AT&T
INTEGRATED SERVICES DIGITAL NETWORK (ISDN) PRIMARY RATE INTERFACE TEST PLAN FOR CUSTOMER PREMISES EQUIPMENT

VERSION 2.0

AT&T Developmental Relations Group

CONTENTS

1. INTRODUCTION

1.1 AT&T ISDN PRI Interface

AT&T Integrated Services Digital Network (ISDN) Primary Rate Interface (PRI) is an ISDN offering based on CCITT Recommendations. The interface is specified using the OSI layered model. The physical layer of the interface is based on the standard DS-1 interface with Extended Superframe Format (ESF) or SF[1] framing and on CCITT Recommendations I.412, I.431 and I.211[1].

A specification of the ISDN PRI interface can be found in AT&T Technical Reference 41459[2]. It contains the description of AT&T Switched Network specific PRI capabilities and features.[2] It refers to AT&T Technical Reference 41449[3], AT&T Technical Reference 62411[4] and AT&T Technical Reference 54016[5] for protocol specification for both the network and user side of the interface.

1.2 Purpose of This Document

A CONformance TEST system for PRI (CONTEST-PRI) has been designed and developed to determine the compatibility of Customer Premises Equipment (CPE) with the AT&T ISDN PRI services and features as described in TR 41459. This document provides details of the test plan and test procedures used in CONTEST-PRI.

This test plan, at AT&T's option, may be altered in order to improve test methods, to reflect new network capabilities or to facilitate testing of the CPE Implementation Under Test (IUT) in unusual cases, where the test procedures described herein are not viable. If it is determined that the test cases, as described in this document are not consistent with AT&T TR 41449 or TR 41459, AT&T reserves the right to modify or include additional test cases. In all these cases, the compatibility statement will additionally reflect the verdicts associated with these modified test cases.

Technical descriptions of the interface requirements as they appear in this document are included solely for clarifying the test procedures. Such technical descriptions are intended to be in agreement with TR 41449 and TR 41459, which remain the authoritative references regarding AT&T ISDN PRI.

1.3 Organization

The remainder of this section identifies the target audience for the test plan and the terminology used in other parts of the test plan. Section 2 provides a general overview of the test plan. Some of the tests can be executed from a remote location using the remote test unit called CONTEST-RTU. A description of the remote testing capabilities are presented in Section 3. Capabilities not covered by the test plan are identified in Section 4. Section 5 lists some of the requirements placed on the IUT in order to complete the tests. Appendicies A, B and C contain the test cases to be used for testing the physical layer, Layer 2 and Layer 3 protocols/procedures, respectively. Appendix D contains the user's manual for using CONTEST-RTU. Finally, Appendix E contains the CPE interoperability questionnaire which is used in preparing for a test session.

1.4 Audience

This test plan is intended for the vendors and developers of ISDN CPE.

---

1. Older documents refer to Superframe Format (SF) framing and formatting as D3, D4, or D3/D4.
2. Some features such as equal access, are specific to Local Exchange Carriers are not within the scope of TR 41459. Therefore, such features are not covered in this test plan.

1.5 Terminology

CCITT	International Telegraph and Telephone Consultative Committee
CONTEST-PRI	CONformance TEST system for PRI
CONTEST-RTU	CONformance TEST Remote Test Unit
CPE	Customer Premise Equipment
ESF	Extended Superframe Format
IUT	Implementation Under Test
ISDN	Integrated Services Digital Network
LAPD	Link Access Procedures on the D-channel
NFAS	Non-Facility Associated Signaling
OSI	Open System Interconnection
PRI	Primary Rate Interface
SF	Superframe Format

2. GENERAL DESCRIPTION OF TEST SUITES

Detailed tests for each set of procedures and protocols for the three layers that will be tested by CONTEST-PRI are provided in the appendicies to this document. Each set of capabilities is tested individually while disabling other testing capabilities. Thus, while testing the Layer 3 (Q.931) protocol, CONTEST-PRI will not test LAPD procedures.

The tests described in this document are either mandatory or conditional:

1. mandatory - successful completion of these tests are required for minimal compatibility;
2. conditional - successful completion of these tests are required if the IUT has implemented the associated capability, feature or service.

For physical layer compatibility, the test plan identifies the mandatory tests. All Layer 2 and Layer 3 tests are named with a prefix. Mandatory tests have the prefix M and conditional tests have the prefix C. Tests for optional procedures for optional capability, feature or service have the prefix CC.

2.1 Layer 1

Layer 1 tests are consistent with those done to verify compatibility with the ACCUNET ® T1.5 Service, including the Customer Controlled Reconfiguration (CCR), M24, and M24-DDT service functions, as well as the enhanced facility maintenance features of the Extended Superframe Format (ESF). Appendix A provides descriptions of the CCR/M24/M24-DDT and ESF tests.

Some special preparation of the IUT is required so that the input jitter tolerance test can be conducted. It is the responsibility of the vendor to prepare for this test before the IUT is presented for testing. The requirement and possible ways to prepare the IUT for these tests are discussed in Appendix A.

2.2 Layer 2

Layer 2 tests verify the conformance of the IUT to the procedures as specified in TR 41449 and TR 41459. This is done by evaluating the response, if any, of the IUT upon receipt of one of the following frame types:

i. Valid frames. These are LAPD frames that are syntactically valid and are sent to the IUT in the correct state.

ii. Inopportune frames. These are LAPD frames that are syntactically valid but are sent to the IUT in an improper state.

iii. Invalid frames. These are frames with syntax that does not conform to the LAPD specifications.

For each test, the CONTEST-PRI will

- initialize the IUT into one of the nine possible states and
- compare the response from the IUT to the expected frame (if any).

The following is the list of the states which are tested:

State	Name
S5	Awaiting Establishment State (establishment process has been started but not completed)
S7.0	Multiple Frame Established (Normal) State (multiple mode of information transfer has started)
S7.1	Multiple Frame Established (Reject) State (reject condition)
S8.4	Timer Recovery (Normal, Peer Receiver Busy) State (timer T200 has expired)
S8.5	Timer Recovery (Reject, Peer Receiver Busy) State (reject condition)
S4	TEI Assigned State (acknowledged information transfer is not possible)
S6	Awaiting Release State (link release procedure has started)
S7.2	Multiple Frame Established (Busy) State (busy condition in multi-frame state)
S8.6	Timer Recovery (Peer and Own Receiver Busy) State (local busy condition causing timer T200 to expire)

In addition to the above tests, Layer 2 tests also verify the implementation of proper rotation of frame sequence numbers, address checking and proper handling of parameters such as timers and window size.

2.3 Layer 3

Layer 3 tests verify the conformance of the IUT to the procedures as specified in TR 41449 and TR 41459. This is done by evaluating the response, if any, of the IUT upon receipt of one of the following messages:

i. Valid messages. These are Q.931 messages that are syntactically valid and are sent to the IUT in the correct state.

ii. Inopportune messages. These are Q.931 messages that are syntactically valid but are sent to the IUT in an improper state.

iii. Invalid messages. These are messages with syntax that does not conform to the Q.931 specifications.

For each test, the CONTEST-PRI will

- initialize the IUT into one of the ten possible reportable states and
- compare the response from the IUT to the expected message (if any).

The following is the list of the reportable states which can be tested:

State	Name
S0	Null
S1	Call Initiated
S3	Outgoing Call Proceeding
S4	Call Delivered
S7	Call Received
S8	Connect Request
S9	Incoming Call Proceeding
S10	Active
S11	Disconnect Indication
S19	Release Request

These tests will be done for maintenance, call control, channel negotiation, sending and receiving of user-to-user information, NFAS and D-channel backup procedures. In addition to the above tests, Layer 3 tests also verify the implementation of proper handling of timers and supported Network provided features and services.

Layer 3 tests rely heavily on the RESTART and STATUS ENQUIRY messages, and on proper IUT response to them through the RESTART ACKNOWLEDGE and STATUS messages, respectively. The processing of these messages is mandatory and without proper operation of these messages and associated IUT actions, further testing is not possible.

3. REMOTE TESTING OF PRI CPE

Associated with CONTEST-PRI, AT&T offers a remote testing capability to vendors as an aid for checking and debugging PRI protocol implementations. This capability is facilitated by a remote test unit called CONTEST-RTU. Appendix D contains the user's manual for this unit. Remote testing is not equivalent to on-site compatibility testing; testing of Layer 1 procedures and certain aspects of testing of Layers 2 and 3 procedures are not supported remotely. Nevertheless, successful completion of remote testing will significantly enhance the probability that an IUT will successfully complete the full suite of ISDN PRI tests when it undergoes on-site testing of Layers 2 and 3 procedures.

4. DIFFERENCES WITH THE 1989 TEST PLAN

The AT&T ISDN PRI test plan has been updated to incorporate new procedures added to the call control protocol to access additional capabilities and features provided by the network. Additional tests have been included to enhance the testing capability. Finally, Layer 2 and Layer 3 tests have been rearranged according to the functions of the protocol and network provided services and features. This regrouping will make it easier to select the tests for a given IUT based on the services and features it supports.

4.1 Physical Layer

There are no changes to the physical layer tests. However, the presentation of information is different.

4.2 Layer 2

The changes in Layer 2 tests are minimal:

1. State numbers have been aligned with the CCITT Recommendation Q.921.
2. Test 11 in Section 5 of the 1989 Version has been fixed.

3. The order of tests have been changed in some places so that all mandatory tests are followed by conditional tests. The following table gives the mapping between the two groupings. (Section number of Appendix 2 of the 1989 version precede the forward slash "/".)

1989 Version	Current Version
2/1-11,14-19 2/12,13	C_STATE4:1-17 C_UIXID:11,12
3/1-10,13-15 3/11,12	M_STATE5:1-11, 13-14 C_UIXID:1,2
4/1-11,14-16 4/12,13	C_STATE6:1-14 C_UIXID:13,14
5/1-12,15-18 5/13,14	M_STATE7.0:1-13, 17-19 C_UIXID:3,4
6/1-12,15-17 6/13,14 6/18	M_STATE7.1:1-15 C_UIXID:5,6 M_TIMER:1
7/1-12,15-17 7/13,14	C_STATE7.2:1-15 C_UIXID:15,16
8/1-13,16-18 8/14,15	M_STATE8.4:1-14, 16-17 C_UIXID:7,8
9/1-13,16-18 9/14,15	M_STATE8.5:1-16 C_UIXID:9,10
10/1-12,15-17 10/13,14	C_STATE8.6:1-15 C_UIXID:17,18
11/1	M_RCVSEQ:1
12/1-3	C_SNDSEQ:1-3
13/1-4	M_TIMER:2-5
14/1,2,5,6 14/3 14/4 14/7,8 14/9	C_STATE4:18-21 Dropped M_STATE7.0:20 M_STATE7.0:21,22 M_STATE7.0:23

4.3 Layer 3

The changes in Layer 3 are more extensive. New Network services and features require additional tests. In addition to new tests, the tests have been grouped differently so as to ease the selection of tests for a given CPE. The following table gives the mapping between the old and new tests. The following scheme is used to specify the tests from the 1989 version: i) tests from Appendix 3.2 (of the 1989 version) has a prefix "B", to distinguish the number from that of Appendix 3.1 (of the 1989 version); ii) the last component is the test number; iii) the rest identify the section number in which the test appears.

The table does not contain information regarding Sections 7, 10 and 11 of Appendix 3.1 and Sections 7 and 8 of Appendix 3.2. These tests have been incorporated and expanded in tests of Section C.15. (The modifications are so extensive, the mapping may not be very useful.)

