a log of all Modbus messages exchanged for each of the tests performed. For implementations using Modbus TCP, connection information is also logged. The detailed test logs may be supplied in a single document or multiple documents in the format outlined below.

The detailed test logs must contain all Modbus messages in unencrypted form that were transferred as part of the test. The logs must also contain connection information for Modbus TCP implementations. The detailed test logs are encoded in JSON format in three different JSON objects: Test Log, Test Logs, and Log Entry. The contents of each JSON object are described below.

4.1.1 Test Log Object

Each test log is represented as a test log object that contains an array of test names the log applies to, an optional context id and the log consisting of an array of log entries that contain either Modbus messages or connection information.

The *tests* array is an array of strings that corresponds to the name specified in the test procedure such as “MOD-1” that the test log applies to. There may be one or more test procedures specified in the array if the log satisfies the requirements of more than one test procedure.

The context id is a string used as an id to provide an association for multiple tests if they are archived as separate items. The context id is optional.

The *entries* array contains the Modbus messages associated with the test. If the test is for a Modbus TCP implementation, connection information must also be provided. The format of a message is specified in the Message Object section below.

Test Log Object contents:

JSON Element	Description
tests	Array of test names
cid	Context id
entries	Array of log entries in specified format

Test Log Object JSON format:

```
{“tests”: [<array of test names>],
 “cid”: <context id>,
 “entries”: [<array of log entries containing Modbus messages
and connection information>]
}
```

4.1.2 Log Entry Object

A log entry object contains the contents of a single Modbus message or Modbus TCP connection information.

Log entries may contain the following elements:

JSON Element	Description
time	Time stamp
type	Message type [“req”, “resp”, “conn”, “disc”]
msg	Modbus message as an ascii hex string
cid	Connection ID
ipaddr	Connection IP address
ipport	Connection IP port

The “time” element is a time stamp that is a number containing a value in seconds that represents the timing of the message. The time may represent sub-second timing using a decimal notation. The time must have second accuracy and ideally sub-second accuracy.

The “type” element is a string (“req”, “resp”, “conn”, “disc”) that indicates whether the message is a Modbus request, a Modbus response, a Modbus TCP connection initiation, or a Modbus TCP connection termination.

The “msg” element is an ascii hex string that contains the contents of a single Modbus request or response. The message must contain the complete Modbus message content as hexadecimal values with the binary representation converted to ascii hex values.

The “cid” element is a Modbus TCP connection identifier. A connection identifier that is unique within the log must be created for each new Modbus TCP connection during the test. The new “cid” is an element in the “conn” type entry. The “cid” element for the associated connection must be included in all Modbus request and response messages for Modbus TCP testing.

The “ipaddr” element is a string that contains the IP address of the Modbus TCP connection request.

The “ipport” element is a string that contains the IP port of the Modbus TCP connection request.

The following log entry requirements apply:

1. All log entries must contain “time” and “type” elements.
2. All log entries with a type of “req” or “resp” must contain a “msg” element.
3. All Modbus TCP log entries must contain a “cid” element.
4. All log entries with a type of “conn” must contain “ipaddr”, “ipport”, and “cid” elements.

Message Object JSON format:

```
{“time”: <time stamp>,
 “type”: <log entry type (“req”, “resp”, “conn”, “disc”)>,
 “msg”: <message content as a hex string>,
 “cid”: <connection ID (Modbus TCP only)>,
 “ipaddr”: <IP address and port (Modbus TCP only)>,
 “ipport”: <IP address and port (Modbus TCP only)>,
}
```

Log entries that contain Modbus TCP connection information have the following contents:

JSON Element	Description
time	Time stamp
type	Message type [“conn”, “disc”]
ipaddr	IP address
ipport	IP port
cid	Connection ID that is unique within the log

Example connection information log entries in JSON format:

```
{“time”: 1539663163.977,
 “type”: “conn”,
 “ipaddr”: “192.168.0.10”,
 “ipport”: 502,
 “cid”: 1234
}

{“time”: 1539663169.214,
 “type”: “disc”,
 “cid”: 1234
}
```

Log entries that contain Modbus message information have the following contents:

JSON Element	Description
time	Time stamp
type	Message type [“req”, “resp”]
msg	Modbus message as an ascii hex string
cid	Modbus TCP connection ID

Example Modbus message log entries in JSON format:

```
{“time”: 1539663164.152,
 “type”: “req”,
 “msg”: “00000000000601039F880001”,
 “cid”: 1234
}

{“time”: 1539663164.347,
 “type”: “resp”,
 “msg”: “00000000000501030202C6”,
 “cid”: 1234
}
```

4.1.3 Test Logs Object

Multiple test logs can be placed in a test logs object along with an optional context id.

The test log array contains test log objects.

The context id is a string used as an id to provide an association for multiple tests if they are archived as separate items. The context id is optional.

Tests Logs Object contents:

JSON Element	Description
logs	Array of test log objects
cid	Context id

Tests Logs Object JSON format:

```
{“logs”: [<array of test log objects>],
 “cid”: <context id>
}
```