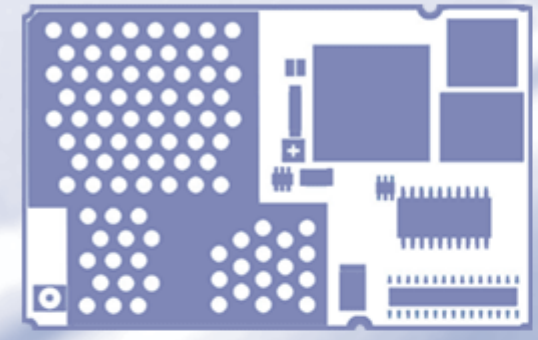
SIEMENS



Multiplexer User's Guide

(Windows XP/2000)

Siemens Cellular Engines

Version: 09
DocId: Mux_Guide_v09

User's Guide

Document Name:	**Multiplexer User's Guide**
Version:	**09**
Date:	**2006-9-27**
DocId:	**Mux_Guide_v09**
Status	**Confidential / Released**

General Notes

Product is deemed accepted by recipient and is provided without interface to recipient's products. The documentation and/or product are provided for testing, evaluation, integration and information purposes. The documentation and/or product are provided on an "as is" basis only and may contain deficiencies or inadequacies. The documentation and/or product are provided without warranty of any kind, express or implied. To the maximum extent permitted by applicable law, Siemens further disclaims all warranties, including without limitation any implied warranties of merchantability, completeness, fitness for a particular purpose and non-infringement of third-party rights. The entire risk arising out of the use or performance of the product and documentation remains with recipient. This product is not intended for use in life support appliances, devices or systems where a malfunction of the product can reasonably be expected to result in personal injury. Applications incorporating the described product must be designed to be in accordance with the technical specifications provided in these guidelines. Failure to comply with any of the required procedures can result in malfunctions or serious discrepancies in results. Furthermore, all safety instructions regarding the use of mobile technical systems, including GSM products, which also apply to cellular phones must be followed. Siemens or its suppliers shall, regardless of any legal theory upon which the claim is based, not be liable for any consequential, incidental, direct, indirect, punitive or other damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information or data, or other pecuniary loss) arising out the use of or inability to use the documentation and/or product, even if Siemens has been advised of the possibility of such damages. The foregoing limitations of liability shall not apply in case of mandatory liability, e.g. under the German Product Liability Act, in case of intent, gross negligence, injury of life, body or health, or breach of a condition which goes to the root of the contract. However, claims for damages arising from a breach of a condition, which goes to the root of the contract, shall be limited to the foreseeable damage, which is intrinsic to the contract, unless caused by intent or gross negligence or based on liability for injury of life, body or health. The above provision does not imply a change on the burden of proof to the detriment of the recipient. Subject to change without notice at any time. The interpretation of this general note shall be governed and construed according to German law without reference to any other substantive law.

Contents

0 Document History 7

1 Introduction 9
1.1 Supported Products 9
1.2 Related Documents 10
1.3 References 10
1.1 Terms and Abbreviations 11

2 Multiplexer Protocol - an Overview 12
2.1 Product Concept and Architecture 12
2.2 Virtual Channels and AT Commands 13

3 Integrating Multiplexer into the External Application 14
3.1 Characteristics 14
3.1.1 Basic Requirements 14
3.1.2 Restrictions 15
3.1.3 Dependencies between Multiplex Channels and Restrictions of Use 15
3.1.4 Functions without Channel Dependencies 16
3.1.5 Timing Conditions 16
3.1.6 Relationship between Multiplex Channels and Non-multiplexed Physical Interfaces 17
3.1.6.1 First Serial Interface ASC0 17
3.1.6.2 Operation of a Second Physical Serial Interface ASC1 (if Applicable) 17
3.1.6.3 Operation of the USB Interface (if Applicable) 17
3.2 Multiplexer Control and Signaling Lines 18
3.2.1 Flow Control 18
3.2.2 Escape Sequence 20
3.3 Power Saving 20
3.4 Bandwidth of Logical Channels 20

4 Structure of the Multiplexer Protocol 21
4.1 Introduction of the Multiplexer Protocol 21
4.2 Data Link Layer 21
4.2.1 Flag Sequence 22
4.2.2 Address Field 22
4.2.3 Control Field 23
4.2.4 Length Indicator 23
4.2.5 Information Field 24
4.2.6 Frame Checking Sequence Field (FCS) 24
4.3 State Diagrams 25
4.3.1 Start-up Procedure 29
4.3.2 DLC Establishment 29
4.3.3 Information Transfer 29
4.3.4 DLC Release 30
4.3.5 Close-down Procedure 30
4.3.6 Multiplexer Control Channel 30
4.3.7 Multiplexer Close Down (CLD) 31
4.3.8 Test Command (Test) 31
4.3.9 Modem Status Command (MSC) 31

4.3.10 Power Saving Control (PSC) 33
4.3.11 Non-supported Command Response (NSC) 34
4.4 Example: Establishing Logical Channels without Parameter Negotiation 35

5 Multiplexer Protocol Version Control 36
5.1 Introduction 36
5.2 Activating a Siemens Multiplexer Protocol Version 36
5.3 Overview of Siemens Multiplexer Protocol Versions 36
5.4 Implementing Version Control 38
5.4.1 Troubleshooting 38
5.4.2 „TestCommand" Message Coding 38
5.4.3 Example of „TestCommand" Message 39
5.5 Siemens Multiplexer Protocol Version 4 40
5.5.1 Activating Siemens Multiplexer Protocol Version 4 40
5.5.2 Usage of Frames in Siemens Multiplexer Protocol Version 4 40
5.5.3 Message Sequence Charts for Error Recovery Mode 42
5.5.4 State / Event Machine for Error Recovery Mode 50
5.5.5 Trace of a Standard Multiplex Protocol Start-up with Siemens Multiplexer Version 4 .. 52

Tables

Table 1: Comparison of multiplex channels 13
Table 2: Allocation of virtual channels to physical interfaces 17
Table 3: Address field 22
Table 4: Assignement of the DLCI 22
Table 5: Use of the command/response bit 22
Table 6: Coding of the control field 23
Table 7: Version differences for MSC 37
Table 8: IEI coding 38
Table 9: Coding of "TestCommand" (Example) 39
Table 10: Frame example for multiplexer protocol version 4 41
Table 11: Control field coding in multiplexer protocol version 4 41
Table 12: Typographical conventions used in the trace example 52

Figures

Figure 1: Multiplexer architecture 12
Figure 2: Logical flow control and RTS/CTS signaling behind the decoder 19
Figure 3: Data link layer 21
Figure 4: Relationship between the customer µC and the GSM engine µC 26
Figure 5: MPI-Startup, DLC establishment and information transfer 27
Figure 6: MP - DLC release and close down 28
Figure 7: DLC establishment 29
Figure 8: Information transfer 29
Figure 9: DLC release 30
Figure 10: Multiplexer control channel 30
Figure 11: Modem status command (MSC) 31
Figure 12: Establishing the multiplexer control channel and the logical channel 35
Figure 13: MSC as used in multiplexer protocol version 37
Figure 14: V.24 signals as used in multiplexer protocol version 3 37
Figure 15: V.24 signals as used in multiplexer protocol version 4 37
Figure 16: Data transmission in one direction 42
Figure 17: Data transmission in both directions 43
Figure 18: Data transmission to busy station 43
Figure 19: Data transmission to busy station with Receive Ready missing 44
Figure 20: Data transmission to busy station with Receive Not Ready missing 45
Figure 21: Data transmission to busy station with T1 timer expired 46
Figure 22: Data transmission with Receive Ready missing, scenario 1 47
Figure 23: Data transmission with Receive Ready missing, scenario 2 48
Figure 24: Sequence error 49
Figure 25: State / event machine (1) 50
Figure 26: State / event machine (2) 51
Figure 27: State / event machine (3) 51

0 Document History

This chapter reports modifications and improvements over previous versions of this document.

Preceding document: "Multiplexer User's Guide" Version 08

New document: "Multiplexer User's Guide" Version **09**

Chapter	What is new
Throughout document	Added further supported products. Added Siemens multiplexer protocol V4.
1.3	Updated list of standards.
3.1.1	Added note on interfaces used for Multiplexer mode. Added UIH_E framing.
3.1.2	Frame size N1 adjustable from Siemens multiplexer protocol V4 onwards.
3.2.1	Changed description of flow control on logical channels and on the physical interface. Added note on handling of FC bit from Siemens multiplexer protocol V4 onwards.
5.4.1	Corrected description of troubleshooting.
5.5	New chapter: Siemens Multiplexer Protocol Version 4.

Preceding document: "Multiplexer User's Guide" Version 07

New document: "Multiplexer User's Guide" Version 08

Chapter	What is new
Throughout document	Added AC75.
2.1	Revised chapter.
3.1.3	Added product specific remark on unframed data on closed virtual channels.
3.1.4	Removed Remark on URCs.
4.3.10	Added coding of PSC message for SLEEP mode AT+CFUN=9.

Preceding document: "Multiplexer User's Guide" Version 06

New document: "Multiplexer User's Guide" Version 0**7**

Chapter	What is new
1.1	Added further supported products.
3.1.1	RTS/CTS not relevant on USB. Remark on periodic output of parameters.
3.1.4	Modified remark on AT&W.
3.1.6, 3.1.6.1	New chapters Relationship between Multiplex Channels and Non-multiplexed Physical Interfaces and First Serial Interface ASC0.
3.1.6.3	New chapter: Operation of the USB Interface (if Applicable).
3.3	More detailed info on supported CFUN levels.
3.2.1	RTS/CTS not relevant on USB.
5.1	Added note on Multiplexer protocol version for MC75, TC63, TC65.

Preceding document: "Multiplexer User's Guide" Version 05

New document: "Multiplexer User's Guide" Version 06

Chapter	What is new
1.1	Added further supported products.
3.1.1	Added note about closing Multiplexer.
3.1.2	Added note about maximum frame size N1.
4.2.5	Second byte for frame size greater than 127 bytes is not supported.
4.3.5	Corrected description of Close-down procedure.
5	Corrected description of multiplexer version control.
5.1	Corrected example.

Preceding document: "Multiplexer User's Guide" Version 04

New document: "Multiplexer User's Guide" Version 05

Chapter	What is new
1.1	Added further supported products.
3.1.4	Modified remark on AT&W.
3.1.6.2	Added chapter Operation of a Second Physical Serial Interface ASC1 (if Applicable).