1989 Version	Current Version	1989 Version	Current Version	1989 Version	Current Version
1.1	M_VAL:1	1.2	M_VAL:2	1.3	M_VAL:3
1.4	Dropped	1.5	C_VAL:1	1.6	M_VAL:4
1.7	M_VAL:5	1.8	M_VAL:6	1.9	M_VAL:7
1.10	M_VAL:8	1.11	M_VAL:9	1.12	C_FAST_CON:1
1.13	C_STATE7:6	1.14	C_STATE7:1	1.15	C_STATE7:2
1.16	C_STATE7:3	1.17	M_VAL:10	1.18	M_VAL:11
1.19	M_VAL:12	1.20	M_VAL:13	1.21	M_VAL:14
1.22	M_VAL:15	1.23	C_FAST_CON:2	1.24	M_VAL:16
1.25	C_FAST_CON:3	1.26	C_STATE9:1	1.27	C_STATE9:7 CC_STATE9:1
1.28	C_STATE9:2	1.29	C_STATE9:3	1.30	M_VAL:17
1.31	M_MAUUI:1	1.32	Dropped	2.1	M_INOP:1
2.2	M_INOP:2	2.3	M_INOP:3	2.4	M_INOP:4
2.5	M_INOP:5	2.6	C_STATE9:6	2.7	C_STATE7:6
2.8	M_INOP:6	2.9	M_INOP:7	2.10	M_INOP:8
3.1	M_INVAL:1 M_MAUUI:2	3.2	M_VAL:18 M_INVAL:2	3.3	M_INVAL:3
3.4	M_INVAL:4	3.5	M_INVAL:5	3.6	C_STATE9:4 C_STATE9:5
3.7	C_STATE7:4 C_STATE7:5	3.8	M_INVAL:6 M_MAUUI:3	3.9	M_INVAL:7
3.10	M_INVAL:8	4.1	C_US1:1	4.2	C_US1:5
4.3	C_US1:2	4.4	C_US1:3	4.5	C_US1:6
4.6	C_US1:6,7	4.7	C_US1:4	5.1	C_US2:1
5.2	C_US2:5	5.3	C_US2:2	5.4	C_US2:3
5.5	C_US2:6	5.6	C_US2:6,7	5.7	C_US2:4
6.1	M_TIMER:1	6.2	C_TIMER:1	6.3	C_TIMER:2
6.4	C_TIMER:3	6.5	M_TIMER:2	8.1	M_IECHK:1
8.2	M_IECHK:2	8.3	M_IECHK:3	8.4	M_IECHK:4
8.5	M_IECHK:5	8.6	M_IECHK:24	8.7	M_IECHK:25
8.8	M_IECHK:6	8.9	M_IECHK:7	8.10	M_IECHK:8
8.11	M_IECHK:9	8.12	M_IECHK:10	8.13	M_IECHK:11
8.14	M_IECHK:26	8.15	M_IECHK:27	8.16	M_IECHK:12
8.17	M_IECHK:13	8.18	M_IECHK:14	8.19	C_RESTART:1
8.20	M_IECHK:15	8.21	C_US1:8 C_US2:8	8.22	C_US1:9 C_US2:9
8.23	C_US1:10 C_US2:10	8.24	C_US1:11 C_US2:11	8.25	C_US1:12 C_US2:12
8.26	C_US1:13 C_US2:13	8.27	M_IECHK:16	8.28	M_IECHK:17
8.29	M_IECHK:18	8.30	M_IECHK:19	8.31	M_IECHK:20
8.32	M_IECHK:22	9.1	M_MAINT:1	9.2	M_MAINT:2

1989 Version	Current Version	1989 Version	Current Version	1989 Version	Current Version
9.3	M_MAINT:3	9.4	M_MAINT:15	9.5	M_MAINT:16
9.6	M_MAINT:17	9.7	C_NFAS:4	9.8	C_NFAS:5
9.9	C_NFAS:6	9.10	M_MAINT:24	9.11	M_MAINT:13 C_NFAS:7
9.12	M_MAINT:4	9.13	M_MAINT:5	9.14	M_MAINT:6
9.15	C_MAINT:2	9.16	C_MAINT:1	9.17	M_MAINT:18
9.18	M_MAINT:19	9.19	M_MAINT:20	9.20	M_MAINT:25
9.21	M_MAINT:8	9.22	M_MAINT:9	9.23	M_MAINT:10
9.24	M_MAINT:11	9.25	M_MAINT:21	9.26	M_MAINT:22
9.27	M_MAINT:23	9.28	M_MAINT:26	12.1	C_NFAS:1
12.2	C_NFAS:2	12.3	C_NFAS:3	13.1	C_CPN_REST:1
14.1	C_CPN_TERM:1	15.1	C_X_MAUUI:1	15.2	C_X_MAUUI:2
16.1	M_MAINT:12	B1.1	C_DCB:1	B1.2	C_DCB:2
B1.3	C_DCB:3	B1.4	C_DCB:4	B1.5	C_DCB:5
B1.6	C_DCB:6	B1.7	C_DCB:7	B1.8	C_DCB:8
B1.9	C_DCB:9	B1.10	C_DCB:10	B1.11	C_DCB:11
B2.1	C_CH_NEG:1	B2.2	C_CH_NEG:2	B2.3	C_CH_NEG:3
B3.1.1	CC_CNEG_NF:1	B3.1.2	CC_CNEG_NF:2	B3.1.3	CC_CNEG_NF:3
B3.1.4	CC_CNEG_NF:4	B3.2.1	CC_DCB_NFAS:1	B3.2.2	CC_DCB_NFAS:2
B4.1	C_CA_TSC:1	B4.2	C_CA_TSC:5	B4.3	C_CA_TSC:4
B4.4	C_CA_TSC:7	B4.5	C_CA_TSC:8	B4.6	C_CA_TSC:14
B4.7	C_CA_TSC:26	B4.8	C_CA_TSC:11	B4.9	CC_CATSC_R:1
B4.10	CC_CATSC_R:6	B4.11	C_CA_TSC:2	B4.12	C_CA_TSC:12
B4.13	C_CA_TSC:9	B4.14	C_CA_TSC:24	B4.15	C_CA_TSC:24
B4.16	C_CA_TSC:2	B4.17	Dropped	B4.18	C_CA_TSC:2
B4.19	C_CA_TSC:2	B4.20	C_CA_TSC:2	B4.21	C_CA_TSC:27
B4.22	CC_CATSC_F:1	B4.23	C_CA_TSC:6	B5.1	C_NCTSC:1
B5.2	C_NCTSC:2	B5.3	CC_NCTSC_R:1	B5.4	C_NCTSC:3 CC_NCTSC_R:2
B5.5	Dropped	B5.6	C_NCTSC:4	B5.7	C_NCTSC:5
B5.8	C_NCTSC:6	B5.9	C_NCTSC:7	B5.10	CC_NCTSC_F:1
B5.11	Dropped	B6.1.1	Dropped	B6.2.1	C_MSGLEN:1
B6.2.2	C_MSGLEN:2				

5. AT&T ISDN PRI CAPABILITIES NOT COVERED BY THIS TEST PLAN

This test plan does not verify whether the IUT uses end-user information carried as user-to-user information appropriately. It also does not verify the interactions between the procedures taking place on the PRI interface and those that take place at the interface between the IUT and its supported devices. This test plan does not address performance parameters like load handling.

6. TESTING REQUIREMENTS ON IUT

Testing in accordance with this test plan requires that the IUT be able to disable or relax some of the protocol requirements during testing. In particular,

- during Layer 2 testing, the IUT must disable Layer 3 operation,
- during Layer 3 testing, the IUT must be able to expand or disable Layer 3 timers at the IUT.

Also, the vendor must provide the necessary equipment for accessing supported services and features.

7. TESTING PROCESS

Testing agreements between a vendor and AT&T Development Relations Group (DRG) initiate the testing process. As part of the agreement, the vendor will receive a copy of this test plan and a vendor questionnaire. The information gathered from this will be used in designing the tests specific to the vendor's product. Upon the receipt of the completed questionnaire, DRG will schedule for remote (if requested) and on-site testing sessions for the CPE. The final preparation includes a conference call between DRG, test engineers and the vendor. During this meeting, any last minute details on both the logistical and technical matters will be reviewed.

During testing, it may be necessary to make modifications or adjustments to the vendor's equipment. Hence, it is necessary that a vendor representative be present during on-site testing.

At the successful completion of a test session, a test report will be generated, which will be reviewed at AT&T Bell Laboratories. The vendor is provided with a copy of the report. A survey form is attached to the report, which will be used by DRG in its continuing goal of improving the quality of the testing process.

8. SUMMARY

The AT&T ISDN PRI testing program, as with other AT&T testing programs, is part of AT&T's objective of supporting vendors to build CPEs compatible with AT&T Switched Network interfaces. Vendors can contact DRG for help in interpreting the interface specifications and the test plan. AT&T Bell Laboratories will be consulted as needed.

REFERENCES

1. CCITT Study Group XVIII, Recommendation I.412, ISDN User-Network Interfaces, Interface Structures and Access Capabilities.
Recommendation I.431, Primary Rate User-Network Interface - Layer 1 Specification.
Recommendation I.211, Bearer Services Supported by an ISDN, December 1985.
2. AT&T Integrated Services Digital Network (ISDN) Primary Rate Interface and Special Applications Specification, AT&T TR 41459, Issue July 1989; Addenda January 1990; July 1990; November 1990.
3. AT&T Integrated Services Digital Network (ISDN) Primary Rate Interface Specification, AT&T PUB 41449, July 1989.
4. ACCUNET ® T1.5 Service Description and Interface Specification, AT&T TR 62411, December 1990.
5. Requirements for Interfacing Digital Terminal Equipment to Services Employing the Extended Superframe Format, AT&T TR 54016, September 1989.

CONTENTS

APPENDIX A - LAYER 1 TEST PLAN

A.1 SCOPE OF THIS APPENDIX

The physical layer protocols for the AT&T ISDN PRI are specified in TR 62411 and TR 54016. This appendix describes the procedure for obtaining a compatibility statement for the physical layer.

A.2 ORGANIZATION

This appendix describes the tests used to verify conformance to the Layer 1 procedures of the AT&T ISDN PRI by a CPE. These tests are derived from the ACCUNET® T1.5 and ACCUNET® ESF compatibility test plans which are also included in this appendix. Also, it has been observed that it is helpful if a vendor prepares the IUT for the input jitter tolerance test. Two ways of mechanizing the detection of errors, is also included in this appendix.

A.3 GLOSSARY OF TERMS

AMI	Alternate Mark Inversion
B8ZS	Bit 8 Zero Suppression
CSU	Channel Service Unit
QRTS	Quasi-Random Test Signal
ZCS	Zero Code Suppression

A.4 REQUIREMENT FOR COMPATIBILITY

The IUT must be compatible with either SF or ESF framing, with ESF being the preferred choice. AT&T encourages vendors to provide both, with the choice being a user-selected option.

The IUT must be compatible with AMI line coding, but AT&T highly recommends that the vendor also provide a B8ZS alternative, as a user-selected option. Also note that B8ZS is a **necessary** requirement for clear channel (i.e. unrestricted) operation at digital data rates of 64 Kbps or higher.

Among ACCUNET® T1.5 compatibility tests, the following tests are mandatory for PRI Layer 1 compatibility:

— **Network connector test.** The result is for record only. It is not required for pass.

— **Electrical characteristics test.**

— **Framing test.**

— **Reframe time test.**

— **Input jitter tolerance test.**

— **Output jitter test.**

— **Transfer jitter test.**

— **Synchronization tests.** (Time interval error test is required only for an IUT that transfers timing.)

— **DS-0 channelization test.**

— **Alarm test.**

— **Very severe burst error test.**

— **Pulse density test.**

— **Operational test.**

— **Inband loopback test.**

The last three test are optional if the IUT is intended to operate through an approved CSU.

