

CUnit - A Unit testing framework for C.

<http://cunit.sourceforge.net/>

Automated Test Run Results

Running Suite TC01 - Initialize a PCAN Channel

Running test TC01.1 - Initialize PCAN-USB interface with default parameters ...	Passed
Running test TC01.2 - Initialize PCAN-USB interface a second time ...	Passed
Running test TC01.3 - Initialize PCAN-USB interface a second time after uninitializing ...	Passed
Running test #TC01.4 - Initialize PCAN-USB interface when interface is not present ...	Passed
Running test #TC01.5 - Initialize PCAN-USB interface when interface is used by another process ...	Passed
Running test TC01.6 - Initialize PCAN-USB interface with parameter 'Channel == PCAN_NONEBUS' ...	Passed
Running test TC01.7 - Initialize PCAN-USB interface with parameter 'Channel == PCAN_USBBUS1 - 1' ...	Passed
Running test TC01.8 - Initialize PCAN-USB interface with parameter 'Channel == PCAN_USBBUS8 + 1' ...	Passed
Running test TC01.9 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_1M' ...	Passed
Running test TC01.10 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_500K' ...	Passed
Running test TC01.11 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_250K' ...	Passed
Running test TC01.12 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_125K' ...	Passed
Running test TC01.13 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_100K' ...	Passed
Running test TC01.14 - Initialize PCAN-USB interface with parameter 'Btr0Btr1 == PCAN_BAUD_50K' ...	Passed
	Passed

Running test TC01.15 - Initialize PCAN-
USB interface with parameter 'Btr0Btr1 ==
PCAN_BAUD_20K' ...

Running test TC01.16 - Initialize PCAN-
USB interface with parameter 'Btr0Btr1 == Passed
PCAN_BAUD_10K' ...

Running test TC01.17 - Initialize PCAN-
USB interface with parameter 'Btr0Btr1 == Passed
PCAN_BAUD_5K' ...

Running test TC01.18 - Initialize PCAN-
USB interface with parameter 'Btr0Btr1 == Passed
0x0000' ...

Running test TC01.19 - Initialize PCAN-
USB interface with parameter 'Btr0Btr1 == Passed
0xFFFF' ...

Running test #TC01.20 - Initialize PCAN-
USB interface with different baud rate than Passed
on real CAN ...

Running test #TC01.21 - Initialize PCAN- Passed
USB interface after hot-plugging ...

Running test #TC01.22 - Initialize PCAN-
USB interface when interface is not Passed
connected to the bus anymore ...

Running Suite TC02 - Uninitialize a PCAN Channel

Running test TC02.1 - Uninitialize PCAN-
USB interface running with default Passed
parameters ...

Running test TC02.2 - Uninitialize PCAN-
USB interface when not initialized Passed
before ...

Running test TC02.3 - Uninitialize PCAN- Passed
USB interface after reinitializing ...

Running test #TC02.4 - Uninitialize
PCAN-USB interface when interface is not Passed
present anymore ...

Running test #TC02.5 - Uninitialize
PCAN-USB interface when interface is Passed
used by another process ...

Running test TC02.6 - Uninitialize PCAN-
USB interface with parameter 'Channel == Passed
PCAN_NONEBUS' after initializing ...

Running test TC02.7 - Uninitialize PCAN-
USB interface with an invalid value for Passed
parameter 'Channel' after initializing ...

Running test #TC02.8 - Uninitialize
PCAN-USB interface when interface is not Passed
connected to the bus anymore ...

Running Suite TC03 - Reset the receive and transmit queues

Passed

Running test TC03.1 - Reset queues while PCAN-USB interface is running with default parameters ...

Running test TC03.2 - Reset queues when PCAN-USB interface is not initialized before ... Passed

Running test TC03.3 - Reset queue with an invalid value for parameter 'Channel' ... Passed

Running test TC03.5 - Reset queues and check if the receive queue is empty ... Passed

Running Suite TC04 - Get the current BUS status

Running test TC04.1 - Get BUS status while PCAN-USB interface is running with default parameters ... Passed

Running test TC04.2 - Get BUS status when PCAN-USB interface is not initialized before ... Passed

Running test TC04.3 - Get BUS status with an invalid value for parameter 'Channel' ... Passed

Running test #TC04.4 - Get BUS status when PCAN-USB interface is not present anymore ... Passed

Running test #TC04.5 - Get BUS status while errors on CAN bus are present ... Passed

Running test #TC04.7 - Get BUS status when PCAN-USB interface is not connected to the bus anymore ... Passed

Running Suite TC05 - Read a CAN message from the receive queue

Running test TC05.1 - Read a CAN message while PCAN-USB interface is running with default parameters ... Passed

Running test TC05.2 - Read a CAN message when PCAN-USB interface is not initialized before ... Passed

Running test TC05.3 - Read a CAN message with an invalid value for parameter 'Channel' ... Passed