Preceding document: "Multiplexer User's Guide" Version 03

New document: "Multiplexer User's Guide" Version 04

Chapter	What is new
1.1	Added further supported products.
3 - 3.4	Restructured and revised all chapters.
3.1.2, 3.3, 4.3.10	To control SLEEP mode use PSC messages rather than entering AT+CFUN=<n>

1 Introduction

Siemens GSM engines support the basic option of the multiplexer defined in the GSM 07.10 Multiplexer Protocol. The multiplexer enables one physical serial interface to be partitioned into three virtual channels. This allows you to take advantage of three simultaneous sessions running on one interface.

Outside the GSM engine, on the application side, another multiplexer must be integrated in order to demultiplex the signal and distribute it on the three virtual channels. The external multiplexer needs to be implemented by the customer. To help system integrators save the time and expense of designing multiplexer applications, Siemens AG offers WinMUX2k, a ready-to-use multiplex driver for Windows 2000 and Windows XP.

This document describes how to use the multiplexer and then explains how to design an external multiplexer and integrate it into an application on top of a Siemens GSM engine.

Multiplexer protocol sources (WinMux2k), provided by Siemens AG, can be obtained on request from your local distributor. For more detailed information please refer to [4].

1.1 Supported Products

Products incorporating multiplexer protocol version 3:

- AC43
- AC45
- MC35i
- MC35i Terminal
- MC39i
- MC45
- MC46
- MC388
- MC5x
- TC35i
- TC35i Terminal
- TC45
- XT55
- XT56
- MC75
- TC63
- TC65
- TC65 Terminal
- Ready-to-use multiplex driver WinMux2k

Products incorporating multiplexer protocol version 4:

- AC65
- AC75
- XT65
- XT75
- Ready-to-use multiplex driver WinMux2k as of version 3.000

1.2 Related Documents

[1] Hardware Interface Description supplied with your GSM engine
[2] AT Command Set supplied with your GSM engine
[3] Release Notes supplied with your GSM engine
[4] Multiplexer Driver Developer's Guide for Windows 2000 and Windows XP
[5] Multiplexer Driver Installation Guide for Windows 2000 and Windows XP

For further documents regarding your GSM engine please refer to the latest Release Notes supplied with the module.

To visit the Siemens Website you can use the following link:

http://www.siemens.com/wm

1.3 References

[1] 3G TS 27.010, 3rd Generation Partnership Project; Technical Specification Group Terminals; Terminal Equipment to Mobile Station (TE-MS) multiplexer protocol
[2] Digital Cellular Telecommunications Systems (Phase 2+); Terminal Equipment to Mobile Station (TE-MS) "Multiplexer Protocol"; ETSI TS 101 369 V7.1.0 (1999-11), GSM 07.10

1.1 Terms and Abbreviations

Abbreviation	Description
CSD	Circuit Switched Data
CTS	Clear to Send
DCD	Data Carrier Detect
DLCI	Data Link Control Identifier
DSB	Developer Support Box
DSR	Data Set Ready
DTR	Data Terminal Ready
FC	Flow Control
FFC	Flat Flex Cable
GPRS	General Packet Radio Service
GSM	Global System of Mobile Communication
IEI	Information Element Identifier
IP	Internet Protocol
MO	Mobile originated
MP	Multiplexer Protocol
MS	Mobile Station
MSDN	Microsoft Developer Network
MT	Mobile terminated
MUX	Multiplexer
OS	Operating System
PC	Personal Computer
PSC	Power saving control
RTS	Request to Send
TE	Terminal Equipment
UART	Universal Asynchronous Receiver Transmitter

2 Multiplexer Protocol - an Overview

2.1 Product Concept and Architecture

The multiplexer mode enables one serial interface to transfer data to or from three different host applications. This is achieved by providing three virtual channels running the multiplexer protocol.

For example, multiplexer mode allows the user to make a call (voice, fax, data or GPRS) on one channel and, at the same time, to control the module and even send or receive short messages on other channels, without disturbing the data flow on either channel. This is of great advantage when several independent electronic devices or interfaces are used.

To access the three virtual interfaces, both the GSM engine and the host application must contain Mux components which communicate over the multiplexer protocol.

In multiplexer mode, AT commands and data are encapsulated into packets. Each packet has a channel identification and may vary in length.

Note: All statements regarding GPRS are valid only for Siemens wireless products capable of GPRS.

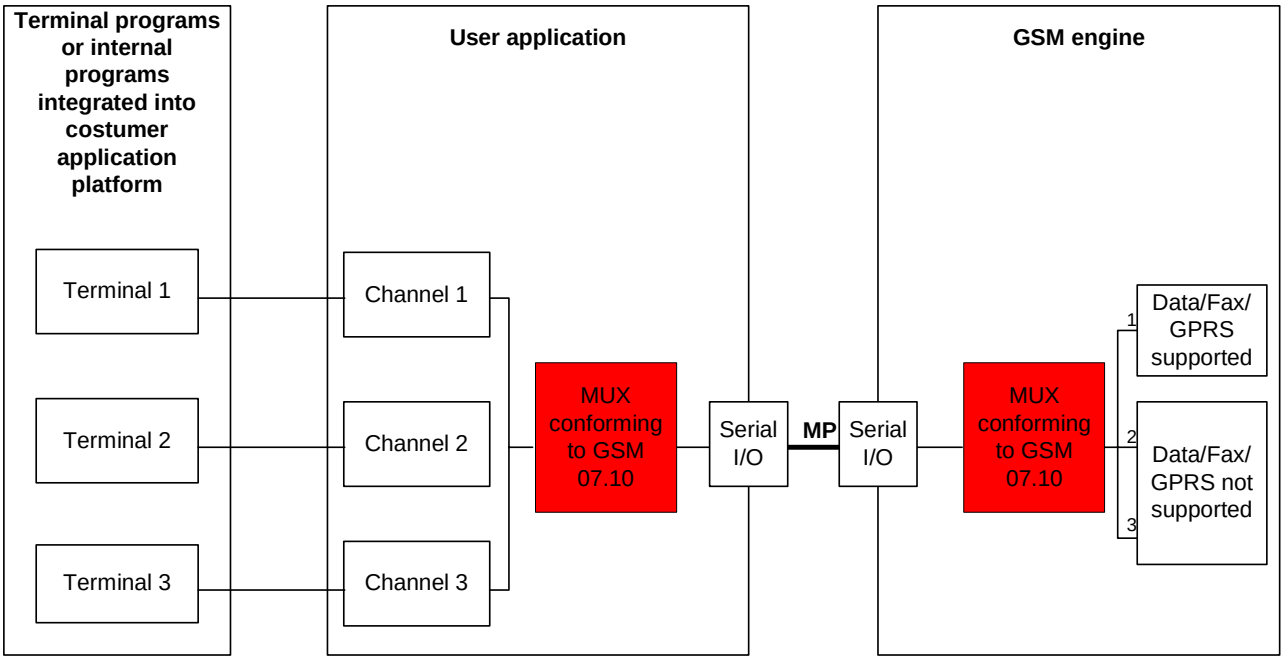


Figure 1: Multiplexer architecture

2.2 Virtual Channels and AT Commands

Please note that a cellular engine designed for multiplex operation does not include three different devices. Only one single air interface (RF part) is available.

As mentioned before the multiplexer enables one serial interface to run three sessions simultaneously. All incoming or outgoing calls are connected to the device.

Channel 1 supports the full range of functions, which is available without multiplexer tool.

Channel 2 and 3 are connected to a different AT interpreter and support a subset of the functional range of channel 1, for more details refer to Table 1.

Table 1: Comparison of multiplex channels

	Voice calls incoming outgoing	Data / fax calls incoming outgoing	SMS incoming outgoing	GPRS connection	Phonebook management	AT commands
Channel 1	•	•	•	•[1]	•	•
Channel 2, 3	•	-	•	•[1]	•	•[2]

1. Only two channels can be used in parallel to transmit GPRS data
2. Except for AT commands related to data and fax calls

- • indicates that the functionality is available on the channel
- - indicates that the functionality is not available on the channel

Examples

- While a data call is in progress on channel 1, you can send a short message on channel 2 and edit the phonebook on channel 3.
- The battery capacity can be queried from channel 2 or 3 while a voice, fax or data call is made on channel 1.

Note

Due to the technical requirements of multiplexer mode, data and fax calls can only be set up on logical channel 1 while GPRS connections can be established on every channel. Several AT commands have a different behavior on channels 2 and 3. Additional information regarding restrictions and interferences between the channels can be found in Section 3.1 and in [2].

3 Integrating Multiplexer into the External Application

When designing a multiplexer application, you can create your own sources or take advantage of the sources delivered upon request by Siemens. The Siemens sources are packed in a *.zip file which includes a driver for Windows 2000 and Windows XP. See [4] for a detailed description.

3.1 Characteristics

After establishing the multiplexer mode according to the multiplexer protocol, three logical channels are available. Please keep the following restrictions and requirements in mind:

3.1.1 Basic Requirements

- If your product provides several physical AT command interfaces please note that only the first serial interface ASC0 and the USB interface support multiplexer mode. Other interfaces, such as the second serial interface ASC1 of some products, do not support the multiplexer protocol. At a time, multiplexer mode can be started only on one physical interface. Bear in mind that all other AT command interfaces are disabled when multiplexer mode is active. See also Section 3.1.6 for further detail.
- The GSM engine supports Basic mode and UIH framing according to the GSM 07.10 specification. With the Siemens multiplexer protocol version 4, error recovery mode and extended UIH_E framing on the data channels have been introduced. UIH_E is a proprietary Siemens specific frame type, for details see Section 5.5.
- Character framing must be configured for 8 data bits, no parity and 1 stop bit (8N1). Please note that if your product includes AT commands for changing the character framing make sure that 8N1 is enabled before you activate the multiplexer.
- RTS/CTS hardware flow control (AT\Q3) is recommended for use with multiplexer mode. If used, it needs to be set before multiplexer mode is entered. On the USB interface, RTS/CTS hardware flow control is not relevant.
- Several customer software applications may be able to change the selected settings. These settings will be stored in the non-volatile memory and used whenever the module is powered up again. In this case the multiplexer fails to start. To avoid this, it is recommended to re-synchronize all settings before using the multiplexer mode again.
- Before closing the multiplexer make sure that there is no ongoing activity on one of the channels. For example, check that voice, CSD or GPRS connections have ended and wait until all pending AT command responses are received. The periodic output of parameters (such as cell information delivered by AT^MONI) must be terminated before closing the multiplexer.