In addition, if ESF framing is supported, some or all of the tests in Attachment 2, "ACCUNET® ESF Compatibility Tests Procedures," apply. Among these tests, the following are mandatory for an IUT which supports ESF framing:

— **Framing test.**

— **CRC test.**

— **Yellow alarm test.**

— **Alarm indication signal (Blue signal) test.**

If the IUT supports the enhanced facility maintenance features, it must pass all tests in Attachment 2.

A.5 Preparation of IUT for Input Jitter Tolerance Test

The input jitter tolerance test measures the IUT's ability to recover the digital information present in a jittered, Quasi-Random Test Signal (QRTS).[1] The test signal will occupy the entire DS-1 bandwidth (except for the framing bits), and will not have any particular relationship to the constituent time slots into which a DS-1 carrier is normally divided. The QRTS source will use Alternate Mark Inversion (AMI) line coding *without* Zero Code Suppression (ZCS). (B8ZS is not allowed.) SF[2] framing must be used. No alternative test signal is acceptable.

To measure the input jitter tolerance, the IUT is required to loop time off the test signal. Sinusoidal jitter is impressed on the test signal at various frequencies. The level of applied jitter at which the IUT can no longer recover the test signal without errors is the IUT's input jitter tolerance. A mechanized means of detecting the onset of errors is useful for ISDN equipments. Typically, we can detect the onset of errors using one of the following methods:

1. The IUT loops the input signal back to the interface where it was applied. The loop-back must be implemented within the IUT at a point beyond the signal-recovery and clock-recovery circuits. Although the framing bits may be shifted in the output signal, the remaining 192 bits in each frame should be unaltered. This implies that (a) the IUT must loop all 24 of the constituent channels back, (b) the IUT must impose equal delay on each of the 24 constituent channels, and (c) the IUT must not employ Zero Code Suppression (ZCS), digital loss, or any other mechanism which may alter the integrity of the data. Any errors detected in the output signal indicate that the applied jitter is beyond the IUT's input jitter tolerance.
2. The IUT routes the input signal to a second DS-1 interface. This method may circumvent two potential obstacles:
    a. While the IUT's administrative software may rule out looping channels on themselves (as with the first method), this method may make more sense to the IUT's administrative software.
    b. If the IUT cannot produce AMI line coding without ZCS, the first method can not be made to work since the IUT's ZCS mechanism would alter some bits to enforce the pulse density constraint. However, it is acceptable for the IUT to recover the AMI-coded QRTS signal and then retransmit on another DS-1 interface using B8ZS line coding.

1. The QRTS is defined in TR 62411.
2. Older documents refer to Superframe Format (SF) framing and formatting as D3, D4, or D3/D4.

As with method #1, the connection through the IUT must be made in such a way that proper operation of the signal recovery and clock recovery circuits can be observed. Although the framing bits may be shifted in the output signal, the remaining 192 bits in each frame should be unaltered. Again, all 24 channels must be switched through, each with equal delay. Any errors detected in the output signal indicate that the applied jitter is beyond the IUT's input jitter tolerance.

It is acceptable for the vendor to make temporary circuit changes in the IUT so that one of these methods can be used, such that the signal recovery and clock recovery circuits remain unaltered. If such changes are needed, the vendor should make them and verify their effectiveness before presenting the IUT for testing.

ATTACHMENT 1 - ACCUNET® T1.5, CCR, and M24 Compatibility Test Procedure

1. Introduction

This document gives a brief description of the tests for compatibility testing of Customer Premises Equipment (CPE) with the AT&T ACCUNET® family of 1.5 Mbps digital services: T1.5, CCR, and M24, complying with Technical Reference PUB 62411 December 1988.

2. Network Connector Test

An inspection of the CPE is made to determine if one or more of the approved network connectors is provided.

3. Electrical Characteristics Test

To function properly with the network the CPE is required to transmit a signal with a specified waveform. A check is made to confirm that the output signal of the CPE conforms to the pulse waveform template. The power levels at 772 KHz and 1.544 MHz of an all ones signal transmitted by the CPE is also checked.

4. Framing Test

Two types of framing patterns are currently supported by AT&T, SF[3] and ESF. This set of tests check for the presense of either the SF pattern or the ESF pattern. Also, for the case of ESF, the CRC-6 is checked. The test of the 4 kb/s data link is made as a part of the ESF compatibility test.

5. Reframe Time Test

It is desirable that the CPE regains framing synchronization within 50 msecs. This test determines if the CPE reframes in that time frame on the incoming DS-1 signal.

6. Pulse Density Test

Long sequences of digital zeroes or signals with a high density of zeroes transmitted from CPE may create synchronization losses within the network. Hence CPE are restricted in the number of consecutive zeroes they may transmit. This test determines whether the amount of zeroes transmitted by the CPE exceed Technical Reference 62411 requirements.

7. Operational Test

The pulse characteristics of the signal from the customer's CPE to the network must be such that it will provide sufficient drive and stability to the network. This test determines whether the CPE output signal does provide sufficient drive and stability to the network and that the CPE's output signal can be recovered and regenerated by the network's first line repeater.

8. B8ZS Test

AT&T is committed to converting the network to support B8ZS to ensure sufficient pulse density to maintain network stability. This test is made to determine if the CPE will properly encode, decode, or transmit without changing the B8ZS symbol.

---

3. Older documents refer to Superframe Format (SF) framing and formatting as D3, D4, or D3/D4.

9. 64 kb/s Clear Channel Capability Test

The 64 kb/s clear channel capability implies that the CPE doesn't use any inband signaling and uses B8ZS to replace zero bytes. This test checks the ability of the CPE to support this capability.

10. Delay Tolerance Test

Terrestrial 1.5 Mb/s circuits will experience a one way absolute time delay of no more than 60 msec. For Satellite circuits the round trip time delay can be up to 700 msec. This test determines if the CPE can tolerate a time delay of the DS-1 signal between two end points.

11. Inband Loopback Test

AT&T maintenance personnel use loopbacks to segment and test line performance. This test determines whether the CPE will respond to the loopback up and loopback down signals.

12. Input Jitter Tolerance Test

Jitter is the random phase modulation of the digital signal. It is generated by digital regenerators and digital multiplexers in digital transmission systems. The CPE is tested to determine the amount of input jitter it will tolerate while operating in a loop timing mode. If the CPE is intended to transfer or terminate timing, the input jitter tolerance of the clock recovery circuit is also tested.

13. Output Jitter Test

Excessively jittered signals from CPE may cause the network to lose signal frame synchronization. The purpose of the output jitter test is to ensure that the CPE does not introduce high jitter levels into the network.

14. Transfer Jitter Test

The jitter component of the recovered clock is transferred to the outgoing signal by the CPE operating in loop timing mode. The purpose of the transfer jitter test is determine how much the CPE attenuates the transfer of the input jitter to the output DS-1 signal while operating in loop timing mode.

15. DACS Compatibility Test

The Digital Access and Cross-Connect System (DACS) is a digital terminal designed for the use with DS-1 facilities. DACS is currently widely used in the AT&T Switched Network. The incoming DS-1 signal can be cross-connected in three modes: the DS-1 level, the DS-0 level, and the bit compression multiplexer bundle format. The CPE is tested to determine if it will provide the proper framing and channelized signal to the DACS, and for its operability with DACS.

16. Superframe Synchronization Test

The DACS doesn't maintain superframe integrity. It will synchronize on the incoming DS-1 signal and generate a new framing pattern on the out going DS-1 signal. This test determines if the CPE is dependent on superframe alignment for recovering information from the digital signal.

17. Synchronization Tests

The task of keeping all the entities in a digitally interconnected network operating at the same frequency so that no information is lost or repeated is referred to as network synchronization. CPE clocks must meet the following synchronization requirements to ensure error free operation. All CPE clocks that terminate or transfer timing must be stratifiable according to stratum levels defined in

Technical Reference 62411. CPE clocks that terminate timing must meet free run accuracy and pull-in range requirements. CPE clocks that transfer timing must meet accuracy and pull-in range requirements as well as maximum time interval error requirements. Stratum 2 and 3 clocks must have duplicated timing hardware and an external timing port available.

17.1 Holdover or Free Run Accuracy Test

The holdover or free run accuracy test checks the CPE clock's stability during extended loss of both primary and secondary timing references.

17.2 Clock Pull-in Range

The clock pull-in range test measures the frequency over which a CPE clock will be able to synchronize and maintain lock.

17.3 Time Interval Error Tests

The time interval error tests measure the maximum phase movement and phase slope of a CPE clock when the clock undergoes any rearrangement activity such as reference switching or changeover of active synchronization hardware.

18. DS-0 Channelization Test

The network assumes that the DS-0 signals are placed in the DS-1 signal using a specified format. This test determines if the CPE conforms to the channelization of DS-0 signal in the DS-1 signal. A check is made to see if the DS-0 signal are composed of 24 eight bit time slots and placed in sequential order in the DS-1 signal stream.

19. A and B Bit Signaling Test

A and B robbed bit signaling is used for trunk supervision on a DS-0 channel. This test determines that the CPE properly implements and responds to the A and B bit signals for trunk supervision.

20. Alarm Test

Alarms are used by the network and the CPE to indicate trouble with the circuit. Two sets of alarms are currently used in ACCUNET® T1.5, red and yellow alarms for DS-1 terminating equipment[4] and blue alarm for non-DS-1 terminating equipment. For DS-1 terminating CPE, the CPE is tested to to show local red alarm and transmit remote yellow alarm to the network for loss of synchronization with the incoming DS-1 signal. Non-DS-1 terminating CPE should transmit blue alarm when it loses the dropside incoming DS-1 signal.

21. Minimum Acceptable Performance Test

The voice frequency channel units of the CPE are tested for minimum acceptable performance limits for voice grade transmission service.

22. Digital Error Performance Test

The error performance of the DS-1 signal is varied to determine the effect on subrate digital data performance. The DS-0 channels are tested for both gaussian and bursty error rates.

---

4. In the context of this document, DS-1 terminating equipment refers to CPE that terminates the framing channel.

23. Very Severe Burst Error Test

The purpose of the very severe burst error test is to check that the CPE can respond to severe burst errors properly. The CPE should not initiate network or local alarms for error events of 2.5 second or shorter duration. Graceful recovery from these burst error events is essential to customer satisfaction. The CPE is tested for burst error rates up to $5*10^{-1}$.

ATTACHMENT 2 - ACCUNET® ESF Compatibility Test Procedure

1. Introduction

Extended Superframe Format (ESF) is a DS-1 Superframe structure that permits more signaling states to be defined (16) than in previous DS-1 Superframe structures. ESF also facilitates maintenance testing and line quality monitoring without interrupting customer information flow. For these reasons, DS-1 circuits are being converted from D4[5] Framing to ESF.

As DS-1 access lines convert from SF Framing to ESF, Customer Premises Equipment (CPE) must also convert from SF Framing to ESF. There is a need to verify that the CPE vendor's implementation of ESF is compatible with AT&T's implementation of ESF. For this reason the ESF Compatibility Tests were devised. This document describes the Extended Superframe Format (ESF) Compatibility Tests.

2. Part I: Frame Structure Tests

2.1 Framing Test

This test determines whether the CPE is compatible with the network ESF framer. The CPE is connected to the test bed and unimpaired transmission conditions are created between the two. Transmission conditions are observed.

2.2 CRC Test

The CPE's capability to correctly calculate and decode Cyclic Redundancy Codes (CRC-6) is tested during this test. A repeating bit pattern is input to the dropside of the CPE. On the network side, the test bed verifies whether the CPE generates the correct CRC for this bit pattern.

3. Part II: Maintenance and Alarm Tests

3.1 Yellow Alarm

DS1 terminal equipment[6] must transmit a yellow alarm after a fixed period of time if the equipment is unable to synchronize on the DS1 signal from the network. The test bed sends an unsynchronizable signal to the CPE and monitors the signal from the CPE for the yellow alarm. (The specific criteria for declaring yellow alarm are described in the ACCUNET® T1.5 interface specification, TR62411 [3]).