Running test TC05.4 - Read a CAN message with illegal parameter 'MessageBuffer == NULL' ... Passed

Running test TC05.5 - Read CAN messages from the receive queue and check for amount and order ... Passed

Running test #TC05.6 - Read CAN messages from the receive queue - long term stress test ... Passed

Running test #TC05.7 - Read CAN messages from the receive queue - timestamp accuracy test ... Passed

Running test #TC05.8 - Read CAN
messages from the receive queue when
interface is not present anymore ... Passed

Running test #TC05.9 - Read CAN
messages from the receive queue while
errors on CAN bus are present ... Failed

File Name ../Testcases/Testcases.c **Line Number** 929

Condition CU_FAIL(Semi-automatic or manual test execution failed for this
testcase)

Running test TC05.10 - Read the status of
the receive queue after an overrun ... Passed

Running test TC05.11 - Read CAN
messages from the receive queue using
'blocking read' ... Passed

Running test #TC05.12 - Read CAN
messages from the receive queue when
interface is not connected to the bus
anymore ... Passed

Running Suite TC06 - Transmit a CAN message

Running test TC06.1 - Transmit a CAN
message while PCAN-USB interface is
running with default parameters ... Passed

Running test TC06.2 - Transmit a CAN
message when PCAN-USB interface is not
initialized before ... Passed

Running test TC06.3 - Transmit a CAN
message with an invalid value for
parameter 'Channel' ... Passed

Running test TC06.4 - Transmit a CAN
message with illegal parameter
'MessageBuffer == NULL' ... Passed

Running test TC06.5 - Transmit a standard
CAN message with a valid 11-bit identifier
and no data bytes (DLC = 0) ... Passed

Running test TC06.6 - Transmit a standard
CAN message with a valid 11-bit identifier
and one data byte (DLC = 1) ... Passed

Running test TC06.7 - Transmit a standard
CAN message with a valid 11-bit identifier
and two data bytes (DLC = 2) ... Passed

Running test TC06.8 - Transmit a standard
CAN message with a valid 11-bit identifier
and three data bytes (DLC = 3) ... Passed

Running test TC06.9 - Transmit a standard
CAN message with a valid 11-bit identifier
and four data bytes (DLC = 4) ... Passed

Running test TC06.10 - Transmit a
standard CAN message with a valid 11-bit
identifier and five data bytes (DLC = 5) ... Passed

Running test TC06.11 - Transmit a standard CAN message with a valid 11-bit identifier and six data bytes (DLC = 6) ... Passed

Running test TC06.12 - Transmit a standard CAN message with a valid 11-bit identifier and seven data bytes (DLC = 7) ... Passed

Running test TC06.13 - Transmit a standard CAN message with a valid 11-bit identifier and eight data bytes (DLC = 8) ... Passed

Running test TC06.14 - Transmit a standard CAN message with a valid 11-bit identifier and invalid data length code (DLC = 9) ... Passed

Running test TC06.15 - Transmit a standard CAN message with a valid 11-bit identifier and invalid data length code (DLC = 255) ... Passed

Running test TC06.16 - Transmit a standard CAN message with a valid 29-bit identifier (invalid for standard messages) ... Passed

Running test TC06.17 - Transmit a standard CAN message with an invalid 11-bit identifier (ID = 0x20000000) ... Passed

Running test TC06.18 - Transmit a standard CAN message with an invalid 11-bit identifier (ID = 0xFFFFFFFF) ... Passed

Running test TC06.19 - Request a standard CAN message with a valid 11-bit identifier and no data bytes (DLC = 0) ... Passed

Running test TC06.20 - Request a standard CAN message with a valid 11-bit identifier and one data byte (DLC = 1) ... Passed

Running test TC06.21 - Request a standard CAN message with a valid 11-bit identifier and two data bytes (DLC = 2) ... Passed

Running test TC06.22 - Request a standard CAN message with a valid 11-bit identifier and three data bytes (DLC = 3) ... Passed

Running test TC06.23 - Request a standard CAN message with a valid 11-bit identifier and four data bytes (DLC = 4) ... Passed

Running test TC06.24 - Request a standard CAN message with a valid 11-bit identifier and five data bytes (DLC = 5) ... Passed

Running test TC06.25 - Request a standard CAN message with a valid 11-bit identifier and six data bytes (DLC = 6) ... Passed

Passed

Running test TC06.26 - Request a standard CAN message with a valid 11-bit identifier and seven data bytes (DLC = 7) ...