3.1.2 Restrictions

If the GSM engine is operated in multiplexer mode, the following restrictions apply:

- MO and MT circuit-switched data and fax calls can only be set up on channel 1.
- It is not recommended to use AT+CFUN=<n> for selecting one of the SLEEP modes. For products supporting multiplexer protocol version 3 or 4, the best approach to properly control SLEEP mode in this case is to issue the PSC messages described in Section 4.3.10.
- During heavy traffic, the performance of the other multiplexer interfaces might be impaired. Some AT commands may behave differently than normal.

The multiplexer cannot be started if one of the following features is activated, nor can these features be used when multiplexer is active:

- Multiplex mode cannot be started while autobauding (AT+IPR=0) is enabled.
- XON/OFF flow control is not supported in multiplexer mode.
- "The multiplexer is not available in charge-only mode and in alarm mode (support of both modes is product specific).

The maximum frame size N1 (defined in GSM 07.10) is fixed to 98 bytes and cannot be changed if the product supports multiplexer protocol versions 1 - 3. With multiplexer protocol version 4, the frame size is adjustable as described in Section 5.5.2. The maximum frame size is the same for sending and receiving. For general explanations regarding frames see also Chapter 4 in this manual and GSM 07.10.

3.1.3 Dependencies between Multiplex Channels and Restrictions of Use

When using the following functions, be aware of possible dependencies between the different channels. One way of avoiding problems may be to dedicate certain commands/features to one of the channels or to assure that the application avoids conflicts.

- Call control: A voice call can be initiated, answered or ended on each channel. Data and fax calls shall be made on channel 1 only. See AT commands like ATD, ATA or ATH.

  ATH terminates each voice, circuit switched data or fax call regardless on which logical channel ATH was executed. In the case of GPRS, ATH is effective only on the channel where executed. For details see [2].

  Example: An ongoing fax call has been established on channel 1. When answering an incoming voice call on channel 2 or 3 and terminating it, the held fax call will be ended as well.
- Phonebook access: If you wish to write the same phonebook entry on two different channels at the same time, please note that the last entry will be stored permanently. All other data will be deleted.
- SMS read, write and delete.
- Time settings: Though the AT commands AT+CALA and AT+CCLK can be used on either channel, the same time setting applies to all three channels. It is only the alarm message <text> which may be specific to each channel. The URC "+CALA" will be issued only on the channel where the last alarm setting was made. For details see [2].
- Device locks set with AT+CLCK.
- SIM card access.
- RF settings.
- Note specific to product families AC4x, MC3x, TC35ix, MC5x, XT5x:

  Open at least channel 1. Otherwise, if only channel 2 or 3 is opened, data associated to the closed virtual channel(s) may be output as unframed data.

  Example: Assume that channel 1 and 2 are closed, a voice call is made on channel 3 and the remote subscriber hangs up. The "NO CARRIER" result code which normally appears on all 3 channels cannot be sent to channel 1 and 2, resulting in unframed data that cannot be allocated to a channel. Yet, there is no loss of data because the unframed characters are only additional data sent between the framed data of the opened channel.

3.1.4 Functions without Channel Dependencies

The following functions or events may be ongoing independently on different channels:

- Device information can be queried separately on each channel.
- Signal quality and cell information can be retrieved separately on each channel.
- Further commands that can be used separately on one channel without effect on the remaining channels: ATZ, AT&F, AT&V, AT+CEER, AT+CMEE.
- User profile: AT&W stores all global settings and the current settings of the interface on which the command was executed. See further details in Section 3.1.6.2 and Section 3.1.6.3.

3.1.5 Timing Conditions

Switching on the multiplexer with AT+CMUX=0 causes a 5s timer to start. If the multiplexer control channel is not established within this time, the module returns to "normal AT command mode" without multiplexer. This prevents the module from being blocked if, for example, AT+CMUX=0 is sent from an application that does not support the multiplexer protocol.

Fax is based on a protocol, which needs to respect timings between the application and the module as well as between the module and the selected terminal equipment (TE). Hence, heavy parallel traffic load in the module can lead to mistiming. This may result in malfunction in both directions. Please consider the following recommendations:

Using the multiplexer it is not possible to define bandwidth and delay time per channel. Therefore, the customer application should take care that the channels 2 and 3 are not heavily loaded when faxing on channel 1.

Example 1: Checking the field strength every 2 seconds does not harm, sending an SMS every 10 seconds may lead to problems.

Example 2: Reading a complete phone book, may cause problems if a fax transmission is ongoing at the same time.

When switching on the module after a firmware update we recommend to wait 5 seconds before entering the first AT command.

3.1.6 Relationship between Multiplex Channels and Non-multiplexed Physical Interfaces

Table 2 summarizes the allocation of non-volatile and user profile settings to the various multiplex channels and non-multiplexed physical interfaces. This allocation scheme shows where stored settings take effect when switching from multiplex to non-multiplex mode and vice versa. See chapters below for further detail.

Table 2: Allocation of virtual channels to physical interfaces

Physical interface	ASC0	ASC1 (if available)	USB (if available)
Corresponding multiplex channel	Multiplex channel 1	Multiplex channel 2	Multiplex channel 3

3.1.6.1 First Serial Interface ASC0

ASC0 and the multiplex channel 1 are using the same parameters, and thus, the same user defined profile (if any). As a result, non-volatile settings and a user profile stored on multiplex channel 1 will take effect on ASC0 after closing the multiplexer and switching to the physical interface ASC0. Likewise, non-volatile settings and a user profile stored on ASC0 will be loaded on multiplex channel 1. See also note on AT&W in Section 3.1.4.

3.1.6.2 Operation of a Second Physical Serial Interface ASC1 (if Applicable)

This chapter applies only to Siemens GSM modules equipped with a second physical serial interface (referred to as ASC1).

ASC1 is disabled when the multiplexer is enabled on the first serial interface ASC0. Yet, both ASC1 and the multiplex channel 2 are using the same parameters, and thus, the same user defined profile (if any). As a result, non-volatile settings and a user profile stored on multiplex channel 2 will take effect on ASC1 after closing the multiplexer and starting up ASC1. Likewise, non-volatile settings and a user profile stored on ASC1 will be loaded on multiplex channel 2. See also note on AT&W in Section 3.1.4.

This may be a problem when ASC1 is not connected, but flow control (for example AT\Q1 or AT\Q3) is stored to the user profile on the multiplex channel 2. In this case, flow control takes effect on ASC1, when the multiplexer is switched off. If then for example a large amount of URCs is generated, their transmission might be stopped due to the flow control. To avoid this problem we recommend that you do not activate flow control on multiplex channel 2 when setting up a user profile with AT&W.

3.1.6.3 Operation of the USB Interface (if Applicable)

This chapter applies only to Siemens GSM modules equipped with a USB interface, for details refer to [1].

ASC0 and ASC1 are disabled when the multiplexer is enabled on the USB interface.

The USB interface and the multiplex channel 3 are using the same parameters, and thus, the same user defined profile (if any). As a result, non-volatile settings and a user profile stored on multiplex channel 3 will take effect on the USB interface after closing the multiplexer and starting up USB. Likewise, non-volatile settings and a user profile stored on the USB interface will be loaded on multiplex channel 3. See also note on AT&W in Section 3.1.4.

3.2 Multiplexer Control and Signaling Lines

The following chapter covers all information you need to develop and set up a virtual driver. Differences and restrictions in comparison to the unframed module are pointed out.

3.2.1 Flow Control

Logical flow control

The internal logical flow control (FC bit in MSC message, see Section 4.3.9) is responsible for the flow control from the host application to the module. For example, if a data call is initiated and the host application transmits data to the module on this logical channel, the module will stop the data transmission from time to time. This happens because the module operates with a bandwidth of 9600 bps on air, but the host application uses a larger width. In this case the module sends an MSC message with the FC bit set. After all data stored in the internal buffer have been sent, the module will send a second MSC message with the FC bit reset. So, logical flow control works like RTS/CTS hardware flow control, but with the FC bit set on every channel. The RTS/CTS lines are not used for flow control because the traffic on the logical channels may cause a temporary loss of bandwidth on another channel. This behavior has no impact on the handshake V.24 lines.

With the proprietary Siemens multiplexer protocol version 4, the handling of the FC bit has been changed. The FC bit is still transmitted within the MSC message, but ignored. For details see Section 5.5.

RTS/CTS on the physical interface

On the physical interface, hardware flow control (AT\Q3) is recommended for use with the multiplexer. For power saving it is indispensable. The setting AT\Q3 needs to be made before switching on the multiplexer. An exception is the USB interface where RTS/CTS flow control is of no use.

The host application decodes and encodes the data. To prevent loss of data, the application must be able process the information about internal flow control (MSC) regulated by the module. Flow control information is transmitted within the data flow and contains messages whether or not the channel is allowed to send. See Section 4.3.9 for MSC.

As of Siemens multiplexer protocol version 2, the host application shall permanently activate RTS on all used logical channels. This is because no control data could be sent if RTS was deactivated. If flow control is needed, it is recommended either to set flow control on the logical channels by using the FC bit as explained above for multiplexer version 2 or 3 or, if available, to use the Siemens multiplexer version 4 as explained in Section 5.5.

RTS/CTS on the logical channels

The customer application needs to regulate the data flow according to the logical flow control. The implementation of the WinMux2k is a good example. It maps the 3 decoded channels to 3 serial interfaces as well as the logical flow control information (FC bit in MSC message) directly on the RTS/CTS-control lines.

In this case CTS superposes the STOP information (data sending disabled) sent by the module to control the data transmission from the customer application to the module. If RTS is reset, a STOP is transmitted to the module to control the data transmission from the module to the customer application. Figure 2 illustrates the data flow.