DS1 through equipment such as CSUs should pass a yellow alarm signal in either direction.

3.2 Alarm Indication Signal (Blue Signal)

If the CPE under test is not DS-1 terminal equipment, it must transmit an Alarm Indication Signal (AIS) after detecting a Loss of Signal (LOS) from the terminal equipment on its DTE side. (For more information on AIS and LOS, see TR62411 [3]).

3.3 Loopbacks

The CPE should pass in and out of payload loopback (PLB) and line loopback (LLB) states as commanded by the network. The test bed sends in-band and maintenance message loopback signals to the CPE and monitors the CPE's response. (For more information on loopbacks, see TR62411

5. Older documents refer to Superframe Format (SF) framing and formatting as D3, D4, or D3/D4.

6. In this document, DS-1 terminal equipment refers to customer premises equipment which terminates the framing channel.

[3]).

4. Part III: Protocol Tests

The Data Link segment of the ESF carries maintenance messages between the network and the CPE. The protocol used by the data link is a simplified X.25 link layer (level 2) with a byte oriented information layer. For more information on the maintenance message set, see TR54016 [1].)

These tests determine the CPE's capability to receive, decode, and respond to maintenance messages from the network. The CPE should transmit appropriate responses to network requests. The responses must be sent less than 50ms after receiving the requests. The test bed transmits maintenance messages to the CPE and monitors the responses. Responses must be correct in format, appropriate in content, and timely.

5. Part IV: Parameter Collection and Storage Tests

The CPE must record logical and framing transmission errors. These error records become criteria for determining Errored Seconds (ES), Unavailable Seconds (UAS), and if the enhanced parameter set is supported, Severely Errored Seconds (SES), Bursty Errored Seconds (BES), Loss of Frame Count (LOFC), and Controlled Slip Seconds (CSS). Upon request, the CPE sends these error records and error-related parameters to the network over the data link in the form of maintenance messages.

These tests ascertain the CPE's ability to collect and store error and error-related parameters. The CPE's ability to zero parameter counters is also tested during the parameter collection tests.

5.1 ES/UAS Test

The CPE must collect both logical and framing errors and record ESs and UASs correctly. The test bed inserts both logical and framing errors into the transmitted signal. The test bed then requests the Performance Data from the CPE and verifies that the errors have been correctly converted into ESs and FSs, stored in the proper ES/FS storage intervals, and accumulated in the ES/FS 24-hr. totals.

5.2 Errored ESF Count Test

These tests ascertain the CPE's ESF error event collection, storage and reporting capabilities. The test bed inserts Gaussian and Burst Errors into the transmitted signal. The test bed then requests the Errored ESF Count from the CPE and verifies that the errors have been collected properly. Several test runs with varying Gaussian and Burst Error rates are made.

5.3 Bursty Errored Seconds (BES) Count Test

This test ascertains the CPE's BES storage and reporting capabilities. The test bed inserts Gaussian and Burst Errors into the transmitted signal. The test bed then requests the BES Count from the CPE and verifies that the errors have been collected properly. This test applies only to CPE that support the enhanced parameter set.

5.4 Severely Errored Seconds (SES) Count Test

This test ascertains the CPE's SES storage and reporting capabilities. The test bed inserts CRC6 and framing errors into the transmitted signal. The test bed then requests the SES Count from the CPE and verifies that the errors have been collected properly. This test applies only to CPE that support the enhanced parameter set.

5.5 Loss of Frame Count (LOFC) Test

This test ascertains the CPE's LOFC storage and reporting capabilities. The test bed inserts framing errors into the transmitted signal. The test bed then requests the LOFC from the CPE and verifies that the errors have been collected properly. This test applies only to CPE that support the enhanced parameter set.

5.6 Controlled Slip Seconds (CSS) Count Test

This test ascertains the CPE's CSS storage and reporting capabilities. To force controlled slips to occur in the CPE, the test bed inserts a fixed frequency offset into the transmitted signal. The test bed then requests the CSS Count from the CPE and verifies that the errors have been collected properly. Implementation of this parameter is optional and applies only to CPE that support the enhanced parameter set.

6. Part V: Maximum Error Check Test

This test verifies proper ES collection and storage, checks that CPE counters do not reset after they reach their highest binary value ($2^{16} - 1$), and checks for the proper maintenance of 15 minute interval boundaries.

The test bed inserts errors at a high rate into the transmitted signal for at least 19 hours. The test bed then requests the Performance and Errored ESF data from the CPE. The test bed verifies that counters did not reset during the test run and that ES information was properly stored for each 15 minute interval.

CONTENTS

LIST OF TABLES

APPENDIX B - LAYER 2 TEST PLAN

B.1 SCOPE OF THIS APPENDIX

This appendix describes the test scripts used to verify conformance to the Layer 2 procedures of the AT&T ISDN PRI by a CPE. These scripts specify tests to verify protocol procedures involving valid, inopportune and invalid frames. They also verify the implementation of proper rotation of frame sequence numbers, address checking and proper handling of parameters such as timers and window size.

At the data link layer, TR 41449 and TR 41459 specify a subset of the Q.921 (LAPD) protocol. Procedures in Q.921 which pertain to automatic Terminal Endpoint Identifier (TEI) management are not used. Therefore, Q.921 states S1, S2 and S3 will not exist and hence no tests are provided for these states. Also, UI and XID frames are not supported on a PRI interface and therefore must be ignored by the IUT.

B.2 ORGANIZATION

What follows is a description of terminology used in this appendix, an explanation of the presentation of the scripts, conditions for selection of tests that will be executed, followed by the Layer 2 scripts themselves.

B.3 GLOSSARY OF TERMS

CONTEST-PRI	CONformance TEST system for PRI
DISC	Disconnect frame
DM	Disconnect Mode frame
FRMR	Frame Reject frame
I	Information frame
IUT	Implementation Under Test
PRI	Primary Rate Interface
REJ	Reject frame
RNR	Receiver Not Ready frame
RR	Receiver Ready frame
SABME	Set Asynchronous Balance Mode Extended frame
SAPI	Service Access Point Identifier
TEI	Terminal End-point Identifier
UA	Unnumbered Acknowledgement frame
UI	Unnumbered Information frame
XID	eXchange IDentification frame
XX	undefined/unrecognizable frame

B.4 DESCRIPTION OF SCRIPT PRESENTATION

The Layer 2 test scripts are divided into 21 test groups, with several individual tests contained in each test group. CONTEST-PRI has been designed so that tests within a test group can be executed individually. Each test group contains a table that summarizes the purpose of the tests included in that test group.

The Layer 2 test scripts are presented in tabular format as in the following example.

TEST M_STATE5:3

Purpose: Inform peer that data link is not in the multiple frame operation mode.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----DM F=1
UA----> F=1	} (S7.0)
DISC--->> P=1	
	<----UA F=1

The table indicates the frames CONTEST-PRI and the IUT will send. The status of P/F bit is also noted. The state the IUT is expected to be in after receiving or sending a frame is noted in parentheses, preceded by the letter S, (e.g., S7.0).

When CONTEST-PRI starts execution of a specified test, it will check the status of the data link layer and bring the link to Multiple Frame Established (Normal) state. (This procedure is described in Section B.7.) After that, CONTEST-PRI will exchange a certain sequence of frames with the IUT. This exchange is the required *Preamble* for the selected test, which will move the IUT into the required state for the test. Then CONTEST-PRI will follow through the exchange of frames identified for the selected test.

B.5 TESTING REQUIREMENTS ON IUT

Testing in accordance with this test plan requires that the IUT have provisions to disable or relax some of the protocol requirements during testing. In particular, for Layer 2 testing, the IUT must disable Layer 3 operation during the execution of Layer 2 tests.

B.6 SELECTION OF TESTS

Based on the capabilities and the features supported by the IUT, a subset of the tests will be selected and the IUT should successfully complete all the selected tests. This section lists the conditions under which the tests will be selected.

1. All IUTs will be tested using the first 7 test groups that are specified in Section B.8.
2. If the IUT can remain in TEI Assigned State long enough for testing purposes (approximately 2 seconds), the tests in Section B.9.1 will be executed.
3. If the IUT can be forced to send a DISC frame on demand, then the tests in Section B.9.2 will be executed.
4. If the IUT can be forced to send a RNR frame while in Multiple Frame Established state, then the tests in Section B.9.3.1 will be executed.
5. If the IUT can be forced to send a RNR frame while in Timer Recovery state, then the tests in Section B.9.3.2 will be executed.
6. If the IUT can be forced to send Information frames on demand, then the tests in Section B.9.4 will be executed.
7. If the IUT supports the procedure for unimplemented frames that is specified in TR 41449, then the tests in Sections B.9.5.1 will be executed.

8. If the IUT supports the procedure for unimplemented frames that is specified in TR 41449, and can remain in TEI Assigned State long enough, then the tests in Section B.9.5.2 will be executed.
9. If the IUT supports the procedure for unimplemented frames that is specified in TR 41449, and can send a DISC frame on demand, then the tests in Section B.9.5.3 will be executed.
10. If the IUT supports the procedure for unimplemented frames that is specified in TR 41449, and can send a RNR frame in Multiple Frame Established State, then the tests in Section B.9.5.4 will be executed.
11. If the IUT supports the procedure for unimplemented frames that is specified in TR 41449, and can send a RNR frame in Timer Recovery State, then the tests in Section B.9.5.5 will be executed.

The selection rules listed above are represented in the following table:

TABLE 1. Selection of Tests Based on the Supported Capabilities

	CAPABILITY	TEST GROUP
	Basic	M_STATE5 M_STATE7.0 M_STATE7.1 M_STATE8.4 M_STATE8.5 M_RCVSEQ M_TIMER
	Stay in State 4	C_STATE4
	Send a DISC Frame	C_STATE6
	Send RNR frame in State 7.2	C_STATE7.2
	Send RNR frame in State 8.6	C_STATE8.6
	Send I frame on demand	C_SNDSEQ
Handling of unimplemented frames	Basic	C_UIXID:1 - C_UIXID:10
	Stay in State 4	C_UIXID:11, C_UIXID:12
	Send a DISC Frame	C_UIXID:13, C_UIXID:14
	Send RNR frame in State 7.2	C_UIXID:15, C_UIXID:16
	Send RNR frame in State 8.6	C_UIXID:17, C_UIXID:18

B.7 STATUS CHECK OF DATA LINK LAYER

This frame sequence is executed at the beginning of each test in order to determine what state the IUT is in and bring it into Multiple-Frame Established State.

B.7.1 IUT not in Awaiting Establishment State

CONTEST-PRI Sends	*IUT Sends*
RR----> P=1	
	<----RR (S7.0,S7.4,S7.5,S8.0,S8.4,S8.5) F=1
SABME----> P=1	
	<----UA (S7.0) F=1
or	
RR----> P=1	
	<----RNR (S7.2,S7.6,S7.7,S8.2,S8.6,S8.7) F=1
SABME----> P=1	
	<----UA (S7.0) F=1
or	
RR----> P=1	
	<----DM (S4) F=1
SABME----> P=1	
	<----UA (S7.0) F=1

B.7.2 IUT in Awaiting Establishment State

CONTEST-PRI Sends	*IUT Sends*
RR----> P=1	} Ignore (S5.0)
UA----> F=1	} no message (S7.0)
SABME----> P=1	<----UA (S7.0) F=1

B.8 DATA LINK PROCEDURES - MANDATORY

This section contains test cases for basic data link procedures. These tests are mandatory for all IUTs.