Running test TC06.27 - Request a standard CAN message with a valid 11-bit identifier and eight data bytes (DLC = 8) ... Passed

Running test TC06.28 - Request a standard CAN message with a valid 11-bit identifier and wrong data length code ... Passed

Running test TC06.29 - Request a standard CAN message with a valid 11-bit identifier and invalid data length code ... Passed

Running test TC06.30 - Request a standard CAN message with a valid 29-bit identifier (invalid for standard messages) ... Passed

Running test TC06.31 - Request a standard CAN message with an invalid 11-bit identifier (ID = 0x20000000) ... Passed

Running test TC06.32 - Request a standard CAN message with an invalid 11-bit identifier (ID = 0xFFFFFFFF) ... Passed

Running test TC06.33 - Transmit an extended CAN message with a valid 29-bit identifier and no data bytes (DLC = 0) ... Passed

Running test TC06.34 - Transmit an extended CAN message with a valid 29-bit identifier and one data byte (DLC = 1) ... Passed

Running test TC06.35 - Transmit an extended CAN message with a valid 29-bit identifier and two data bytes (DLC = 2) ... Passed

Running test TC06.36 - Transmit an extended CAN message with a valid 29-bit identifier and three data bytes (DLC = 3) ... Passed

Running test TC06.37 - Transmit an extended CAN message with a valid 29-bit identifier and four data bytes (DLC = 4) ... Passed

Running test TC06.38 - Transmit an extended CAN message with a valid 29-bit identifier and five data bytes (DLC = 5) ... Passed

Running test TC06.39 - Transmit an extended CAN message with a valid 29-bit identifier and six data bytes (DLC = 6) ... Passed

Running test TC06.40 - Transmit an extended CAN message with a valid 29-bit identifier and seven data bytes (DLC = 7) ... Passed

Running test TC06.41 - Transmit an extended CAN message with a valid 29-bit identifier and eight data bytes (DLC = 8) ... Passed

Running test TC06.42 - Transmit an extended CAN message with a valid 29-bit identifier and invalid data length code (DLC = 9) ... Passed

Running test TC06.43 - Transmit an extended CAN message with a valid 29-bit identifier and invalid data length code (DLC = 255) ... Passed

Running test TC06.44 - Transmit an extended CAN message with an invalid 29-bit identifier (ID = 0x20000000) ... Passed

Running test TC06.45 - Transmit an extended CAN message with an invalid 29-bit identifier (ID = 0xFFFFFFFF) ... Passed

Running test TC06.46 - Request an extended CAN message with a valid 29-bit identifier and no data bytes (DLC = 0) ... Passed

Running test TC06.47 - Request an extended CAN message with a valid 29-bit identifier and one data byte (DLC = 1) ... Passed

Running test TC06.48 - Request an extended CAN message with a valid 29-bit identifier and two data bytes (DLC = 2) ... Passed

Running test TC06.49 - Request an extended CAN message with a valid 29-bit identifier and three data bytes (DLC = 3) ... Passed

Running test TC06.50 - Request an extended CAN message with a valid 29-bit identifier and four data bytes (DLC = 4) ... Passed

Running test TC06.51 - Request an extended CAN message with a valid 29-bit identifier and five data bytes (DLC = 5) ... Passed

Running test TC06.52 - Request an extended CAN message with a valid 29-bit identifier and six data bytes (DLC = 6) ... Passed

Running test TC06.53 - Request an extended CAN message with a valid 29-bit identifier and seven data bytes (DLC = 7) ... Passed

Running test TC06.54 - Request an extended CAN message with a valid 29-bit identifier and eight data bytes (DLC = 8) ... Passed

Running test TC06.55 - Request an extended CAN message with a valid 29-bit identifier and wrong data length code ... Passed

Running test TC06.56 - Request an extended CAN message with a valid 29-bit identifier and invalid data length code ... Passed

Passed

Running test TC06.57 - Request an extended CAN message with an invalid 29-bit identifier (ID = 0x20000000) ...

Running test TC06.58 - Request an extended CAN message with an invalid 29-bit identifier (ID = 0xFFFFFFFF) ...

Running test TC06.59 - Transmit a CAN message with illegal parameter
'MessageBuffer.MSGTYPE == 0x80' ... Passed

Running test TC06.60 - Transmit a CAN message with illegal parameter
'MessageBuffer.MSGTYPE == 0xFF' ... Passed

Running test #TC06.61 - Transmit CAN messages when interface is not present anymore ... Passed

Running test #TC06.62 - Transmit CAN messages while errors on CAN bus are present ... Failed

File Name ../Testcases/Testcases.c **Line Number** 929

Condition CU_FAIL(Semi-automatic or manual test execution failed for this testcase)

Running test TC06.63 - Transmit standard CAN messages with a valid 11-bit identifier back-on-back ... Passed

Running test #TC06.64 - Transmit CAN messages when interface is not connected to the bus anymore ... Passed

Running test TC06.65 - Transmit standard CAN messages and check for amount and order on receiver side ... Passed

Running Suite TC07 - Retrieve information from a PCAN Channel

Running test TC07.17.1 - Get parameter PCAN_DEVICE_ID while PCAN-USB interface is running with default parameters ... Passed

Running test TC07.17.2 - Get parameter PCAN_DEVICE_ID when PCAN-USB interface is not initialized before ... Passed

Running test TC07.17.3 - Get parameter PCAN_DEVICE_ID with an invalid value for parameter 'Channel' ... Passed

Running test TC07.17.4 - Get parameter PCAN_DEVICE_ID with illegal parameter 'Buffer == NULL' ... Passed

Running test TC07.17.5 - Get parameter PCAN_DEVICE_ID with an invalid value for parameter 'BufferLength' (too small) ... Passed

Running test TC07.17.6 - Get parameter
PCAN_DEVICE_ID with an invalid value
for parameter 'BufferLength' (too big) ...