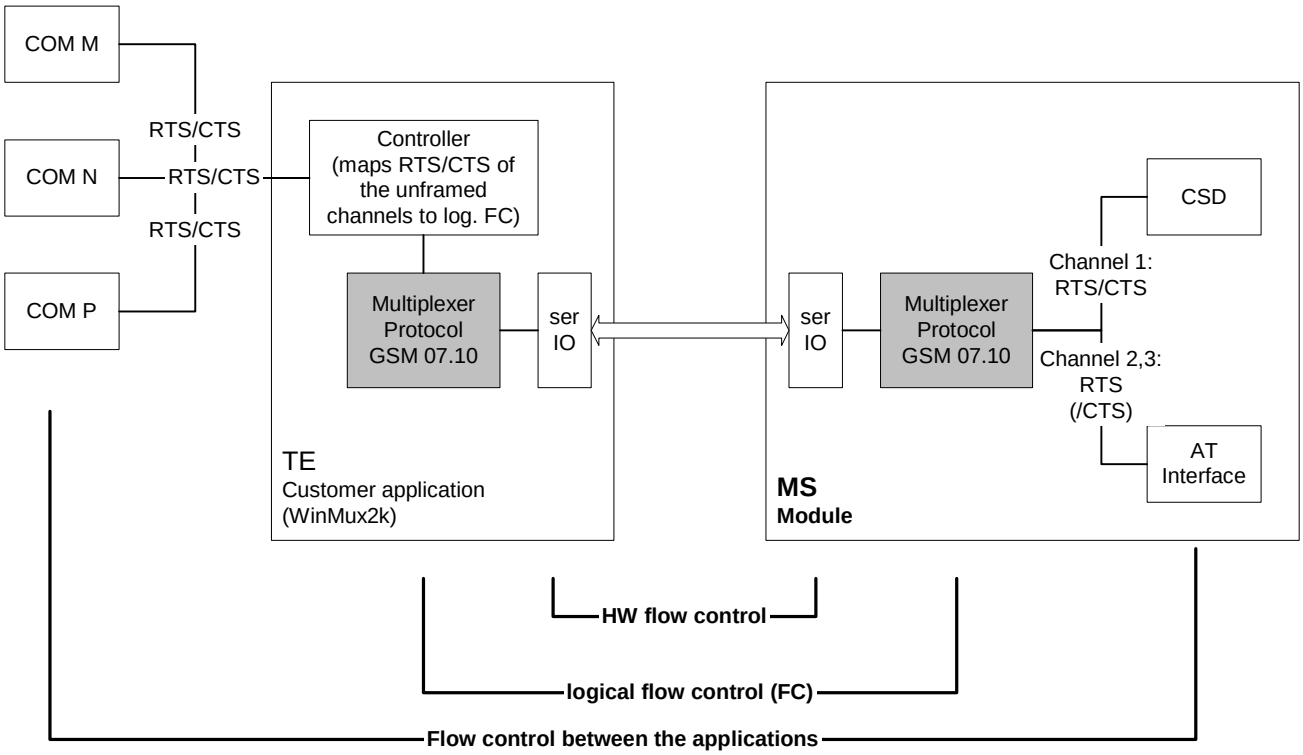


Figure 2: Logical flow control and RTS/CTS signaling behind the decoder

3.2.2 Escape Sequence

When the multiplexer protocol is active only coded data is transmitted over the UART. The coding includes a header and a checksum. Therefore, the direct parsing of this sequence is not possible. An escape might be undetected because the decoded time relations may be disturbed.

The following transmission path for the ESC signal has been implemented:

- DTR is transported within the logical channel. To terminate a call, the normal way of using DTR is available. Keep in mind that the multiplexer cannot transport this signal in real time. Please use a certain gap time between signaling with DTR.
- It is possible to detect "+++" on the customer multiplex application and transport this information via the MSC signal to the module (see Section 4.3.9).
- As an alternative, ATH may be sent on one of the other channels, for more detailed information please refer to [2].

3.3 Power Saving

SLEEP mode reduces the functionality of the module to a minimum and, thus, minimizes the current consumption to the lowest level. SLEEP mode can be set with the AT+CFUN command which provides the choice of the functionality levels <fun>=0, 1, 5, 6, 7, 8 or 9. For further details on power saving and the functionality levels supported by your Siemens product see [1] and [2].

If the module is in multiplexer mode, it is not recommended to activate SLEEP mode with AT+CFUN=<n>. For products supporting Multiplexer Protocol version 3, the best approach to properly control SLEEP mode in this case is to issue the PSC messages described in Section 4.3.10.

3.4 Bandwidth of Logical Channels

Please take into account that a data transmission, e.g. on channel 1, causes a transmission delay on the remaining channels (see Section 3.1). The multiplexer mode according to the GSM 07.10 multiplexer protocol encapsulates data and AT commands into packets which may vary in length. Therefore a header including protocol information located at the beginning of the protocol data unit has to be transmitted. To summarize, if the module is set to 115200 bps and an incoming GPRS call requires 5 kByte per second, the two other channels have to operate within the range of the remaining 5 kByte per second.

If three large data transmissions are running simultaneously, the available bandwidth will be shared equally among all channels. In such a case if channel 2 and 3 were used for data transmissions, e.g. editing the phonebook, both channels would need to share a bandwidth of approximately 3 kByte per second.

4 Structure of the Multiplexer Protocol

4.1 Introduction of the Multiplexer Protocol

The supported multiplexer protocol conforms to the GSM 07.10 Multiplexer Protocol. All products incorporating the multiplexer protocol version 1 - 3 have the non-error recovery mode implemented with the basic option. Products based on multiplexer protocol version 4 are capable of the error recovery mode defined in GSM 07.10 (see Section 5.5).

The frames have a start and a stop byte. A checksum is calculated to protect the transferred data. Frame repetition is not enabled.

Data and fax calls are transferred in the logical channel DLCI = 1 (DLCI: Data Link Connection Identifier). The remaining DLCIs are in AT command mode; two GPRS connections can be established simultaneously on every channel. The multiplexer protocol must be started and the logical channels opened in compliance with specified procedures.

This chapter also discusses the following issues:

- Opening logical channels without parameter negotiation
- Opening logical channels with parameter negotiation
- Closing of logical channels

4.2 Data Link Layer

The following chapters show the detailed structure of a data link frame.

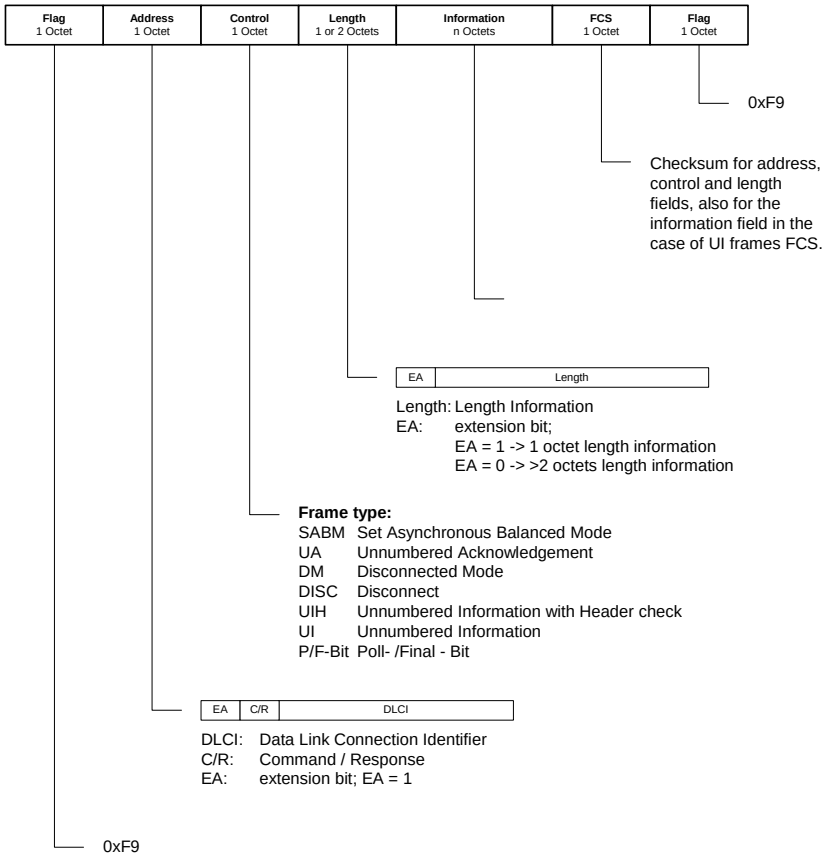


Figure 3: Data link layer

4.2.1 Flag Sequence

A flag sequence is a specific bit pattern (usually 11111001; hexadecimal: 0xF9) used to mark the beginning and the end of a frame of data.

Each frame begins and ends with a flag sequence. Only one flag sequence occurs between any two frames. If two successive flag sequences do occur, the frame is regarded as being empty and is discarded.

The flag sequence is used for the synchronization of frames.

4.2.2 Address Field

Data link connection identifier is a frame relay term defining a 10-bit field of the address field. The DLCI identifies the data link and its service parameters, including the frame size.

The values for the Data Link Connection Identifier (DLCI) are dynamically defined apart from DLCI = 0.

Table 3: Address field

EA	C/R	DLCI

DLCI: Data Link Connection Identifier
C/R: Command / Response
EA: extension bit; EA = 1

Table 4: Assignement of the DLCI

	DLCI number (decimal)	Priority
Multiplexer control channel (see Section 4.3.6)	0	0 highest priority
AT commands, data, fax, GPRS	1	7
AT commands, GPRS	2,3	7

The command/response bit identifies the frame as a command or response. A command contains the address of the data link connection to which the command is sent. A response contains the address of the data link connection sending the frame.

Table 5: Use of the command/response bit

Command/Response	Direction	C/R
Command (SABM, DISC)	Customer µC → GSM engine	1
	GSM engine → Customer µC	0
Response (UA, DM)	Customer µC → GSM engine	0
	GSM engine → Customer µC	1

Every command expects a response. No provision is made to repeat the command if no response is received.

4.2.3 Control Field

The control field contains control information to define the frame.

Table 6: Coding of the control field

Frame Type	1	2	3	4	5	6	7	8
SABM (set asynchronous balanced mode)	1	1	1	1	P/F	1	0	0
UA (unnumbered acknowledgement)	1	1	0	0	P/F	1	1	0
DM (disconnected mode)	1	1	1	1	P/F	0	0	0
DISC (disconnect)	1	1	0	0	P/F	0	1	0
UIH (unnumbered information with header check)	1	1	1	1	P/F	1	1	1

P/F: Poll/Final bit

Commands: P = 1, Responses: F = 1

For each DLCI, only one frame with P = 1 may ever be expected.

4.2.4 Length Indicator

The length indicator specifies the length of the following information field. As the maximum frame size N1 is 98 bytes and cannot be changed the E/A bit is always 1. The setting E/A = 0 defined in GSM 07.10 for a frame size greater than 127 bytes is not supported. See also Section 3.1.2 for details on the maximum frame size.