B.8.1 Awaiting Establishment - State S5

Purpose of Test	
Test M_STATE5:	*Verify IUT can ...*
1	Transition into Multiple Frame Established State; stop timer T200 and start timer T203
2	Accept peer's request to transition into Multi-Frame Established State while remaining in Awaiting Establishment State
3	Inform peer that data link is not in the multiple frame operation mode
4-10	Ignore inopportune frames
11	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option) **or** ignore
12	Initiate re-establishment upon receipt of a long frame (the FRMR is an implementation option) **or** ignore.
13,14	Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
DM---->	
F=1	
	<----SABME (S5) †
	P=1

† The (S#) denotes the state the IUT will be in after sending this frame

B.8.1.1 Valid Frames

TEST M_STATE5:1

Purpose: Transition into Multiple Frame Established State; stop timer T200 and start timer T203.

CONTEST-PRI Sends	*IUT Sends*
UA----> F=1	} (S7.0)
DISC---> P=1	
	<----UA F=1

TEST M_STATE5:2

Purpose: Accept peer's request to transition into Multi-Frame Established State while remaining in Awaiting Establishment State.

CONTEST-PRI Sends	*IUT Sends*
SABME----> P=1	
	<----UA F=1
UA----> F=1	} (S7.0)
DISC---> P=1	
	<----UA F=1

TEST M_STATE5:3

Purpose: Inform peer that data link is not in the multiple frame operation mode.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----DM F=1
UA----> F=1	} (S7.0)
DISC---> P=1	
	<----UA F=1

B.8.1.2 Inopportune Frames

TEST M_STATE5:4

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
UA---->	}Ignore
F=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:5

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
DM---->	}Ignore
F=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:6

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	}Ignore
P=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:7

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	}Ignore
P=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:8

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	}Ignore
P=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:9

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	}Ignore
F=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
I---->	}Ignore
P=0	
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

B.8.1.3 Invalid Frames

TEST M_STATE5:11

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option) **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR F=0
	<----SABME P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*---->	
	<----SABME P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*----> DISC----> P=1	} Ignore
	<----DM F=1

* XX is an undefined/unrecognizable frame

TEST M_STATE5:12

Purpose: Initiate re-establishment upon receipt of a long frame (the FRMR is an implementation option) **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR F=0
	<----SABME P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*---->	
	<----SABME P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*----> DISC----> P=1	} Ignore
	<----DM F=1

* XX is a long UA frame with F=1

TEST M_STATE5:13

Purpose: Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	}Ignore
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE5:14

Purpose: Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	} Ignore
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

B.8.2 Multiple Frame Established (Normal) - State S7.0

Purpose of Test	
Test M_STATE7.0:	*Verify IUT can ...*
1	Restart the data link when already in multiple frame operation mode.
2	Disconnect the data link
3	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
4	Acknowledge receipt of a valid Information frame
5	Respond to a normal status check of the data link
6	Report that it is not in an *own receiver busy* condition
7	Report its current V(R) value
8	Initiate re-establishment, knowing that retransmission of a frame will not clear an error condition.
9	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
10-12	Ignore inopportune frames
13	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
14	Initiate re-establishment upon receipt of a frame with a long RR frame (the FRMR is an implementation option).
15	Initiate re-establishment upon receipt of a frame with a long SABME frame (the FRMR is an implementation option).
16	Initiate re-establishment upon receipt of a frame with a long DISC frame (the FRMR is an implementation option).
17	Reject an I frame with an invalid N(S) and report the proper value of its V(R).
18	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)
19	Initiate re-establishment upon receipt of an I frame with invalid N(S) and N(R); IUT response to the I frame depends on the order in which the N(S) and N(R) are checked (FRMR and REJ are both implementation options)
20	Ignore a command frame that is too short while in multiple frame established state
21	Ignore a frame with a non-zero TEI while in multiple frame established state
22	Ignore a frame with a non-zero SAPI while in multiple frame established state
23	Reject a command frame with an improper C/R bit; initiate reestablishment of data link

All the tests in this section will execute the status check procedure described in Section B.7. These tests do not require preamble.

B.8.2.1 Valid Frames

TEST M_STATE7.0:1

Purpose: Restart the data link when already in multiple frame operation mode.

CONTEST-PRI Sends	*IUT Sends*
SABME----> P=1	
	<----UA F=1

TEST M_STATE7.0:2

Purpose: Disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----UA (S4) † F=1

TEST M_STATE7.0:3

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM----> F=1	
	<----SABME (S5) P=1
UA----> F=1	
or	
DM----> F=1 DISC----> P=1	}Ignore
	<----UA F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

TEST M_STATE7.0:4

Purpose: Acknowledge receipt of a valid Information frame.

CONTEST-PRI Sends	*IUT Sends*
I---->	
P=1	
	<----RR
	F=1

TEST M_STATE7.0:5

Purpose: Respond to a normal status check of the data link.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RR
	F=1

TEST M_STATE7.0:6

Purpose: Report that it is not in an *own receiver busy* condition.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR (S7.4)
	F=1
RR---->	
P=0	

TEST M_STATE7.0:7

Purpose: Report its current V(R) value.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RR
	F=1

TEST M_STATE7.0:8

Purpose: Initiate re-establishment, knowing that retransmission of a frame will not clear an error condition.

CONTEST-PRI Sends	*IUT Sends*
FRMR----> F=1	
	<----SABME (S5) P=1
UA----> F=1	

TEST M_STATE7.0:9

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA----> F=1	
	<----SABME (S5) P=1
UA----> F=1	
or	
UA----> F=1	}Ignore

B.8.2.2 Inopportune Frames

TEST M_STATE7.0:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR----> P=0	} (S7.4)
RR----> P=0	}
DISC----> P=1	
	<----UA F=1

TEST M_STATE7.0:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	} Ignore
P=0	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE7.0:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	} Ignore
P=0	
DISC---->	
P=1	
	<----UA
	F=1

B.8.2.3 Invalid Frames

TEST M_STATE7.0:13

Purpose: Initiate re-establishment upon receipt of a command frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX † ---->	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	
F=1	
or	
XX † ---->	
	<----SABME (S5)
	P=1
UA---->	
F=1	

† XX is an undefined/unrecognizable frame

TEST M_STATE7.0:14

Purpose: Initiate re-establishment upon receipt of a frame with a long RR frame (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX † ---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
XX † ---->	
	<----SABME (S5) P=1
UA----> F=1	

TEST M_STATE7.0:15

Purpose: Initiate re-establishment upon receipt of a frame with a long SABME frame (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX ‡ ---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
XX ‡ ---->	
	<----SABME (S5) P=1
UA----> F=1	

† XX is a long RR frame.

‡ XX is a long SABME frame.

TEST M_STATE7.0:16

Purpose: Initiate re-establishment upon receipt of a frame with a long DISC frame (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX † ---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
XX † ---->	
	<----SABME (S5) P=1
UA----> F=1	

TEST M_STATE7.0:17

Purpose: Reject an I frame with an invalid N(S) and report the proper value of its V(R).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))----> P=0	
	<----REJ (S7.1) F=0

† XX is a long DISC frame.

TEST M_STATE7.0:18

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))----> P=0	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
I (with invalid N(R))----> P=0	
	<----SABME (S5) P=1
UA----> F=1	

TEST M_STATE7.0:19

Purpose: Initiate re-establishment upon receipt of an I frame with invalid N(S) and N(R); IUT response to the I frame depends on the order in which the N(S) and N(R) are checked (FRMR and REJ are both implementation options).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R) & N(S))----> P=0	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
I (with invalid N(R) & N(S))----> P=0	
	<----SABME (S5) P=1
UA----> F=1	
or	
I (with invalid N(R) & N(S))----> P=0	
	<----REJ F=0
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
or	
I (with invalid N(R) & N(S))----> P=0	
	<----REJ F=0
	<----SABME (S5) P=1
UA----> F=1	

CONTEST-PRI Sends	*IUT Sends*

TEST M_STATE7.0:20

Purpose: Ignore a command frame that is too short while in multiple frame established state.

CONTEST-PRI Sends	*IUT Sends*
Command Frame,too short ----> P=0	
	} Ignore
DISC----> P=1	
	<----UA F=1

TEST M_STATE7.0:21

Purpose: Ignore a frame with a non-zero TEI while in multiple frame established state.

CONTEST-PRI Sends	*IUT Sends*
Command Frame with incorrect TEI----> P=1	
	} Ignore
RR----> P=1	
	<----RR F=1

TEST M_STATE7.0:22

Purpose: Ignore a frame with a non-zero SAPI while in multiple frame established state.

CONTEST-PRI Sends	*IUT Sends*
Command Frame with incorrect SAPI----> P=1	
	} Ignore
RR----> P=1	
	<----RR F=1

TEST M_STATE7.0:23

Purpose: Reject a command frame with an improper C/R bit; initiate reestablishment of data link.

CONTEST-PRI Sends	*IUT Sends*
Command Frame with incorrect C/R*----> P=1	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	
RR----> P=1	
	<----RR F=1
or	
Command Frame with incorrect C/R*----> P=1	
	<----SABME (S5) P=1
UA----> F=1	
RR----> P=1	
	<----RR F=1

* This is considered a control field error. An example of such a frame would be a DISC response.

B.8.3 Multiple Frame Established (Reject) - State S7.1

Purpose of Test	
Test M_STATE7.1:	*Verify IUT can ...*
1	Clear *reject exception* condition and re-establish the data link
2	Clear *reject exception* condition and disconnect the data link
3	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
4	Acknowledge receipt of a valid Information frame and transition into normal multiple frame operation mode
5	Respond to a normal status check of the data link and report its current V(S) and V(R) values.
6	Report that it is not in an *own receiver busy* condition
7	Report its current V(R) value
8	Initiate re-establishment, knowing that retransmission of the frame will not clear an error condition.
9	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
10-12	Ignore inopportune frames
13	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
14	Reject an I frame with an invalid N(S) and report the value of its V(R).
15	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
	<----REJ (S7.1) †

† The (S#) denotes the new state the IUT will be in after sending this frame

B.8.3.1 Valid Frames

TEST M_STATE7.1:1

Purpose: Clear *reject exception* condition and re-establish the data link.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----UA (S7.0)
	F=1

TEST M_STATE7.1:2

Purpose: Clear *reject exception* condition and disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC---->	
P=1	
	<----UA (S4)
	F=1

TEST M_STATE7.1:3

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
DM---->	}Ignore
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE7.1:4

Purpose: Acknowledge receipt of a valid Information frame and transition into normal multiple frame operation mode.

CONTEST-PRI Sends	*IUT Sends*
I---->	
P=1	
	<----RR (S7.0)
	F=1

TEST M_STATE7.1:5

Purpose: Respond to a normal status check of the data link and report its current V(S) and V(R) values.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RR
	F=1

TEST M_STATE7.1:6

Purpose: Report that it is not in an *own receiver busy* condition.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR (S7.5)
	F=1
RR---->	
P=0	

TEST M_STATE7.1:7

Purpose: Report its current V(R) value.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RR
	F=1

TEST M_STATE7.1:8

Purpose: Initiate re-establishment, knowing that retransmission of the frame will not clear an error condition.