Running test #TC07.17.7 - Get parameter
PCAN_DEVICE_ID when interface is not
present anymore ... Passed

Running test #TC07.17.8 - Get parameter
PCAN_DEVICE_ID when interface is not
connected to the bus anymore ... Passed

Running test TC07.19.1 - Get parameter
PCAN_RECEIVE_EVENT while PCAN-
USB interface is running with default
parameters ... Passed

Running test TC07.19.2 - Get parameter
PCAN_RECEIVE_EVENT when PCAN-
USB interface is not initialized before ... Passed

Running test TC07.19.3 - Get parameter
PCAN_RECEIVE_EVENT with an invalid
value for parameter 'Channel' ... Passed

Running test TC07.19.4 - Get parameter
PCAN_RECEIVE_EVENT with illegal
parameter 'Buffer == NULL' ... Passed

Running test TC07.19.5 - Get parameter
PCAN_RECEIVE_EVENT with an invalid
value for parameter 'BufferLength' (too
small) ... Passed

Running test TC07.19.6 - Get parameter
PCAN_RECEIVE_EVENT with an invalid
value for parameter 'BufferLength' (too
big) ... Passed

Running test #TC07.19.7 - Get parameter
PCAN_RECEIVE_EVENT when interface
is not present anymore ... Passed

Running test #TC07.19.8 - Get parameter
PCAN_RECEIVE_EVENT when interface
is not connected to the bus anymore ... Passed

Running test TC07.21.1 - Get parameter
PCAN_API_VERSION while PCAN-USB
interface is running with default
parameters ... Passed

Running test TC07.21.2 - Get parameter
PCAN_API_VERSION when PCAN-USB
interface is not initialized before ... Passed

Running test TC07.21.3 - Get parameter
PCAN_API_VERSION with legal
parameter 'Channel ==
PCAN_NONEBUS' ... Passed

Running test TC07.21.4 - Get parameter
PCAN_API_VERSION with illegal
parameter 'Buffer == NULL' ... Passed

Running test TC07.21.5 - Get parameter PCAN_API_VERSION with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test #TC07.21.6 - Get parameter PCAN_API_VERSION when interface is not present anymore ...	Passed
Running test #TC07.21.7 - Get parameter PCAN_API_VERSION when interface is not connected to the bus anymore ...	Passed
Running test #TC07.21.8 - Get parameter PCAN_API_VERSION when interface is used by another process ...	Passed
Running test TC07.22.1 - Get parameter PCAN_CHANNEL_VERSION while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.22.2 - Get parameter PCAN_CHANNEL_VERSION when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.22.3 - Get parameter PCAN_CHANNEL_VERSION with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.22.4 - Get parameter PCAN_CHANNEL_VERSION with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.22.5 - Get parameter PCAN_CHANNEL_VERSION with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test #TC07.22.6 - Get parameter PCAN_CHANNEL_VERSION when interface is not present anymore ...	Passed
Running test #TC07.22.7 - Get parameter PCAN_CHANNEL_VERSION when interface is not connected to the bus anymore ...	Passed
Running test #TC07.22.8 - Get parameter PCAN_CHANNEL_VERSION when interface is used by another process ...	Passed
Running test TC07.24.1 - Get parameter PCAN_LISTEN_ONLY while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.24.2 - Get parameter PCAN_LISTEN_ONLY when PCAN-USB interface is not initialized before ...	Passed
	Passed

Running test TC07.24.3 - Get parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'Channel' ...	
Running test TC07.24.4 - Get parameter PCAN_LISTEN_ONLY with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.24.5 - Get parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.24.6 - Get parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.24.7 - Get parameter PCAN_LISTEN_ONLY when interface is not present anymore ...	Passed
Running test #TC07.24.8 - Get parameter PCAN_LISTEN_ONLY when interface is not connected to the bus anymore ...	Passed
Running test #TC07.24.9 - Get parameter PCAN_LISTEN_ONLY when interface is used by another process ...	Passed
Running test TC07.29.1 - Get parameter PCAN_CHANNEL_CONDITION while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.29.2 - Get parameter PCAN_CHANNEL_CONDITION when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.29.3 - Get parameter PCAN_CHANNEL_CONDITION with illegal parameter 'Channel == PCAN_NONEBUS' ...	Passed
Running test TC07.29.4 - Get parameter PCAN_CHANNEL_CONDITION with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.29.5 - Get parameter PCAN_CHANNEL_CONDITION with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.29.6 - Get parameter PCAN_CHANNEL_CONDITION with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.29.7 - Get parameter PCAN_CHANNEL_CONDITION when interface is not present anymore ...	Passed
	Passed