1st octet:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
E/A	L1	L2	L3	L4	L5	L6	L7

2nd octet (not supported by Siemens wireless modules):

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
L8	L9	L10	L11	L12	L13	L14	L15

E/A = 1: only one octet for the Length Indicator

E/A = 0: two octets for the Length Indicator

4.2.5 Information Field

The information field contains the data and has an octet structure. The field only exists for UIH frames (unnumbered information with header check).

To transfer information fields, the P/F bit is set to 0; a response is not necessarily expected.

4.2.6 Frame Checking Sequence Field (FCS)

The Frame Checking Sequence (FCS) is computed with the address, control and length fields. It is a field added to the end of a frame that contains transmission error-checking information. This field contains a value which is calculated by the module. The host application performs the same calculation. If the calculation of the host application does not match the result sent by the module, the packet is considered corrupt and discarded. An FCS calculation is made for each packet.

4.3 State Diagrams

The multiplexer protocol is based on two state machines (see Figure 4). One state machine initiates the setup of the logical channels, the other one responds to the requests.

The module can only respond to requests. A higher level for controlling the state machines is not implemented.

The procedure for setting up the two state machines – the one for the customer µC and the one for the GSM engine – is shown in Figure 5 and Figure 6.

Executing the AT command AT+CMUX=0 starts the switchover from AT command mode to the multiplexer protocol and parameterizes the multiplexer control channel DLCI = 0. Both state machines are entering the DISCONNECTED state and immediately have the option of setting up the multiplexer control channel DLCI = 0 and other logical channels.

The logical channels are then set up (DLC establishment). If the DLC has been established successfully the state machine for that particular channel changes to CONNECTED. If the request is unsuccessful the logical channel cannot be established and the state machine remains in DISCONNECTED on this particular channel.

Information can be transferred over all channels in CONNECTED. Control commands can be transferred in the multiplexer control channel DLCI = 0; the other channels transfer data.

The parameters for all logical channels DLCI = 1...4 in DISCONNECTED can be set for the requested logical channels by parameter negotiation.

Disconnecting individual channels (DLC release) causes the state machine for those channels to revert to DISCONNECTED. Release of the multiplexer control channel DLCI = 0 corresponds to a CLOSE DOWN. The CLOSE DOWN command terminates the Multiplex mode, and the GSM engine reverts to AT command mode.

