TE001 Test Report on the Module codecs_lib.cobs

Conventions

Each test is defined following the same format. Each test receives a unique test identifier and a reference to the ID(s) of the requirements it covers (if applicable). The goal of the test is described to clarify what is to be tested. The test steps are described in brief but clear instructions. For each test it is defined what the expected results are for the test to pass. Finally, the test result is given, this can be only pass or fail.

The test format is as follows:

Test Identifier: TEST-[I/A/D/T]-XYZ

Requirement ID(s): REQ-uvw-xyz

Verification method: I/A/D/T

Test goal: Description of what is to be tested

Expected result: What test result is expected for the test to pass

Test steps: Step by step instructions on how to perform the test

Test result: PASS/FAIL

The test ID starts with the fixed prefix 'TEST'. The prefix is followed by a single letter, which defines the test type / verification method. The last part of the ID is a 3-digits *hexadecimal* number (0..9|A..F), with the first digit identifing the module, the second digit identifing a class / function, and the last digit - the test ordering number for this object. E.g. 'TEST-T-112'. Each test type has its own counter, thus 'TEST-T-112' and 'TEST-A-112' tests are different entities, but they refer to the same object (class or function) within the same module.

The verification method for a requirement is given by a single letter according to the table below:

Term	Definition
Inspection (I)	Control or visual verification
Analysis (A)	Verification based upon analytical evidences
Test (T)	Verification of quantitative characteristics with quantitative measurement
Demonstration (D)	Verification of operational characteristics without quantitative measurement

Test definitions (Test)

Test Identifier: TEST-T-100

Requirement ID(s): REQ-FUN-101, REQ-FUN-110

Verification method: T

Test goal: Test that an arbitrary byte string containg any amount of the zero characters ('\x00') and the non-zero characters sub-strings of the length less than 254 chararacter as well as more than or equal to 254 characters are properly COBS encoded.

Expected result: All string comparison pass, no exception is raised.

Test steps: Pass each of the original example byte strings into the encoding method and compare the returned byte string with the expected encoded counter part. Each time the string must be equal. **N.B.** implemented as a test case in the test suit module codecs_lib.tests.ut001_cobs.py

Test result: PASS

Test Identifier: TEST-T-101

Requirement ID(s): REQ-FUN-101, REQ-FUN-120

Verification method: T

Test goal: Test that the properly COBS encoded strings (no zero characters '\x00') are decoded as defined by the algorithm.

Expected result: All string comparison pass, no exception is raised.

Test steps: Pass each of the encoded example byte strings into the decoding method and compare the returned byte string with the corresponding original counter part. Each time the string must be equal. **N.B.** implemented as a test case in the test suit module codecs_lib.tests.ut001_cobs.py

Test result: PASS

Test Identifier: TEST-T-120

Requirement ID(s): REQ-FUN-120

Verification method: T

Test goal: Tests that the zero characters ('\x00') added to a properly COBS encoded byte string into the leading / tailing positions are ignored.

Expected result: The example encoded strings with added leading and tailing zero characters are properly decoded into their original counterparts without any exception being raised.

Test steps: For each of the proper test examples taken from the algorithm description add an arbitrary but not zero amount (1+) of the leading and tailing characters. Pass the resulting strings to the decoding method. Compare the results with the expected original counterparts. Repeat several times with each encoded - original pair of the examples. **N.B.** implemented as a test case in the test suit module codecs_lib.tests.ut001_cobs.py.

Test result: PASS

Test Identifier: TEST-T-102

Requirement ID(s): REQ-AWM-100

Verification method: T

Test goal: Only **bytearray** or **bytestring** instances are allowed as an input for the encoding en decoding functions / methods.

Expected result: TypeError (sub-class of) exception is raised during the respective unit test.

Test steps: Pass different data types, including Unicode strings, but excepting **bytearray** or **bytestring** instances in the both methods / functions. Each time the **TypeError** or its sub-class must be raised. **N.B.** implemented as a test case in the test suit module codecs_lib.tests.ut001_cobs.py

Test result: PASS

Test Identifier: TEST-T-121

Requirement ID(s): REQ-AWM-120

Verification method: T

Test goal: Test that the function / method for decoding a string raises an exception if it contains at least one zero character ('\x00') not in the leading / tailing position.

Expected result: ValueError (sub-class of) exception is raised during the respective unit test.

Test steps: For each of the proper test examples taken from the algorithm description insert a zero character into an arbitrary position but not as the first of last character. Pass the resulting strings to the decoding method. Each time the **ValueError** or its sub-class must be raised. Repeat several times with each encoded - original pair of the examples. **N.B.** implemented as a test case in the test suit module codecs_lib.tests.ut001_cobs.py

Test result: PASS

Test definitions (Analysis)

Test Identifier: TEST-A-100

Requirement ID(s): REQ-FUN-100

Verification method: A

Test goal: Test that the module implement all of the required functionality and it performs according to the COBS algorithm specifications.

Expected result: All of the unit tests defined by the test cases TEST-T-100, TEST-T-101, TEST-T-102, TEST-T-120 and TEST-T-121 must pass.

Test steps: Run the test suit module codecs_lib.tests.ut001_cobs.py

Test result: PASS

Traceability

For traceability the relation between tests and requirements is summarized in the table below:

Requirement ID	Covered in test(s)	Verified [YES/NO])
REQ-FUN-100	TEST-A-100	YES
REQ-FUN-101	TEST-T-100, TEST-T-101	YES
REQ-FUN-110	TEST-T-100	YES
REQ-FUN-120	TEST-T-101, TEST-T-120	YES
REQ-AWM-100	TEST-T-102	YES
REQ-AWM-120	TEST-T-121	YES

Software ready for production [YES/NO]	Rationale
YES	All tests are passed