Running test #TC07.29.8 - Get parameter PCAN_CHANNEL_CONDITION when interface is not connected to the bus anymore ...	
Running test #TC07.29.9 - Get parameter PCAN_CHANNEL_CONDITION when interface is used by another process ...	Passed
Running test TC07.30.1 - Get parameter PCAN_HARDWARE_NAME while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.30.2 - Get parameter PCAN_HARDWARE_NAME when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.30.3 - Get parameter PCAN_HARDWARE_NAME with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.30.4 - Get parameter PCAN_HARDWARE_NAME with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.30.5 - Get parameter PCAN_HARDWARE_NAME with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test #TC07.30.6 - Get parameter PCAN_HARDWARE_NAME when interface is not present anymore ...	Passed
Running test #TC07.30.7 - Get parameter PCAN_HARDWARE_NAME when interface is not connected to the bus anymore ...	Passed
Running test #TC07.30.8 - Get parameter PCAN_HARDWARE_NAME when interface is used by another process ...	Passed
Running test TC07.31.1 - Get parameter PCAN_RECEIVE_STATUS while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.31.2 - Get parameter PCAN_RECEIVE_STATUS when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.31.3 - Get parameter PCAN_RECEIVE_STATUS with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.31.4 - Get parameter PCAN_RECEIVE_STATUS with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.31.5 - Get parameter PCAN_RECEIVE_STATUS with an	Passed

invalid value for parameter 'BufferLength' (too small) ...	
Running test TC07.31.6 - Get parameter PCAN_RECEIVE_STATUS with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.31.7 - Get parameter PCAN_RECEIVE_STATUS when interface is not present anymore ...	Passed
Running test #TC07.31.8 - Get parameter PCAN_RECEIVE_STATUS when interface is not connected to the bus anymore ...	Passed
Running test #TC07.31.9 - Get parameter PCAN_RECEIVE_STATUS when interface is used by another process ...	Passed
Running test TC07.33.1 - Get parameter PCAN_TRACE_LOCATION while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.33.2 - Get parameter PCAN_TRACE_LOCATION when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.33.3 - Get parameter PCAN_TRACE_LOCATION with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.33.4 - Get parameter PCAN_TRACE_LOCATION with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.33.5 - Get parameter PCAN_TRACE_LOCATION with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test #TC07.33.6 - Get parameter PCAN_TRACE_LOCATION when interface is not present anymore ...	Passed
Running test #TC07.33.7 - Get parameter PCAN_TRACE_LOCATION when interface is not connected to the bus anymore ...	Passed
Running test TC07.34.1 - Get parameter PCAN_TRACE_STATUS while PCAN- USB interface is running with default parameters ...	Passed
Running test TC07.34.2 - Get parameter PCAN_TRACE_STATUS when PCAN- USB interface is not initialized before ...	Passed
	Passed

Running test TC07.34.3 - Get parameter PCAN_TRACE_STATUS with an invalid value for parameter 'Channel' ...	
Running test TC07.34.4 - Get parameter PCAN_TRACE_STATUS with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.34.5 - Get parameter PCAN_TRACE_STATUS with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.34.6 - Get parameter PCAN_TRACE_STATUS with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.34.7 - Get parameter PCAN_TRACE_STATUS when interface is not present anymore ...	Passed
Running test #TC07.34.8 - Get parameter PCAN_TRACE_STATUS when interface is not connected to the bus anymore ...	Passed
Running test TC07.35.1 - Get parameter PCAN_TRACE_SIZE while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.35.2 - Get parameter PCAN_TRACE_SIZE when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.35.3 - Get parameter PCAN_TRACE_SIZE with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.35.4 - Get parameter PCAN_TRACE_SIZE with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.35.5 - Get parameter PCAN_TRACE_SIZE with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.35.6 - Get parameter PCAN_TRACE_SIZE with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.35.7 - Get parameter PCAN_TRACE_SIZE when interface is not present anymore ...	Passed
Running test #TC07.35.8 - Get parameter PCAN_TRACE_SIZE when interface is not connected to the bus anymore ...	Passed
Running test TC07.36.1 - Get parameter PCAN_TRACE_CONFIGURE while	Passed