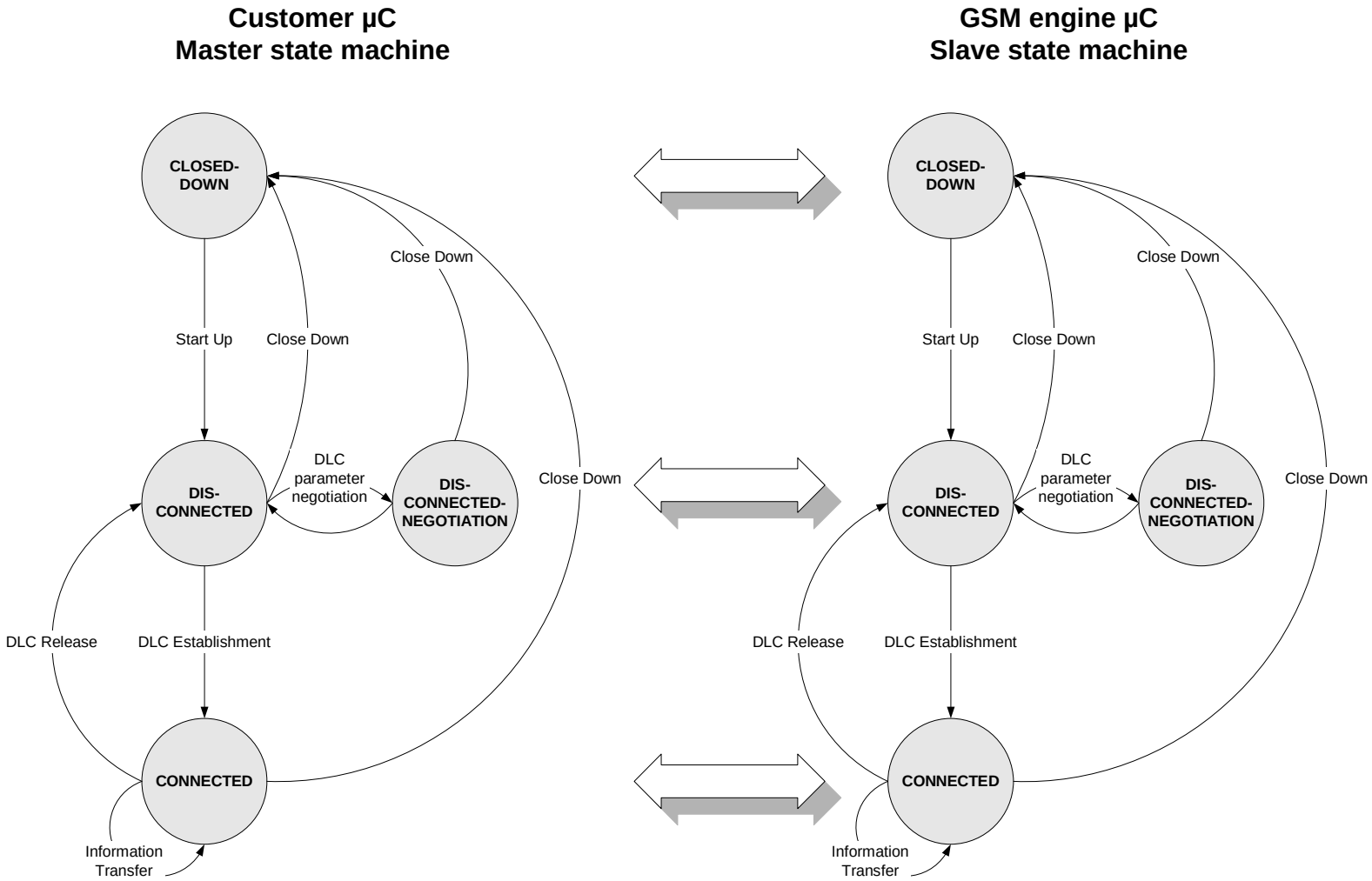


Figure 4: Relationship between the customer µC and the GSM engine µC

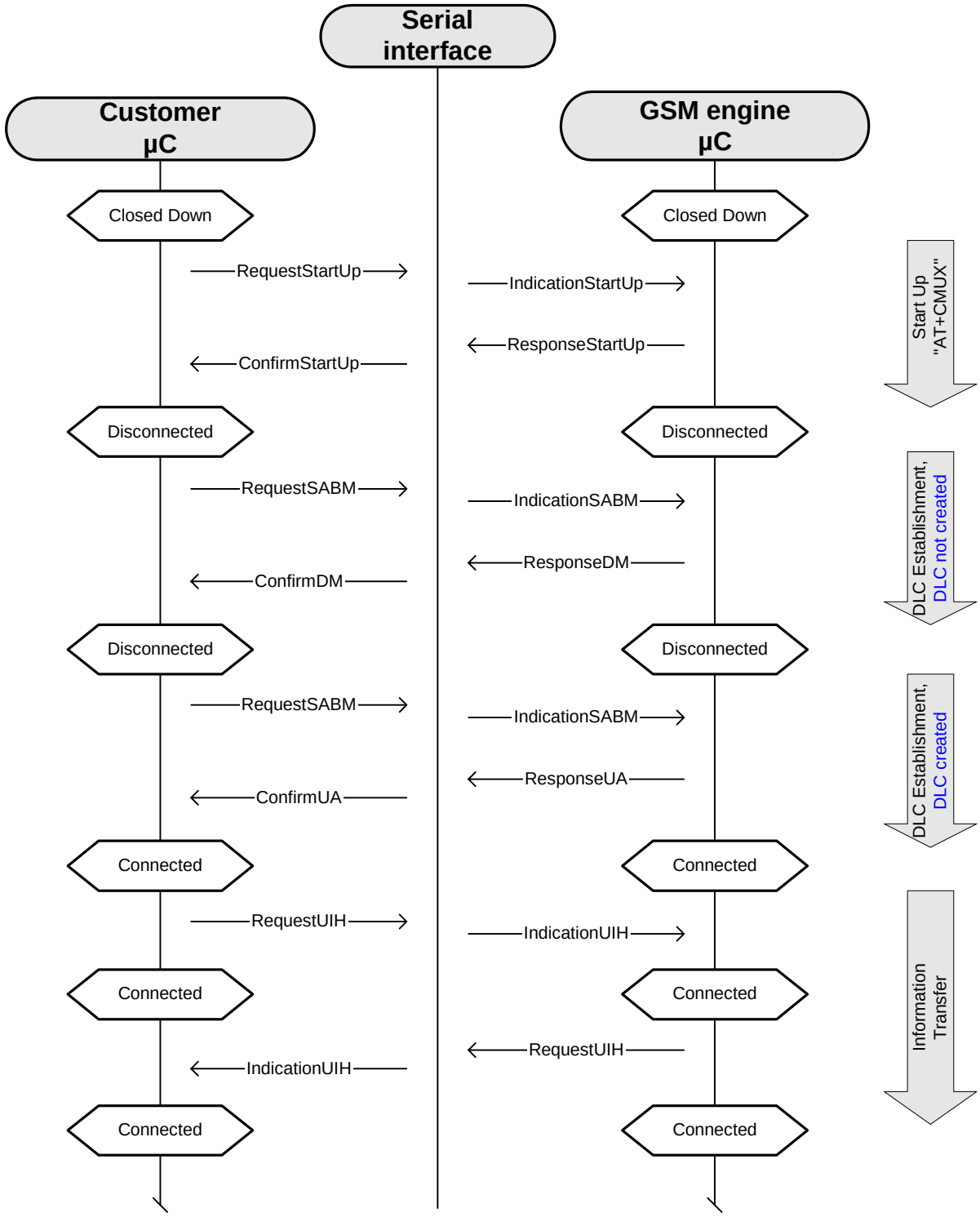


Figure 5: MPI-Startup, DLC establishment and information transfer

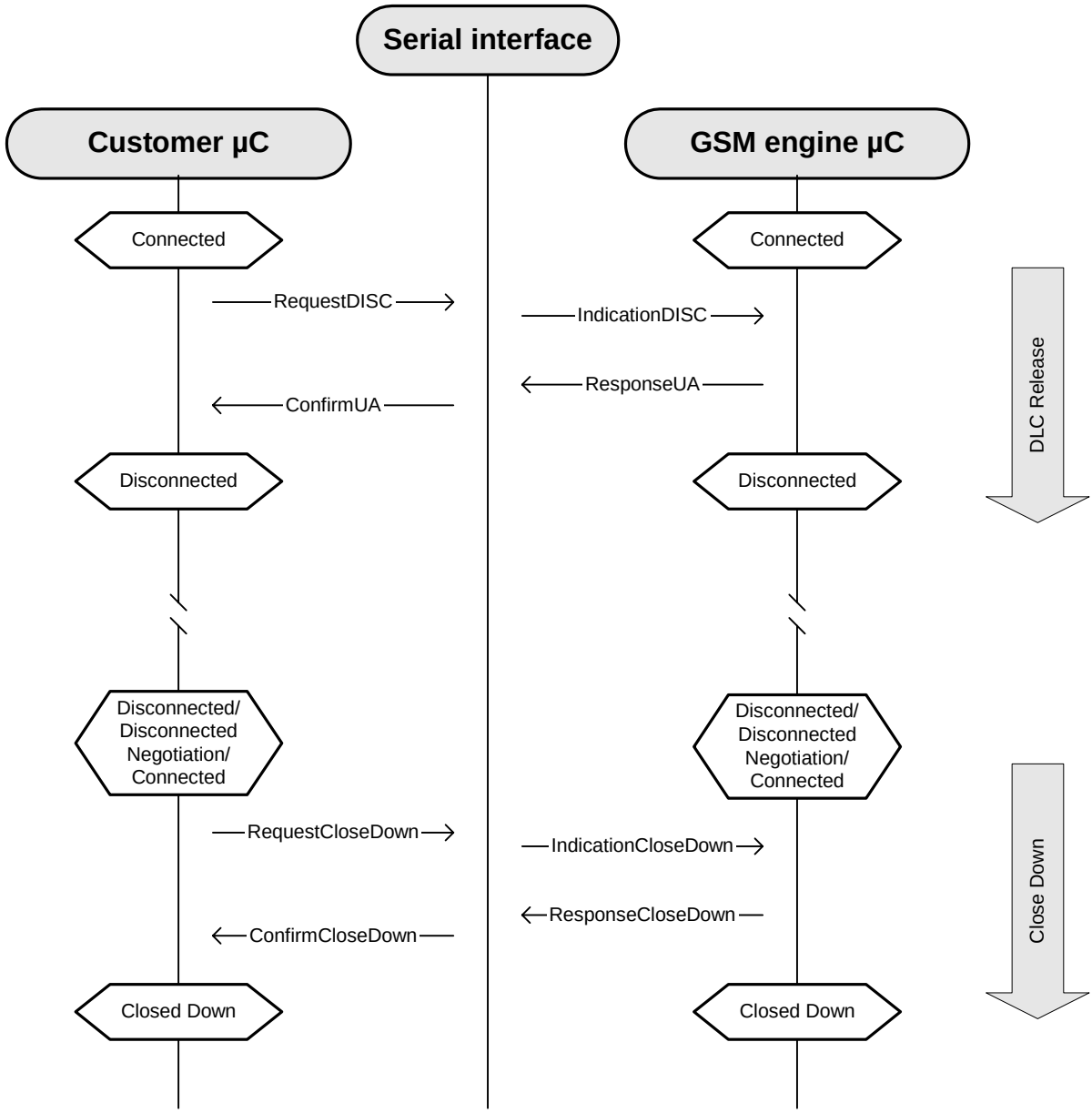


Figure 6: MP - DLC release and close down

4.3.1 Start-up Procedure

The only approach to activate the multiplexer protocol is entering the appropriate AT command AT+CMUX=0. This enables the multiplexer control channel. The next step is to set up the multiplexer control channel as described in Section 4.3.2.

4.3.2 DLC Establishment

The multiplexer control channel must be set up as the first channel followed by all other DLCIs. To do so, a SABM frame (see Section 4.2.3) must be sent to the GSM engine.

The module responds either with a UA frame if the DLCI was set up, or with a DM frame if the DLCI was not set up.

No provision is made for repeating the request if a response is not received.

The state machine requesting the multiplexer control channel DLCI = 0 is the "initiating station", while the other is called the "responding station".

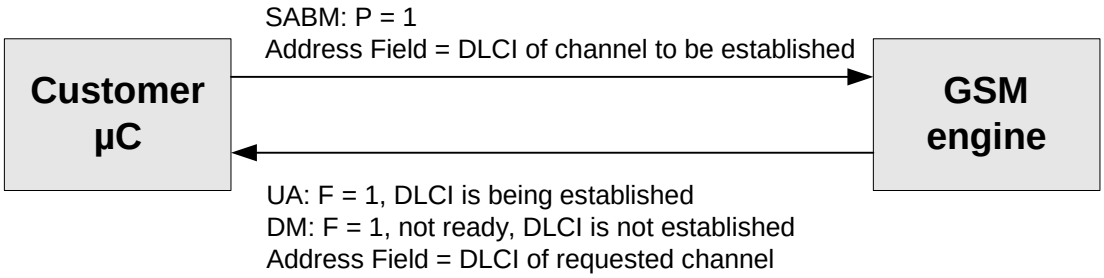


Figure 7: DLC establishment

4.3.3 Information Transfer

A response is not essential for every command – for example, an unsolicited result code does not require a response.

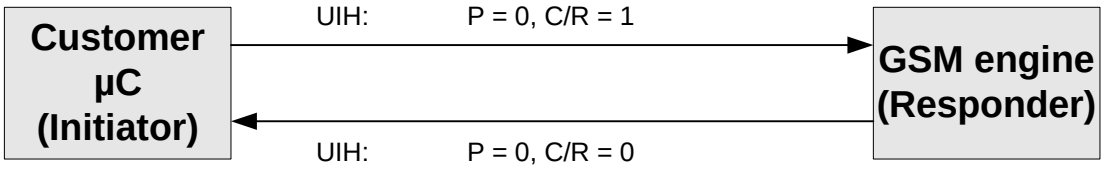


Figure 8: Information transfer

4.3.4 DLC Release

No provision is made to repeat the request if no response is received.

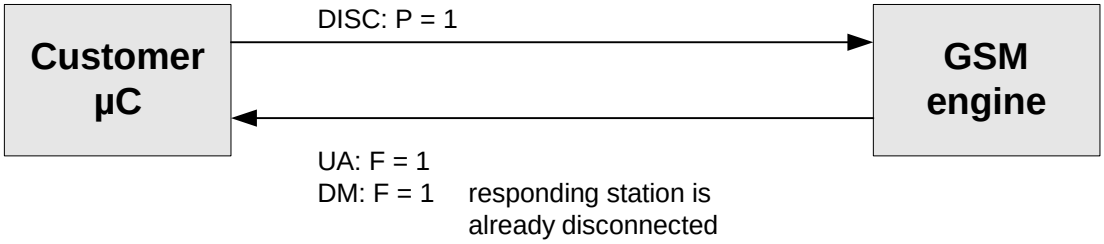


Figure 9: DLC release

4.3.5 Close-down Procedure

To close down the multiplexer follow these two steps:

- First, disconnect all DLCIs by sending the DLCI Release command within the multiplexer control channel frame (as described in Section 4.3.6).
- Finally, close down the multiplexer control channel (DLCI = 0) by sending the multiplexer close down command CLD (see Section 4.3.7). After this, both the "initiating station" and the "responding station" revert to AT command mode.

Before closing the multiplexer make sure that there is no ongoing activity on one of the channels. For example, check that voice, CSD or GPRS connections have ended and wait until all pending AT command responses are received.

4.3.6 Multiplexer Control Channel

DLCI = 0

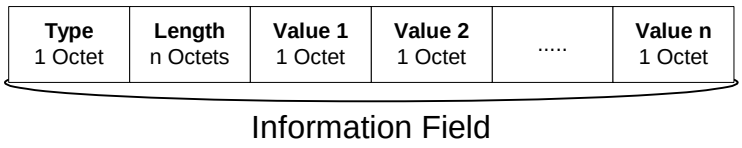


Figure 10: Multiplexer control channel

The commands are sent as information in the multiplexer control channel frame.

Type field:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
EA	C/R	T1	T2	T3	T4	T5	T6

EA bit: Extension bit.

In the last octet of the sequence the EA bit = 1, otherwise = 0.

If there is only on octet, EA bit = 1 is set.

C/R bit: Indicates whether the sequence is a command or a response.

T-bits: Coding of the command type.

Length field:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
EA	L1	L2	L3	L4	L5	L6	L7

EA bit: Extension bit.

In the last octet of the sequence the EA bit = 1, otherwise = 0.

If there is only one octet, EA bit = 1 is set.

L-bits: Number of value octets; the following L1 is the LSB, L7 the MSB.

Multiple commands can be sent in a single frame only.

4.3.7 Multiplexer Close Down (CLD)

Type field:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	0	0	1	1

Length byte = 0, no value octet

4.3.8 Test Command (Test)

The test command is intended to test the connection between MS and TE.

Type field:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	0	1	0	0

The length byte indicates the number of test bytes sent in the value bytes. The responding station should answer with exactly the same bit sequence. The test command is used for the version control. For more detailed information see Chapter 5.

4.3.9 Modem Status Command (MSC)

The Modem Status Command is used for software flow control.

Command	Length	DLCI	V.24 signals	Break Signals (optional)
1 octet	1 octet	1 octet	1 octet	1 octet

Command:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	0	1	1	1

Figure 11: Modem status command (MSC)

C/R bit: Indicates whether the sequence is a command or a response.

Length: Length = 2 , EA-Bit = 1

DLCI:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	1	DLCI					

V.24 signals:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	FC	RTC	RTR	reserved 0	reserved 0	RING	DCD

FC bit: Flow control, included in all multiplexer versions

FC = 1: no frames are accepted

The following bits for V.24 status lines are included in the Siemens multiplexer protocol from version 3 onwards. To benefit from these functions activate at least version 3. Otherwise version 1 will be used. See Section 5.2 and Section 5.3 for details on version control handshake and version differences.

Direction host application → module (for request only) as of multiplexer protocol V3:

RTC: mapped to DTR
RTR: mapped to RTS
Bit 5, 6, 7, 8 are not valid.

Direction module → host application (for request only) as of multiplexer protocol V3:

RTC: mapped to DSR
RTR: mapped to CTS
RING: mapped to RING
DCD: mapped to DCD
Bit 5, 6 are not valid

Note:
The mappings are valid only for an MSC request as of multiplexer protocol version 3. Information on older multiplexer protocol versions can be gathered from Section 5.3.
The response to any MSC must be always the same data already sent. Please keep in mind that it is impossible to remap any response bits.