CONTEST-PRI Sends	*IUT Sends*
FRMR----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)

TEST M_STATE7.1:9

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
UA----> F=1	}Ignore

B.8.3.2 Inopportune Frames

TEST M_STATE7.1:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR----> P=0	} (S7.5)
RR----> P=0	} (S7.1)
DISC----> P=1	
	<----UA F=1

TEST M_STATE7.1:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	} Ignore
P=0	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE7.1:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	} Ignore
P=0	
DISC---->	
P=1	
	<----UA
	F=1

B.8.3.3 Invalid Frames

TEST M_STATE7.1:13

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
XX*---->	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

* XX is an undefined/unrecognizable frame

TEST M_STATE7.1:14

Purpose: Reject an I frame with an invalid N(S) and report the value of its V(R).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
P=1	<----RR
	F=1

TEST M_STATE7.1:15

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	
P=0	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
I (with invalid N(R))---->	
P=0	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

B.8.4 Timer Recovery (Normal, Peer Receiver Busy) - State S8.4

Purpose of Test	
Test M_STATE8.4:	*Verify IUT can ...*
1	Re-establish the data link and transition into multiple frame established mode
2	Disconnect the data link
3	Exit timer recovery procedure and transition into multiple frame established state
4	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
5	Acknowledge a valid I frame, update its receive state variable V(R)
6-8	Set its V(A) to the value $N(R)_j$; transmit an RR response
9	Initiate re-establishment
10	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
11-13	Ignore inopportune frames
14	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
15	Initiate re-establishment upon receipt of a frame with a long RR frame (the FRMR is an implementation option).
16	Reject an I frame with an invalid N(S) and report the value of its V(R).
17	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RNR----> P=1	
	<----RR F=1
Do Nothing	
	T200 expires
	<----RR (S8.4) † P=1

† The (S#) denotes the new state the IUT will be in after sending this frame

B.8.4.1 Valid Frames

TEST M_STATE8.4:1

Purpose: Re-establish the data link and transition into multiple frame established mode.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----UA (S7.0)
	F=1

TEST M_STATE8.4:2

Purpose: Disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC---->	
P=1	
	<----UA (S4)
	F=1

TEST M_STATE8.4:3

Purpose: Exit timer recovery procedure and transition into multiple frame established state.

CONTEST-PRI Sends	*IUT Sends*
RR---->	}(S7.0)
F=1	
DISC--->	
P=1	
	<----UA
	F=1

TEST M_STATE8.4:4

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
DM----> F=1	}Ignore
DISC----> P=1	
	<----UA F=1

TEST M_STATE8.4:5

Purpose: Acknowledge a valid I frame,update its receive state variable V(R).

CONTEST-PRI Sends	*IUT Sends*
I----> P=1	
	<----RR F=1
RR----> F=1	

TEST M_STATE8.4:6

Purpose: Set its V(A) to the value N(R); transmit an RR response.

CONTEST-PRI Sends	*IUT Sends*
RR----> P=1	
	<----RR F=1
RR----> F=1	

TEST M_STATE8.4:7

Purpose: Set its V(A) to the value N(R); transmit an RR response.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR
	F=1
RR---->	
F=1	

TEST M_STATE8.4:8

Purpose: Set its V(A) to the value N(R); transmit an RR response.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RR
	F=1
RR---->	
F=1	

TEST M_STATE8.4:9

Purpose: Initiate re-establishment.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

TEST M_STATE8.4:10

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
UA---->	}Ignore
F=1	

B.8.4.2 Inopportune Frames

TEST M_STATE8.4:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	} Ignore
P=0	
RR---->	} Ignore
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE8.4:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	} (S8.0)
P=0	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE8.4:13

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	} (S8.0)
P=0	
DISC---->	
P=1	
	<----UA
	F=1

B.8.4.3 Invalid Frames

TEST M_STATE8.4:14

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX † ---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
XX † ---->	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)

TEST M_STATE8.4:15

Purpose: Initiate re-establishment upon receipt of a frame with a long RR frame (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX † ---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
XX † ---->	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)

† XX is an undefined/unrecognizable frame

† XX is a long RR frame.

TEST M_STATE8.4:16

Purpose: Reject an I frame with an invalid N(S) and report the value of its V(R).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
P=0	
	<----REJ (S8.5)
	F=0
RR---->	
F=1	

TEST M_STATE8.4:17

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	
P=0	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
I (with invalid N(R))---->	
P=0	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

B.8.5 Timer Recovery (Reject, Peer Receiver Busy) - State S8.5

Purpose of Test	
Test M_STATE8.5:	*Verify IUT can ...*
1	Re-establish the data link and transition into multiple frame established mode, clearing its *reject* condition
2	Disconnect the data link
3	Exit timer recovery procedures and transition into multiple frame established state
4	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
5	Acknowledge a valid I frame, update its receive state variable V(R)
6-8	Respond to inquiry but don't clear timer recovery condition until get RR (F=1)
9	Initiate re-establishment
10	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
11-13	Ignore inopportune frames
14	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
15	Report the value of its V(R); reject condition is already set
16	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
	<----REJ (S7.1) †
RNR----> P=1	
	<----RR (S7.5) F=1
Do Nothing	
	T200 expires
	<----RR (S8.5) P=1

B.8.5.1 Valid Frames

TEST M_STATE8.5:1

Purpose: Re-establish the data link and transition into multiple frame established mode, clearing its *reject* condition.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----UA (S7.0)
	F=1

TEST M_STATE8.5:2

Purpose: Disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC---->	
P=1	
	<----UA (S4)
	F=1

TEST M_STATE8.5:3

Purpose: Exit timer recovery procedures and transition into multiple frame established state.

CONTEST-PRI Sends	*IUT Sends*
RR---->	}(S7.1)
F=1	
DISC--->	
P=1	
	<----UA
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

TEST M_STATE8.5:4

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
DM---->	}Ignore
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE8.5:5

Purpose: Acknowledge a valid I frame, update its receive state variable V(R).

CONTEST-PRI Sends	*IUT Sends*
I---->	
P=1	
	<----RR (S8.4)
	F=1
RR---->	
F=1	

TEST M_STATE8.5:6

Purpose: Respond to inquiry but don't clear timer recovery condition until get RR (F=1).

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<---RR (S8.1)
	F=1
RR---->	
F=1	

TEST M_STATE8.5:7

Purpose: Respond to inquiry but don't clear timer recovery condition until get RR (F=1).

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR
	F=1
RR---->	
F=1	

TEST M_STATE8.5:8

Purpose: Respond to inquiry but don't clear timer recovery condition until get RR (F=1).

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RR (S8.1)
	F=1
RR---->	
F=1	

TEST M_STATE8.5:9

Purpose: Initiate re-establishment.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

TEST M_STATE8.5:10

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
UA---->	}Ignore
F=1	

B.8.5.2 Inopportune Frames

TEST M_STATE8.5:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	} Ignore
P=0	
RR---->	
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST M_STATE8.5:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	} (S8.1)
P=0	
DISC---->	
P=1	
	<-----UA
	F=1

TEST M_STATE8.5:13

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	} (S8.1)
P=0	
DISC---->	
P=1	
	<----UA
	F=1

B.8.5.3 Invalid Frames

TEST M_STATE8.5:14

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
XX*---->	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)

TEST M_STATE8.5:15

Purpose: Report the value of its V(R); reject condition is already set.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))----> P=1	
	<----RR F=1
RR----> F=1	

* XX is an undefined/unrecognizable frame

TEST M_STATE8.5:16

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with Invalid N(R))---->	
P=0	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
I (with Invalid N(R))---->	
P=0	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

B.8.6 Receive Sequence Number Test

Purpose of Test	
Test M_RCVSEQ:	*Verify IUT can ...*
1	Rotate the value of its Receive State variable V(R), within the proper range (0...127)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RR
	F=1 N(R)=0

TEST M_RCVSEQ:1

Purpose: Rotate the value of its Receive State variable V(R), within the proper range (0...127)

CONTEST-PRI Sends	*IUT Sends*
I---->	
P=1 N(S)=0	
	<----RR
	F=1 N(R)=1
I---->	
P=1 N(S)=1	
	<----RR
	F=1 N(R)=2
.	
.	
.	
I---->	
P=1 N(S)=126	
	<----RR
	F=1 N(R)=127
I---->	
P=1 N(S)=127	
	<----RR
	F=1 N(R)=0
I---->	
P=1 N(S)=0	
	<----RR
	F=1 N(R)=1
I---->	
P=1 N(S)=1	
	<----RR
	F=1 N(R)=2

B.8.7 Timers

Purpose of Test	
Test M_TIMER:	*Verify the IUT's ...*
1	Perform proper T200 timer recovery procedures with *reject exception* condition set
2	T200 value and proper T200 timer expiry
3	N200 value and proper retransmission of a frame
4	T203 value and proper T203 timer expiry
5	Timer expiry recovery procedures in which it will stop timer T203 after its first expiry and start timer T200

All the tests in this section will execute the status check procedure described in Section B.7.

B.8.7.1 Timer T200

TEST M_TIMER:1

Perform proper T200 timer recovery procedures with *reject exception* condition set.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
	<----REJ (S7.1)
RNR---->	
P=1	
	<----RR (S7.5)
	F=1
Do Nothing	
	T200 expires
	<----RR (S8.5)
	P=1
RR---->	} (S7.0)
F=1	

TEST M_TIMER:2

T200 value and proper T200 timer expiry.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR (S7.4)
	F=1
Do Nothing	
	T200 expires
	<----RR (S8.4)
	P=1
RR---->	
F=1	

TEST M_TIMER:3

N200 value and proper retransmission of a frame.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR (S7.4)
	F=1
Do Nothing	
	T200 expires
	<----RR (S8.4)
	P=1
Do Nothing	
	T200 expires
	.
	.
	N200 Times
	.
	<----SABME (S5)
	P=1
UA---->	
F=1	

B.8.7.2 Timer T203

TEST M_TIMER:4

T203 value and proper T203 timer expiry.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----UA
	F=1
Do Nothing	
	T203 expires
	<----RR (S8.0)
	P=1
RR---->	
F=1	

TEST M_TIMER:5

Timer expiry recovery procedures in which it will stop timer T203 after its first expiry and start timer T200.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----UA
	F=1
Do Nothing	
	T203 expires
	<----RR
	P=1
Do Nothing	
	T200 expires
	<----RR (S8.0)
	P=1
RR---->	
F=1	

B.9 CONDITIONAL TESTS

This section contains test cases for optional data link procedures or additional capabilities. The tests will be selected based on the capabilities of the IUT. The exact criterion for selection is indicated at the beginning of each group.

B.9.1 TEI Assigned - State S4 These tests are included for an IUT which can be configured to remain in TEI Assigned State long enough for testing purposes (approximately 2 seconds). Since this is not a required capability, these tests are optional. However, if this feature has been implemented, these tests are mandatory.

Purpose of Test	
Test C_STATE4:	*Verify IUT can ...*
1	Transition into Multi-Frame Established State, if able
2	Acknowledge that data link is already disconnected
3	Report that data link is disconnected by non-acknowledgement of I frame **or** by ignoring it
4	Report status: in a disconnected mode and not ready to receive I frames
5	Report status: in a disconnected mode and don't care if peer is not ready
6	Report status: in a disconnected mode; REJ doesn't initiate error recovery procedures
7-11	Ignore inopportune frames
12,13	Ignore inopportune frames
14	Ignore frame with undefined control bit field
15,16	Ignore errored I frames with poll bit not set
17	Report that data link is disconnected **or** ignore the errored I frame; don't initiate error recovery in a state where multiple frame operation is not possible
18	Ignore a command frame that is too long while in a disconnected mode
19	Ignore a command frame that is too short while in a disconnected state
20	Ignore a frame with a non-zero TEI while in a disconnected mode
21	Ignore a frame with a non-zero SAPI while in a disconnected mode

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT sends*
DISC----> P=1	
	<----UA (S4) † F=1

B.9.1.1 Valid Frames

TEST C_STATE4:1

Purpose: Transition into Multi-Frame Established State.

CONTEST-PRI Sends	*IUT Sends*
SABME----> P=1	
	<----UA (S7.0) F=1
or	
SABME----> P=1	
	<----DM F=1

TEST C_STATE4:2

Purpose: Acknowledge that data link is already disconnected.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----DM F=1

TEST C_STATE4:3

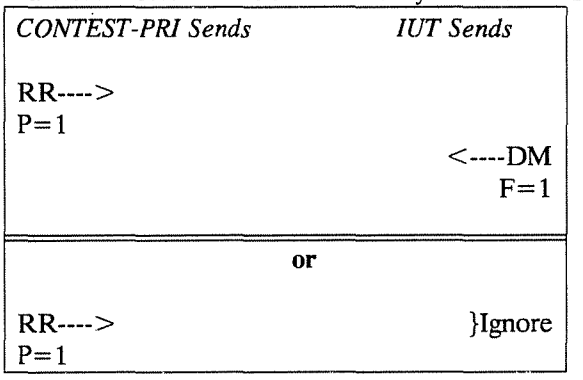
Purpose: Report that data link is disconnected by non-acknowledgement of I frame **or** by ignoring it.