PCAN-USB interface is running with default parameters ...	
Running test TC07.36.2 - Get parameter PCAN_TRACE_CONFIGURE when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.36.3 - Get parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.36.4 - Get parameter PCAN_TRACE_CONFIGURE with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.36.5 - Get parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.36.6 - Get parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.36.7 - Get parameter PCAN_TRACE_CONFIGURE when interface is not present anymore ...	Passed
Running test #TC07.36.8 - Get parameter PCAN_TRACE_CONFIGURE when interface is not connected to the bus anymore ...	Passed
Running test TC07.38 - Get parameter PCAN_EXT_BTR0BTR1 while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.39 - Get parameter PCAN_EXT_TX_COUNTER when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.40 - Get parameter PCAN_EXT_RX_COUNTER with illegal parameter 'Channel == PCAN_NONEBUS' ...	Passed
Running test TC07.41 - Get parameter PCAN_EXT_ERR_COUNTER with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.42 - Get parameter PCAN_EXT_RX_QUE_OVERRUN with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.43 - Get parameter PCAN_EXT_HARDWARE_VERSION with an invalid value for parameter 'BufferLength' (too big) ...	Passed

Running test #TC07.44.1 - Get parameter PCAN_EXT_SOFTWARE_VERSION when interface is not present anymore ...	Passed
Running test #TC07.44.2 - Get parameter PCAN_EXT_SOFTWARE_VERSION when interface is not connected to the bus anymore ...	Passed
Running test TC07.47.1 - Get parameter PCAN_CHANNEL_FEATURES while PCAN-USB interface is running with default parameters ...	Passed
Running test TC07.47.2 - Get parameter PCAN_CHANNEL_FEATURES when PCAN-USB interface is not initialized before ...	Passed
Running test TC07.47.3 - Get parameter PCAN_CHANNEL_FEATURES with an invalid value for parameter 'Channel' ...	Passed
Running test TC07.47.4 - Get parameter PCAN_CHANNEL_FEATURES with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC07.47.5 - Get parameter PCAN_CHANNEL_FEATURES with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC07.47.6 - Get parameter PCAN_CHANNEL_FEATURES with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test #TC07.47.7 - Get parameter PCAN_CHANNEL_FEATURES when interface is not present anymore ...	Passed
Running test #TC07.47.8 - Get parameter PCAN_CHANNEL_FEATURES when interface is not connected to the bus anymore ...	Passed
Running test #TC07.47.9 - Get parameter PCAN_CHANNEL_FEATURES when interface is used by another process ...	Passed
Running test #TC07.49.9 - Get parameter PCAN_BITRATE_INFO when interface is not connected to the bus anymore ...	Passed
Running test TC07.99 - Get value with illegal value for parameter 'Parameter' ...	Passed
Running Suite TC08 - Set a configuration or information value within a PCAN Channel	
Running test TC08.17.1 - Set parameter PCAN_DEVICE_ID while PCAN-USB interface is running with default parameters ...	Passed

Running test TC08.17.2 - Set parameter PCAN_DEVICE_ID when PCAN-USB interface is not initialized before ... Passed

Running test TC08.17.3 - Set parameter PCAN_DEVICE_ID with an invalid value for parameter 'Channel' ... Passed

Running test TC08.17.4 - Set parameter PCAN_DEVICE_ID with illegal parameter 'Buffer == NULL' ... Passed

Running test TC08.17.5 - Set parameter PCAN_DEVICE_ID with an invalid value for parameter 'BufferLength' (too small) ... Passed

Running test TC08.17.6 - Set parameter PCAN_DEVICE_ID with an invalid value for parameter 'BufferLength' (too big) ... Passed

Running test TC08.17.7 - Set parameter PCAN_DEVICE_ID with an invalid value '*Buffer == 256' ... Passed

Running test TC08.17.8 - Set parameter PCAN_DEVICE_ID with an invalid value '*Buffer == -1' ... Passed

Running test #TC08.17.9 - Set parameter PCAN_DEVICE_ID when interface is not present anymore ... Passed

Running test #TC08.17.10 - Set parameter PCAN_DEVICE_ID when interface is not connected to the bus anymore ... Passed

Running test TC08.21.1 - Set parameter PCAN_API_VERSION while PCAN-USB interface is running with default parameters ... Passed

Running test TC08.21.2 - Set parameter PCAN_API_VERSION when PCAN-USB interface is not initialized before ... Passed

Running test TC08.22.1 - Set parameter PCAN_CHANNEL_VERSION while PCAN-USB interface is running with default parameters ... Passed

Running test TC08.22.2 - Set parameter PCAN_CHANNEL_VERSION when PCAN-USB interface is not initialized before ... Passed

Running test TC08.24.1 - Set parameter PCAN_LISTEN_ONLY while PCAN-USB interface is running with default parameters ... Passed

Running test TC08.24.2 - Set parameter PCAN_LISTEN_ONLY when PCAN-USB interface is not initialized before ... Passed