Break signal (optional):

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	Not supported						

Usually the break signal octet carries information about a break condition detected from the host application in the data stream for the DLC.

Note:
This command supports no parameters. Instead we use this optional parameter to transport the escape sequence detection from the host to the module. If the customer application detects an escape sequence (usually +++), it sends this optional octet with bit 1 set to 1. The module calls its original escape sequence.

4.3.10 Power Saving Control (PSC)

The power saving control message uses the following type field octet:

Type:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	0	0	1	0

C/R bit: Indicates whether the sequence is a command or a response.

Length: The length byte contains the value 0 (no value octet) or 1 (one value octet).

Value octet (Length=1)

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
P1	P2	P3	P4	0	0	0	0

The P-bits are defining the parameter value.

In commands:

Bit 1	Bit 2	Bit 3	Bit 4	Description
0	0	0	0	Switches to the same mode as without a value octet
1	0	0	0	Switches into full functionality mode, like AT+CFUN=1
0	1	0	0	Switches into NON-CYCLIC SLEEP mode, like AT+CFUN=0
1	1	0	0	Switches into CYCLIC SLEEP mode, like AT+CFUN=5
0	0	1	0	Switches into CYCLIC SLEEP mode, like AT+CFUN=6
1	0	1	0	Switches off, like AT^SMSO
0	1	1	0	Resets, like AT+CFUN=1,1
1	1	1	0	Switches into CYCLIC SLEEP mode, like AT+CFUN=7
0	0	0	1	Switches into CYCLIC SLEEP mode, like AT+CFUN=8
1	0	0	1	Switches into CYCLIC SLEEP mode, like AT+CFUN=9

All wake up events and details of the CYCLIC and NON-CYCLIC SLEEP mode are specified in [2].

In responses:

Bit 1	Bit 2	Bit 3	Bit 4	Description
0	0	0	0	Failure
1	0	0	0	Success

No Value octet (Length=0)

Switches into SLEEP mode, like AT+CFUN=0

Note: According to the GSM 07.10 standard PSC supports no value octets. The optional value octet was added to increase flexibility.

Developed as a substitute to the AT+CFUN command, PSC messages are recommended to control the various SLEEP modes and to reset the mobile. Be sure not to enter any PSC messages until after all responses to AT commands have been received and, in the case of a received URC, the logical ring line has been activated for 1 second and deactivated again. Please note that the behavior of the logical ring line is identical with the behavior of the physical RING0 line described in [1].

4.3.11 Non-supported Command Response (NSC)

This response is sent whenever a command type is not supported by the receiving entity.

Type field:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	1	0	0	0

C/R bit: Indicates whether the sequence is a command or a response.

Value octet:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
EA	C/R	Command type of the non-supported command					

C/R bit: Returns the same value as in the received, non-supported command

Frames not recognized by the receiving entity are responded by an NSC frame.

4.4 Example: Establishing Logical Channels without Parameter Negotiation

- Send AT+CMUX=0; wait for the response
- Send Request SABM for DLCI = 0; wait for the response
- Send Request SABM for all requested DLCIs; wait for the response

As a result the multiplexer is established and information / data can be transmitted (⇒ ready for Information Transfer).

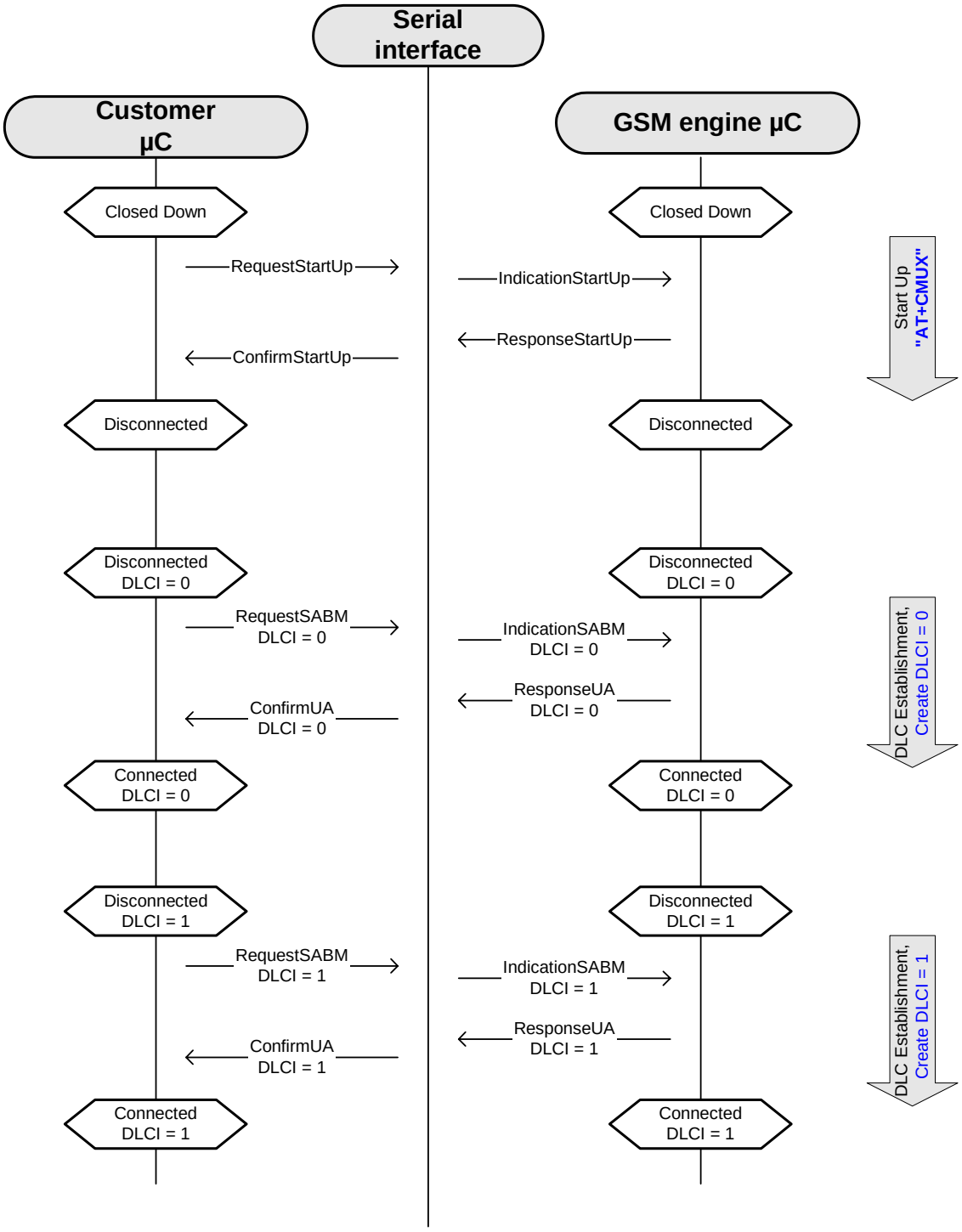


Figure 12: Establishing the multiplexer control channel and the logical channel

5 Multiplexer Protocol Version Control

5.1 Introduction

The multiplexer protocol offers a version control to ensure that TE and MS support the same extent of functionality and to maintain upward and downward compatibility when later firmware versions of the GSM engines are released. The implementation of version control is a subset of the GSM 07.10 standards.

5.2 Activating a Siemens Multiplexer Protocol Version

When the multiplexer is started, the host application (the TE) and the connected module (the MS) negotiate which multiplexer protocol version to use. If TE and MS do not support the same multiplexer protocol, the lower version will be agreed upon. In other words, the highest protocol version common to both sides will be used.

If no version check is done the TE reverts, due to lack of version information, to multiplexer protocol version 1. This means that both sides only agree on version 1, even though they may have the same and even higher version.

The TE and MS multiplexer protocol version numbers can be traced on the serial interface. They are indicated in the following format:

- TE version (e.g. version 1): TEMUXVERSION001
- MS version (e.g. version 2): MSMUXVERSION002

The firmware of the Siemens products and the ready-to-use Windows drivers delivered by Siemens integrate a version control. When designing the host application based on your own sources take care to implement the version check as well, especially if you wish to upgrade to later firmware releases. **It is strongly recommended to implement the latest multiplexer protocol version available.**

5.3 Overview of Siemens Multiplexer Protocol Versions

This chapter summarizes the differences of the existing multiplexer protocol versions.

Siemens multiplexer protocol version 1:

- No version check.
- No break signal is sent.

Siemens multiplexer protocol version 2:

- First version including the version check.
- Additional features: Transparent signals DTR and RTS, escape sequence +++ transportable via MSC.

Siemens multiplexer protocol version 3:

- Includes all features of version 2.
- Transparent signals DSR, CTS, RING and DCD.
- Send MSC request from module to host after version check on every channel to signal the initial state.

Siemens multiplexer protocol version 4:

- Includes all major features of version 3.
- Implements chapter 6 of the GSM 07.10 standard "Error recovery mode option".
- MSC frame will still transmit V.24 status lines as in V3, but the flow control bit (FC) is always ignored.
- See Section 5.5 for Siemens specific implementation details.

Modem status command (MSC):

Command	Length	DLCI	V.24 signals	Break Signals (optional)
1 octet	1 octet	1 octet	1 octet	1 octet

Command:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	C/R	0	0	0	1	1	1

Figure 13: MSC as used in multiplexer protocol version.

V.24 signals in multiplexer protocol version 3:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	FC	RTC	RTR	reserved 0	reserved 0	RING	DCD

Figure 14: V.24 signals as used in multiplexer protocol version 3

V.24 signals in multiplexer protocol version 4:

Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8
1	FC (ignored)	RTC	RTR	reserved 0	reserved 0	RING	DCD

Figure 15: V.24 signals as used in multiplexer protocol version 4

Table 7 summarizes version specific differences in handling the modem status command MSC and the V.24 signals.

Table 7: Version differences for MSC

Version number	RTC	RTR	RTC	RTR	RING	DCD	FC
	Host application ⇒ Module		**Module ⇒ Host application**				
1	1	1	Not used				Used
2	DTR	RTS	Not used				Used
3	DTR	RTS	DSR	CTS	RING	DCD	Used
4	DTR	RTS	DSR	CTS	RING	DCD	Set, but ignored

5.4 Implementing Version Control

The TE initiates the version check by sending the Test command via the multiplexer control channel DLCI 0 (with TEMUX_Version).