CONTEST-PRI Sends	*IUT Sends*
I----> P=1	
	<----DM F=1
or	
I----> P=1	}Ignore

† The (S#) denotes the new state the IUT will be in after sending this frame.

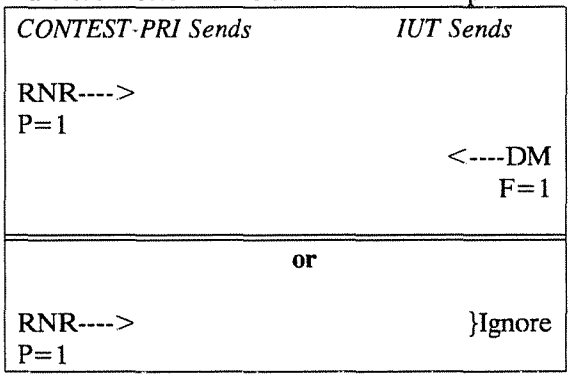
TEST C_STATE4:4

Purpose: Report status: in a disconnected mode and not ready to receive I frames.



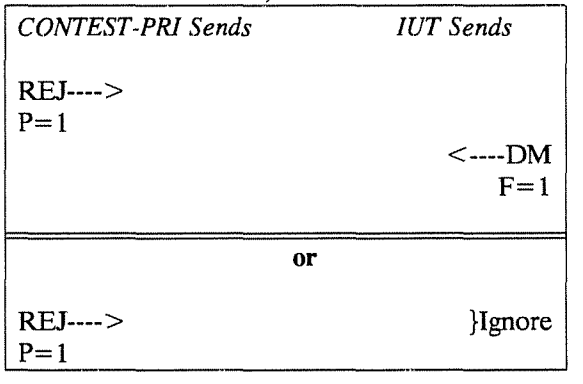
TEST C_STATE4:5

Purpose: Report status: in a disconnected mode and don't care if peer is not ready.



TEST C_STATE4:6

Purpose: Report status: in a disconnected mode; REJ doesn't initiate error recovery procedures.



B.9.1.2 Inopportune Frames

TEST C_STATE4:7

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
UA---->	}Ignore
F=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:8

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
DM---->	}Ignore
F=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:9

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	} Ignore
F=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
I---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:13

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

B.9.1.3 Invalid Frames

TEST C_STATE4:14

Purpose: Ignore frame with undefined control bit field.

CONTEST-PRI Sends	*IUT Sends*
XX*---->	} Ignore
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:15

Purpose: Ignore errored I frames with poll bit not set.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE4:16

Purpose: Ignore errored I frames with poll bit not set.

CONTEST-PRI Sends	*IUT Sends*
I (with Invalid N(R))---->	} Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

* XX is an undefined/unrecognizable frame

TEST C_STATE4:17

Purpose: Report that data link is disconnected **or** ignore the errored I frame; don't initiate error recovery in a state where multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))----> P=1	
	<----DM F=1
or	
I (with Invalid N(R))----> P=1	} Ignore

TEST C_STATE4:18

Purpose: Ignore a command frame that is too long while in a disconnected mode.

CONTEST-PRI Sends	*IUT Sends*
Command Frame, too long ----> P=1	
	} Ignore
DISC----> P=1	
	<----DM F=1

TEST C_STATE4:19

Purpose: Ignore a command frame that is too short while in a disconnected state.

CONTEST-PRI Sends	*IUT Sends*
Command Frame,too short ----> P=0	
	} Ignore
DISC----> P=1	
	<----DM F=1

TEST C_STATE4:20

Purpose: Ignore a frame with a non-zero TEI while in a disconnected mode.

CONTEST-PRI Sends	*IUT Sends*
Command Frame with incorrect TEI----> P=1	
	} Ignore
DISC----> P=1	
	<----DM F=1

TEST C_STATE4:21

Purpose: Ignore a frame with a non-zero SAPI while in a disconnected mode.

CONTEST-PRI Sends	*IUT Sends*
Command Frame with incorrect SAPI----> P=1	
	} Ignore
DISC----> P=1	
	<----DM F=1

B.9.2 Awaiting Release - State S6 These tests are included for an IUT which can be configured to send DISC frames. Since it is not required that an IUT support this feature, these tests are optional. However, if this feature has been implemented, these tests are mandatory.

Purpose of Test	
Test C_STATE6:	*Verify IUT can ...*
1,2	Transition into TEI Assigned State and stop timer T200
3	Report that link is in a state such that multiple frame operation is not possible
4	Acknowledge the DISC in order to permit its peer to disconnect the data link.
5-11	Ignore inopportune frames
12	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option) **or** ignore
13,14	Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
	<----DISC (S6) † P=1

† The (S#) denotes the state the IUT will be in after sending this frame

B.9.2.1 Valid Frames

TEST C_STATE6:1

Purpose: Transition into TEI Assigned State and stop timer T200.

CONTEST-PRI Sends	*IUT Sends*
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:2

Purpose: Transition into TEI Assigned State and stop timer T200.

CONTEST-PRI Sends	*IUT Sends*
DM---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:3

Purpose: Report that link is in a state such that multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
SABME---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:4

Purpose: Acknowledge the DISC in order to permit its peer to disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC---->	
P=1	
	<----UA
	F=1

B.9.2.2 Inopportune Frames

TEST C_STATE6:5

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
UA---->	}Ignore
F=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:6

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
DM---->	}Ignore
F=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:7

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	}Ignore
P=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:8

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR---->	} Ignore
P=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:9

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	}Ignore
P=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	}Ignore
F=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
I---->	}Ignore
P=0	
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

B.9.2.3 Invalid Frames

TEST C_STATE6:12

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option) **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*---->	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
DISC----> P=1	
	<----UA F=1
or	
XX*----> DISC----> P=1	} Ignore
	<----UA F=1

* XX is an undefined/unrecognizable frame

TEST C_STATE6:13

Purpose: Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	}Ignore
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_STATE6:14

Purpose: Ignore errored I frames; don't initiate error recovery in a state where multiple frame operation is not possible.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	} Ignore
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

B.9.3 Sending RNR Frames

B.9.3.1 Multiple Frame Established (Busy) - State S7.2 These tests are included for an IUT which can be configured to send RNR frames. Since it is not required that an IUT support this feature, these tests are optional. However, if this feature has been implemented, these tests are mandatory.

Purpose of Test	
Test C_STATE7.2:	*Verify IUT can ...*
1	Re-establish the data link and clear its *own receiver busy* condition
2	Disconnect the data link
3	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
4	Discard received I frame, after updating its acknowledge state variable V(A); respond with RNR
5-7	Process all received supervisory frames, updating its acknowledge state variable V(A); respond with RNR
8	Initiate re-establishment and clear its *own receiver busy* condition
9	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
10-12	Ignore inopportune frames
13	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
14	Discard an I frame with an invalid N(S) and report the value of its V(R); respond with RNR
15	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
	<----RNR (S7.2) †

B.9.3.1.1 Valid Frames

† The (S#) denotes the new state the IUT will be in after sending this frame

TEST C_STATE7.2:1

Purpose: Re-establish the data link and clear its *own receiver busy* condition.

CONTEST-PRI Sends	*IUT Sends*
SABME----> P=1	
	<----UA (S7.0) F=1

TEST C_STATE7.2:2

Purpose: Disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----UA (S4) F=1

TEST C_STATE7.2:3

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
DM----> F=1 DISC----> P=1	}Ignore
	<----UA F=1

TEST C_STATE7.2:4

Purpose: Discard received I frame, after updating its acknowledge state variable V(A); respond with RNR.

CONTEST-PRI Sends	*IUT Sends*
I----> P=1	
	<----RNR F=1

TEST C_STATE7.2:5

Purpose: Process all received supervisory frames, updating its acknowledge state variable V(A); respond with RNR.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RNR
	F=1

TEST C_STATE7.2:6

Purpose: Process all received supervisory frames, updating its acknowledge state variable V(A); respond with RNR.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RNR (S7.6)
	F=1
RR---->	
P=0	

TEST C_STATE7.2:7

Purpose: Process all received supervisory frames, updating its acknowledge state variable V(A); respond with RNR.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RNR
	F=1

TEST C_STATE7.2:8

Purpose: Initiate re-establishment and clear its *own receiver busy* condition.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

TEST C_STATE7.2:9

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
UA----> F=1	}Ignore

B.9.3.1.2 Inopportune Frames

TEST C_STATE7.2:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR----> P=0	} (S7.6)
RR----> P=0	}
DISC----> P=1	
	<----UA F=1

TEST C_STATE7.2:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR----> P=0	} Ignore
DISC----> P=1	
	<----UA F=1

TEST C_STATE7.2:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	} Ignore
P=0	
DISC---->	
P=1	
	<----UA
	F=1

B.9.3.1.3 Invalid Frames

TEST C_STATE7.2:13

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
XX*---->	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

TEST C_STATE7.2:14

Purpose: Discard an I frame with an invalid N(S) and report the value of its V(R); respond with RNR.

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
P=1	
	<----RNR
	F=1

* XX is an undefined/unrecognizable frame

TEST C_STATE7.2:15

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	
P=0	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
I (with invalid N(R))---->	
P=0	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

B.9.3.2 Timer Recovery (Peer and Own Receiver Busy) - State S8.6 These tests are included for an IUT which can be configured to send RNR frames. Since it is not required that an IUT support this feature, these tests are optional. However, if this feature has been implemented, these tests are mandatory.

Purpose of Test	
Test C_STATE8.6:	*Verify IUT can ...*
1	Re-establish the data link and transition into multiple frame established mode, clearing *own receiver busy* and *reject exception* conditions
2	Disconnect the data link
3	Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore
4	Acknowledge valid I frame, update its receive state variable V(R)
5-7	Respond to supervisory inquiry but don't clear timer recovery condition until get response to its outstanding supervisory command
8	Initiate re-establishment
9	Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA
10-12	Ignore inopportune frames
13	Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option)
14	Report the value of its V(R)
15	Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option)

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
	<----RNR (S7.6)
Do Nothing	T200 expires
	<----RNR (S8.6) †

B.9.3.2.1 Valid Frames

† The (S#) denotes the new state the IUT will be in after sending this frame

TEST C_STATE8.6:1

Purpose: Re-establish the data link and transition into multiple frame established mode, clearing *own receiver busy* and *reject exception* conditions.

CONTEST-PRI Sends	*IUT Sends*
SABME----> P=1	
	<----UA (S7.0) † F=1

TEST C_STATE8.6:2

Purpose: Disconnect the data link.

CONTEST-PRI Sends	*IUT Sends*
DISC----> P=1	
	<----UA (S4) F=1

TEST C_STATE8.6:3

Purpose: Initiate re-establishment upon indication that its peer is not in multiple frame operation mode **or** ignore.

CONTEST-PRI Sends	*IUT Sends*
DM----> F=1	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
DM----> F=1 DISC----> P=1	}Ignore
	<----UA F=1

TEST C_STATE8.6:4

Purpose: Acknowledge valid I frame, update its receive state variable V(R).

CONTEST-PRI Sends	*IUT Sends*
I---->	
P=1	
	<----RNR
	F=1
RR---->	
F=1	

TEST C_STATE8.6:5

Purpose: Respond to supervisory inquiry but don't clear timer recovery condition until get response to its outstanding supervisory command.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RNR (S8.2)
	F=1
RR---->	
F=1	

TEST C_STATE8.6:6

Purpose: Respond to supervisory inquiry but don't clear timer recovery condition until get response to its outstanding supervisory command.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RNR
	F=1
RR---->	
F=1	

TEST C_STATE8.6:7

Purpose: Respond to supervisory inquiry but don't clear timer recovery condition until get response to its outstanding supervisory command.