Running test TC08.24.3 - Set parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'Channel' ...	Passed
Running test TC08.24.4 - Set parameter PCAN_LISTEN_ONLY with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC08.24.5 - Set parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC08.24.6 - Set parameter PCAN_LISTEN_ONLY with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test TC08.24.7 - Set parameter PCAN_LISTEN_ONLY with an invalid value '*Buffer == 2' ...	Passed
Running test TC08.24.8 - Set parameter PCAN_LISTEN_ONLY with an invalid value '*Buffer == -1' ...	Passed
Running test TC08.24.9 - Set parameter PCAN_LISTEN_ONLY with value '*Buffer == PCAN_PARAMETER_ON' before initialization ...	Passed
Running test TC08.24.10 - Set parameter PCAN_LISTEN_ONLY with value '*Buffer == PCAN_PARAMETER_ON' after initialization ...	Passed
Running test TC08.24.11 - Set parameter PCAN_LISTEN_ONLY with value '*Buffer == PCAN_PARAMETER_ON' and check mode after uninitializing ...	Passed
Running test TC08.24.12 - Set parameter PCAN_LISTEN_ONLY with value '*Buffer == PCAN_PARAMETER_OFF' after initialization in listen-only mode ...	Passed
Running test #TC08.24.13 - Set parameter PCAN_LISTEN_ONLY when interface is not present anymore ...	Passed
Running test #TC08.24.14 - Set parameter PCAN_LISTEN_ONLY when interface is not connected to the bus anymore ...	Passed
Running test #TC08.24.15 - Set parameter PCAN_LISTEN_ONLY when interface is used by another process ...	Passed
Running test TC08.29.1 - Set parameter PCAN_CHANNEL_CONDITION while PCAN-USB interface is running with default parameters ...	Passed
	Passed

Running test TC08.29.2 - Set parameter PCAN_CHANNEL_CONDITION when PCAN-USB interface is not initialized before ...	
Running test TC08.30.1 - Set parameter PCAN_HARDWARE_NAME while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.30.2 - Set parameter PCAN_HARDWARE_NAME when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.33.1 - Set parameter PCAN_TRACE_LOCATION while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.33.2 - Set parameter PCAN_TRACE_LOCATION when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.33.3 - Set parameter PCAN_TRACE_LOCATION with an invalid value for parameter 'Channel' ...	Passed
Running test TC08.33.4 - Set parameter PCAN_TRACE_LOCATION with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC08.33.5 - Set parameter PCAN_TRACE_LOCATION with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC08.33.6 - Set parameter PCAN_TRACE_LOCATION with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test TC08.33.7 - Set parameter PCAN_TRACE_LOCATION with valid strings for parameter '*Buffer' and read back ...	Passed
Running test TC08.33.8 - Set parameter PCAN_TRACE_LOCATION with an empty string for parameter '*Buffer' ...	Passed
Running test #TC08.33.9 - Set parameter PCAN_TRACE_LOCATION when interface is not present anymore ...	Passed
Running test #TC08.33.10 - Set parameter PCAN_TRACE_LOCATION when interface is not connected to the bus anymore ...	Passed
Running test TC08.34.1 - Set parameter PCAN_TRACE_STATUS while PCAN-	Passed

USB interface is running with default parameters ...

Running test TC08.34.2 - Set parameter PCAN_TRACE_STATUS when PCAN-USB interface is not initialized before ... Passed

Running test TC08.34.3 - Set parameter PCAN_TRACE_STATUS with an invalid value for parameter 'Channel' ... Passed

Running test TC08.34.4 - Set parameter PCAN_TRACE_STATUS with illegal parameter 'Buffer == NULL' ... Passed

Running test TC08.34.5 - Set parameter PCAN_TRACE_STATUS with an invalid value for parameter 'BufferLength' (too small) ... Passed

Running test TC08.34.6 - Set parameter PCAN_TRACE_STATUS with an invalid value for parameter 'BufferLength' (too big) ... Passed

Running test TC08.34.7 - Set parameter PCAN_TRACE_STATUS with valid values for parameter '*Buffer' and read back ... Passed

Running test TC08.34.8 - Set parameter PCAN_TRACE_STATUS with invalid values for parameter '*Buffer' ... Passed

Running test #TC08.34.9 - Set parameter PCAN_TRACE_STATUS when interface is not present anymore ... Passed

Running test #TC08.34.10 - Trace a lot of CAN messages and check the trace file for format and content ... Passed

Running test #TC08.34.12 - Set parameter PCAN_TRACE_STATUS when interface is not connected to the bus anymore ... Passed

Running test TC08.34.13 - Set parameter PCAN_TRACE_STATUS to PCAN_PARAMETER_OFF when trace file not opened before ... Passed

Running test TC08.34.14 - Trace some CAN messages and check the trace file for correct format ... Passed

Running test TC08.35.1 - Set parameter PCAN_TRACE_SIZE while PCAN-USB interface is running with default parameters ... Passed

Running test TC08.35.2 - Set parameter PCAN_TRACE_SIZE when PCAN-USB interface is not initialized before ... Passed

Running test TC08.35.3 - Set parameter PCAN_TRACE_SIZE with an invalid value for parameter 'Channel' ...	Passed
Running test TC08.35.4 - Set parameter PCAN_TRACE_SIZE with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC08.35.5 - Set parameter PCAN_TRACE_SIZE with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC08.35.6 - Set parameter PCAN_TRACE_SIZE with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test TC08.35.7 - Set parameter PCAN_TRACE_SIZE with valid values for parameter '*Buffer' and read back ...	Passed
Running test TC08.35.8 - Set parameter PCAN_TRACE_SIZE with invalid values for parameter '*Buffer' ...	Passed
Running test #TC08.35.9 - Set parameter PCAN_TRACE_SIZE when interface is not present anymore ...	Passed
Running test #TC08.35.10 - Set parameter PCAN_TRACE_SIZE when interface is not connected to the bus anymore ...	Passed
Running test TC08.36.1 - Set parameter PCAN_TRACE_CONFIGURE while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.36.2 - Set parameter PCAN_TRACE_CONFIGURE when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.36.3 - Set parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'Channel' ...	Passed
Running test TC08.36.4 - Set parameter PCAN_TRACE_CONFIGURE with illegal parameter 'Buffer == NULL' ...	Passed
Running test TC08.36.5 - Set parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'BufferLength' (too small) ...	Passed
Running test TC08.36.6 - Set parameter PCAN_TRACE_CONFIGURE with an invalid value for parameter 'BufferLength' (too big) ...	Passed
Running test TC08.36.7 - Set parameter PCAN_TRACE_CONFIGURE with valid	Passed

values for parameter '*Buffer' and read back ...	
Running test TC08.36.8 - Set parameter PCAN_TRACE_CONFIGURE with invalid values for parameter '*Buffer' ...	Passed
Running test #TC08.36.9 - Set parameter PCAN_TRACE_CONFIGURE when interface is not present anymore ...	Passed
Running test #TC08.36.10 - Set parameter PCAN_TRACE_CONFIGURE when interface is not connected to the bus anymore ...	Passed
Running test TC08.45 - Set parameter PCAN_EXT_LOG_USB and write all function calls to a log-file ...	Passed
Running test TC08.46 - Set parameter PCAN_EXT_RECEIVE_CALLBACK and check for call back of the callback function ...	Passed
Running test TC08.47.1 - Set parameter PCAN_CHANNEL_FEATURES while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.47.2 - Set parameter PCAN_CHANNEL_FEATURES when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.49.1 - Set parameter PCAN_BITRATE_INFO while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.49.2 - Set parameter PCAN_BITRATE_INFO when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.50.1 - Set parameter PCAN_BITRATE_INFO_FD while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.50.2 - Set parameter PCAN_BITRATE_INFO_FD when PCAN-USB interface is not initialized before ...	Passed
Running test TC08.51.1 - Set parameter PCAN_BUSSPEED_NOMINAL while PCAN-USB interface is running with default parameters ...	Passed
Running test TC08.51.2 - Set parameter PCAN_BUSSPEED_NOMINAL when PCAN-USB interface is not initialized before ...	Passed

Running test TC08.52.1 - Set parameter PCAN_BUSSPEED_DATA while PCAN-USB interface is running with default parameters ... Passed

Running test TC08.52.2 - Set parameter PCAN_BUSSPEED_DATA when PCAN-USB interface is not initialized before ... Passed

Running test TC08.99 - Set an illegal value for parameter 'Parameter' ... Passed

Running Suite TC10 - Retrieve an error text

Running test TC10.1 - Get error text while PCAN-USB interface is running with default parameters ... Passed

Running test TC10.2 - Get error text when PCAN-USB interface is not initialized before ... Passed

Running test TC10.3 - Get error text with illegal parameter 'Error == 4294967295' ... Passed

Running test TC10.4 - Get error text with illegal parameter 'Language == 57005' ... Passed

Running test TC10.5 - Get error text with illegal parameter 'Buffer == NULL' ... Passed

Running test TC10.6 - Get the English error text for each defined error code and check it ... Passed

Running test TC10.7 - Get the German error text for each defined error code and check it ... Passed

Running test TC10.8 - Get the French error text for each defined error code and check it ... Passed

Running test TC10.9 - Get the Italian error text for each defined error code and check it ... Passed

Running test TC10.10 - Get the Spanish error text for each defined error code and check it ... Passed

Running test TC10.11 - Get error text for OR-ed parameter 'Error == PCAN_ERROR_ANYBUSERR' ... Passed

Running test TC10.12 - Get error text for OR-ed parameter 'Error == PCAN_ERROR_ILHHANDLE' ... Passed

Cumulative Summary for Run

Type	Total	Run	Succeeded	Failed	Inactive
Suites	9	9	- NA -	0	0
Test Cases	378	335	333	2	43

Assertions	636862	636862	636860	2	n/a
------------	--------	--------	--------	---	-----

File Generated By CUnit v2.1-1 - Sun Dec 5 10:42:51 2021