CONTEST-PRI Sends	*IUT Sends*
REJ---->	
P=1	
	<----RNR (S8.2)
	F=1
RR---->	
F=1	

TEST C_STATE8.6:8

Purpose: Initiate re-establishment.

CONTEST-PRI Sends	*IUT Sends*
FRMR---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

TEST C_STATE8.6:9

Purpose: Initiate re-establishment upon receipt of an unsolicited UA, **or**, ignore the unsolicited UA.

CONTEST-PRI Sends	*IUT Sends*
UA---->	
F=1	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
UA---->	}Ignore
F=1	

B.9.3.2.2 Inopportune Frames

TEST C_STATE8.6:10

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RNR---->	} Ignore
P=0	
RR---->	
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST C_STATE8.6:11

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
RR----> P=0	} (S8.2)
DISC----> P=1	
	<----UA F=1

TEST C_STATE8.6:12

Purpose: Ignore inopportune frames.

CONTEST-PRI Sends	*IUT Sends*
REJ----> P=0	} (S8.2)
DISC----> P=1	
	<----UA F=1

B.9.3.2.3 Invalid Frames

TEST C_STATE8.6:13

Purpose: Initiate re-establishment upon receipt of a frame with an undefined control field (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
XX*---->	
	<----FRMR F=0
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)
or	
XX*---->	
	<----SABME (S5) P=1
UA----> F=1	} (S7.0)

* XX is an undefined/unrecognizable frame

TEST C_STATE8.6:14

Purpose: Report the value of its V(R).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
P=1	
	<----RNR
	F=1
RR---->	
F=1	

TEST C_STATE8.6:15

Purpose: Initiate re-establishment upon receipt of an invalid N(R) value (the FRMR is an implementation option).

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(R))---->	
P=0	
	<----FRMR
	F=0
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	
or	
I (with invalid N(R))---->	
P=0	
	<----SABME (S5)
	P=1
UA---->	} (S7.0)
F=1	

B.9.4 Send Sequence Number Tests These tests are included for an IUT which can be configured to send Information frames on demand. Since it is not required that an IUT support this feature, these tests are optional. However, if this feature has been implemented, these tests are mandatory.

Purpose of Test	
Test C_SNDSEQ:	*Verify IUT can ...*
1	Accept the acknowledgement of <=*k* Information frames with a single RR frame
2	Transmit only *k* I frames without acknowledgement
3	Retransmit I frames starting with the I frame indicated by the N(R) contained in the REJ frame

All the tests in this section will execute the status check procedure described in Section B.7. Additionally, the following preamble will also be executed.

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1	
	<----RR
	F=1 N(R)=0

TEST C_SNDSEQ:1

Purpose: Accept the acknowledgement of <=*k* Information frames with a single RR frame.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1 N(R)=0	
	<----RR
	F=1 N(R)=0
	<----I N(S)=0
	<----I N(S)=1
	<----I N(S)=2
RR---->	
F=0 N(R)=3	
	<----I N(S)=3
	<----I N(S)=4
	<----I N(S)=5
	<----I N(S)=6
	<----I N(S)=7
	<----I N(S)=8
	<----I N(S)=9
RR---->	
F=0 N(R)=10	

TEST C_SNDSEQ:2

Purpose: Transmit only *k* I frames without acknowledgement.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1 N(R)=0	
	<----RR
	F=1 N(R)=0
	<----I N(S)=0
	<----I N(S)=1
	<----I N(S)=2
	<----I N(S)=3
	<----I N(S)=4
	<----I N(S)=5
	<----I N(S)=6
Do Nothing	
	<----RR
	P=1
RR---->	
F=1 N(R)=7	

TEST C_SNDSEQ:3

Purpose: Retransmit I frames starting with the I frame indicated by the N(R) contained in the REJ frame.

CONTEST-PRI Sends	*IUT Sends*
RR---->	
P=1 N(R)=0	
	<----RR
	F=1 N(R)=0
	<----I N(S)=0
	<----I N(S)=1
	<----I N(S)=2
REJ---->	
P=1 N(R)=1	
	<----RR
	F=1
	<----I N(S)=1
	<----I N(S)=2
RR---->	
P=1	
	<----RR
	F=1

B.9.5 Handling UI/XID Frames All the tests in this section will execute the status check procedure described in Section B.7. Additionally, each subsection identifies the preamble that will be executed.

B.9.5.1 Basic Capabilities These tests are included for an IUT which supports the procedure for unimplemented frames that is specified in TR 41449. Since it is not required that an IUT support this procedure, these tests are optional. However, if this procedure has been implemented, these tests are mandatory.

Purpose of Test	
Test C_UIXID:	*Verify IUT can ...*
1	Ignore UI frames in S5
2	Ignore XID frames in S5
3	Ignore UI frames in S7.0
4	Ignore XID frames in S7.0
5	Ignore UI frames in S7.1
6	Ignore XID frames in S7.1
7	Ignore UI frames in S8.4
8	Ignore XID frames in S8.4
9	Ignore UI frames in S8.5
10	Ignore XID frames in S8.5

B.9.5.1.1 Awaiting Establishment - State S5

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
DM---->	
F=1	
	<----SABME (S5) †
	P=1

TEST C_UIXID:1

Purpose: Ignore UI frames in S5.

CONTEST-PRI Sends	*IUT Sends*
UI---->	}Ignore
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:2

Purpose: Ignore XID frames in S5.

CONTEST-PRI Sends	*IUT Sends*
XID---->	}Ignore
UA---->	} (S7.0)
F=1	
DISC---->	
P=1	
	<----UA
	F=1

† The (S#) denotes the state the IUT will be in after sending this frame

B.9.5.1.2 Multiple Frame Established (Normal) - State S7.0

TEST C_UIXID:3

Purpose: Ignore UI frames in S7.0.

CONTEST-PRI Sends	*IUT Sends*
UI---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:4

Purpose: Ignore XID frames in S7.0.

CONTEST-PRI Sends	*IUT Sends*
XID---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

B.9.5.1.3 Multiple Frame Established (Reject) - State S7.1

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
	<----REJ (S7.1) †

TEST C_UIXID:5

Purpose: Ignore UI frames in S7.1.

CONTEST-PRI Sends	*IUT Sends*
UI---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:6

Purpose: Ignore XID frames in S7.1.

CONTEST-PRI Sends	*IUT Sends*
XID---->	}Ignore
DISC---->	
P=1	
	<----UA (S4)
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

B.9.5.1.4 Timer Recovery (Normal, Peer Receiver Busy) - State S8.4

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
P=1	
	<----RR
	F=1
Do Nothing	
	T200 expires
	<----RR (S8.4) †
	P=1

TEST C_UIXID:7

Purpose: Ignore UI frames in S8.4.

CONTEST-PRI Sends	*IUT Sends*
UI---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:8

Purpose: Ignore XID frames in S8.4.

CONTEST-PRI Sends	*IUT Sends*
XID---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

B.9.5.1.5 Timer Recovery (Reject, Peer Receiver Busy) - State S8.5

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
I (with invalid N(S))---->	
	<----REJ (S7.1) †
RNR----> P=1	
	<----RR (S7.5) F=1
Do Nothing	
	T200 expires <----RR (S8.5) P=1

TEST C_UIXID:9

Purpose: Ignore UI frames in S8.5.

CONTEST-PRI Sends	*IUT Sends*
UI----> DISC----> P=1	} Ignore
	<----UA F=1

TEST C_UIXID:10

Purpose: Ignore XID frames in S8.5.

CONTEST-PRI Sends	*IUT Sends*
XID----> DISC----> P=1	} Ignore
	<----UA F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

B.9.5.2 TEI Assigned - State S4 These tests are included for an IUT which supports the procedure for unimplemented frames that is specified in TR 41449. Since it is not required that an IUT support this procedure, these tests are optional. However, if this procedure has been implemented and if the IUT can remain in TEI Assigned State long enough, these tests are mandatory.

Purpose of Test	
Test C_UIXID:	*Verify IUT can ...*
11	Ignore UI frames in S4
12	Ignore XID frames in S4

PREAMBLE

CONTEST-PRI Sends	*IUT sends*
DISC---->	
P=1	
	<----UA (S4) †
	F=1

TEST C_UIXID:11

Purpose: Ignore UI frames in S4.

CONTEST-PRI Sends	*IUT Sends*
UI---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_UIXID:12

Purpose: Ignore XID frames in S4.

CONTEST-PRI Sends	*IUT Sends*
XID---->	}Ignore
P=0	
DISC---->	
P=1	
	<----DM
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame.

B.9.5.3 Awaiting Release - State S6 These tests are included for an IUT which supports the procedure for unimplemented frames that is specified in TR 41449. Since it is not required that an IUT support this procedure, these tests are optional. However, if this procedure has been implemented and if the IUT can send a DISC frame on demand, these tests are mandatory.

Purpose of Test	
Test C_UIXID:	*Verify IUT can ...*
13	Ignore UI frames in S6
14	Ignore XID frames in S6

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
	<----DISC (S6) †
	P=1

TEST C_UIXID:13

Purpose: Ignore UI frames in S6.

CONTEST-PRI Sends	*IUT Sends*
UI---->	}Ignore
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

TEST C_UIXID:14

Purpose: Ignore XID frames in S6.

CONTEST-PRI Sends	*IUT Sends*
XID---->	}Ignore
UA---->	} (S4)
F=1	
DISC---->	
P=1	
	<----DM
	F=1

† The (S#) denotes the state the IUT will be in after sending this frame

B.9.5.4 Multiple Frame Established (Busy) - State S7.2 These tests are included for an IUT which supports the procedure for unimplemented frames that is specified in TR 41449. Since it is not required that an IUT support this procedure, these tests are optional. However, if this procedure has been implemented and if the IUT can send a RNR frame in Multiple Frame Established State, these tests are mandatory.

Purpose of Test	
Test C_UIXID:	*Verify IUT can ...*
15	Ignore UI frames in S7.2
16	Ignore XID frames in S7.2

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
	<----RNR (S7.2) †

TEST C_UIXID:15

Purpose: Ignore UI frames in S7.2.

CONTEST-PRI Sends	*IUT Sends*
UI---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:16

Purpose: Ignore XID frames in S7.2.

CONTEST-PRI Sends	*IUT Sends*
XID---->	} Ignore
DISC---->	
P=1	
	<----UA (S4)
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame

B.9.5.5 Timer Recovery (Peer and Own Receiver Busy) - State S8.6 These tests are included for an IUT which supports the procedure for unimplemented frames that is specified in TR 41449. Since it is not required that an IUT support this procedure, these tests are optional. However, if this procedure has been implemented and if the IUT can send a RNR frame in Timer Recovery State, these tests are mandatory.

Purpose of Test	
Test C_UIXID:	*Verify IUT can ...*
17	Ignore UI frames in S8.6
18	Ignore XID frames in S8.6

PREAMBLE

CONTEST-PRI Sends	*IUT Sends*
RNR---->	
	<----RNR (S7.6)
Do Nothing	T200 expires
	<----RNR (S8.6) †

TEST C_UIXID:17

Purpose: Ignore UI frames in S8.6.

CONTEST-PRI Sends	*IUT Sends*
UI---->	} Ignore
DISC---->	
P=1	
	<----UA
	F=1

TEST C_UIXID:18

Purpose: Ignore XID frames in S8.6.

CONTEST-PRI Sends	*IUT Sends*
XID---->	} Ignore
DISC---->	
P=1	
	<----UA (S4)
	F=1

† The (S#) denotes the new state the IUT will be in after sending this frame