As specified in the GSM recommendation 07.10 (Chapter 5.4.6.3.4) the opposite entity shall respond with exactly the same value bytes.

The MS shall return the Test command response with the same contents for the verification pattern. Hereafter the MS shall send a Test command message (with MSMUX_Version) to the TE, and the TE shall respond with the same contents. After sending the response a version compare is made on both sides. As a result, both sides shall agree upon the same multiplexer protocol version.

5.4.1 Troubleshooting

When the MS realizes the implemented software but the TE does not respond correctly, the following error might occur:

- The "Request Test" message is not sent from the TE:

  No version check takes place. No retransmission for "Request Test" message is triggered. The multiplexer starts with protocol version 1 because no version information was exchanged between TE and MS.

However, it is possible to send test commands with "any contents" (except for test messages with the specific IEI for the version check). If a test command with "any contents" is sent, it has to be sent back to the originator with the same contents.

5.4.2 „TestCommand" Message Coding

The coding of the multiplexer stack version is used specifically for SIEMENS equipment and is not defined in ETSI standards. The IEI values defined for the verification pattern of the "TestCommand" message are indicated in Table 8. See GSM recommendation 07.10, Chapter 5.4.6.3.4).

Table 8: IEI coding

IEI coding								**Information element name**
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	
0	0	0	0	0	1	0	0	TEMUX_VERSION
0	0	0	0	1	0	0	0	MSMUX_VERSION
Other values								reserved for future use

For easier analysis of multiplexer traces the message shall be sent in the following format:

1 Version IEI

2 TEMUXVERSION/MSMUXVERSION (send as ASCII)

3 Version Number (1...999 send as ASCII)

The message part after the Version IEI is coded with ASCII characters. This allows to read the version information from the trace file.

The version number must have a value between 1-999. If not all digits of the version number are used only the used digits are coded as ASCII sign(s). Digits that are not used are sent as zero string in the test message.

5.4.3 Example of „TestCommand“ Message

An example for coding a “TestCommand” message is illustrated in Table 9.

Table 9: Coding of “TestCommand” (Example)

Information element name	
0x	
F9	START Flag
03	Address Field DLCI=0,C/R=0,EA=0
EF	Control Field UIH Frame, P/F=0
25	Length LENGTH=18, EA=1
23	Type Field **TestCommand**, C/R=1, EA=1
21	Length Length=16, EA=1
04	TEMUX_VERSION
54	T
45	E
4D	M
55	U
58	X
56	V
45	E
52	R
53	S
49	I
4F	O
4E	N
39	Version number = 999
39	
39	
XX	FCS (is calculated)
F9	END Flag

5.5 Siemens Multiplexer Protocol Version 4

The major advantage of the proprietary Siemens multiplexer protocol version 4 is that it integrates the error recovery mechanism specified by the GSM 07.10 standard to provide better security: Modules with multiplexer protocol version 4 are using a different method of HDLC framing to enable more secure data transmission on the virtual ports and to avoid loss of data when using high data transfer rates and heavy data traffic. Multiplexer protocol version 4 also improves flow control.

It is assumed that you are familiar with the GSM 07.10 standard and, in particular, its chapter 6 "Error recovery mode option". Therefore, the present chapter discusses only the specifics of the Siemens implementation and the differences over GSM 07.10.

5.5.1 Activating Siemens Multiplexer Protocol Version 4

The procedure of activating multiplexer protocol version 4 is the same as in all previous versions. So, there are two ways to start version 4:

- If you are using your own multiplex sources take care that the TE and MS agree upon value 004 as described in Section 5.2 and Section 5.4.
- The ready-to-use WinMux2k 03.000 driver automatically launches multiplexer protocol version 4 if the MS supports the same version. If the firmware of the MS does not yet incorporate version 4 the next lower common version will be negotiated.

5.5.2 Usage of Frames in Siemens Multiplexer Protocol Version 4

With the Siemens multiplexer protocol version 4, basic mode with UIH frames is still used on the control channel, but is no longer supported on the data channels.

After activation of multiplexer protocol version 4, UIH frames are not parsed anymore on the data channels because now a proprietary Siemens frame type, referred to as UIH_E frame, applies. The MS does *NOT* check the PN frame type parameter of a channel to perform DCLI parameter negotiation as defined in the GSM 07.10 standard. Instead, it will always code/decode the new Siemens frame type UIH_E on the data channels.

On the control channel, the MS uses the UIH frames for management and control information as MSC or for parameter negotiation like in all earlier multiplexer protocol versions.

Frame types used for error recovery mode as of multiplexer protocol version 4:

- UIH_E — UIH with N(S) and N(R). New Siemens frame type implemented on all data channels to provide better performance.
- RR — Receive ready
- RNR — Receive not ready
- REJ — Reject

Frames and procedures not supported in multiplexer protocol version 4:

- I frames (performance reasons)
- P/F bit checkpoint handling (GSM 07.10, chapter 6.2.2.5.3), use REJ instead

New frame behavior in multiplexer protocol version 4:

- The MSC frame will still transmit V.24 status lines, but the flow control FC bit (Bit 2) is always ignored.
- SABM/DISC frames will reset all internal HDLC states and counters.
- PN frame: Separate parameter negotiation on each channel allows using different configurations on either channel (see below).

Changed behavior of the PN frame:

In multiplexer protocol version 4, the following system parameters can be adjusted on each channel. To configure those parameters, send a proposal, read the response and use the values returned by the module in the host application:

- Windows size: Any value from 2 (default) to 7.
- Frame size: minimum value 98
  maximum value = (internal Rx buffer of MS) / (WindowsSize+1)

The following values are read only in all multiplexer protocol versions (the module does not change them anyway):

- T1: Acknowledgement timer given in 1/100 seconds from 10 to 255.
  The module always returns, within the PN frame, the value 200. So the result of the negotiation is always 200.
- Frame type: = 0x00 (UIH) if multiplexer protocol version $\leq$3
  = 0x95 (UIH_E) if multiplexer protocol version $\leq$4

Changed behavior of UIH frames in multiplexer protocol version 4:

IMPORTANT: Since the module has not enough CPU power to perform I frame handling (ISO HDLC transparency mechanism with complete checksum), it is designed to use UIH_E frames instead. Payload data are always transported with length byte and header checksum. Nevertheless, send sequence number N(S) and receive sequence number N(R) are supported and must be used. The control type is coded as I frame.

The following table shows an example frame with 6 Bytes payload, N(R) is 2, N(S) is 2.

Table 10: Frame example for multiplexer protocol version 4

Startflag	Address Field	Control Field, includes N(R) and N(S)	Length Field	Payload	Header Checksum	Endflag
F9	07	22	0D	0D 0A 4F 4B 0D 0A	30	F9

Coding of the control field:

In multiplexer protocol version 4, the module uses the following control field coding. Please notice that the UIH_E frame type has the same control field coding as the standard I frame in GSM 07.10. N(R) and N(S) are receive and sequence number, the P/F bit (Bit 5) is always set to 0 from the module, and not read on any received frame.

Table 11: Control field coding in multiplexer protocol version 4

Frame Type	1	2	3	4	5	6	7	8
UIH_E payload	0	N(S)			0	N(R)		
RR Receive Ready	1	0	0	0	0	N(R)		
RNR Receive Not Ready	1	0	1	0	0	N(R)		
REJ Reject	1	0	0	1	0	N(R)		

5.5.3 Message Sequence Charts for Error Recovery Mode

List of state variables and frame types used in this chapter

V(S) Send state variable
The send state variable denotes the sequence number of the next in-sequence UIH_E frame to be transmitted.

V(R) Receive state variable
The receive state variable denotes the sequence number of the next UIH_E frame expected to be received.

V(A) Acknowledge state variable
The acknowledge state variable denotes the sequence number of the next non-acknowledged UIH_E frame within the send window.

RR[N(R)] Receive Ready frame with the receive sequence number
The receive sequence number indicates the N(S) sequence number of the next expected UIH_E frame.

RNR[N(R)] Receive Not Ready frame with the receive sequence number
The receive sequence number indicates the N(S) sequence number of the next expected UIH_E frame.

REJ[N(R)] Reject frame with the receive sequence number
A station receiving an REJ frame shall initiate sequential retransmission of UIH_E frames starting with the UIH_E frame indicated by the N(R) contained in the REJ frame.

T1 Acknowledgement timer for starting the UIH_E frame retransmission

I[N(S), N(R)] I frame (Information frame) with two sequence numbers:
- N(S) is the send sequence number of the I frame
- N(R) is the receive sequence number of the next expected I frame to be received.
Consequently, N(R) indicates that the I frames numbered up to N(R) -1 inclusive have been received correctly.

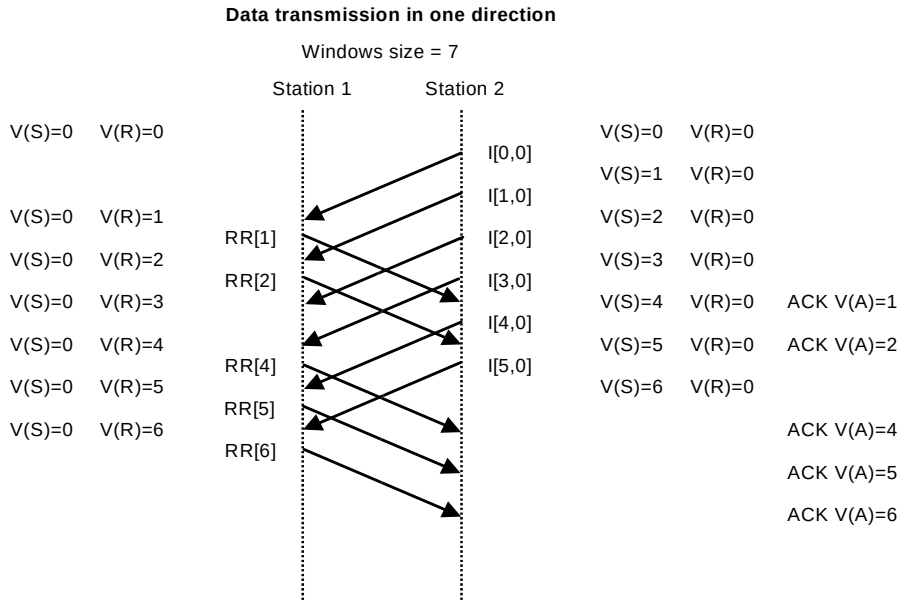


Figure 16: Data transmission in one direction

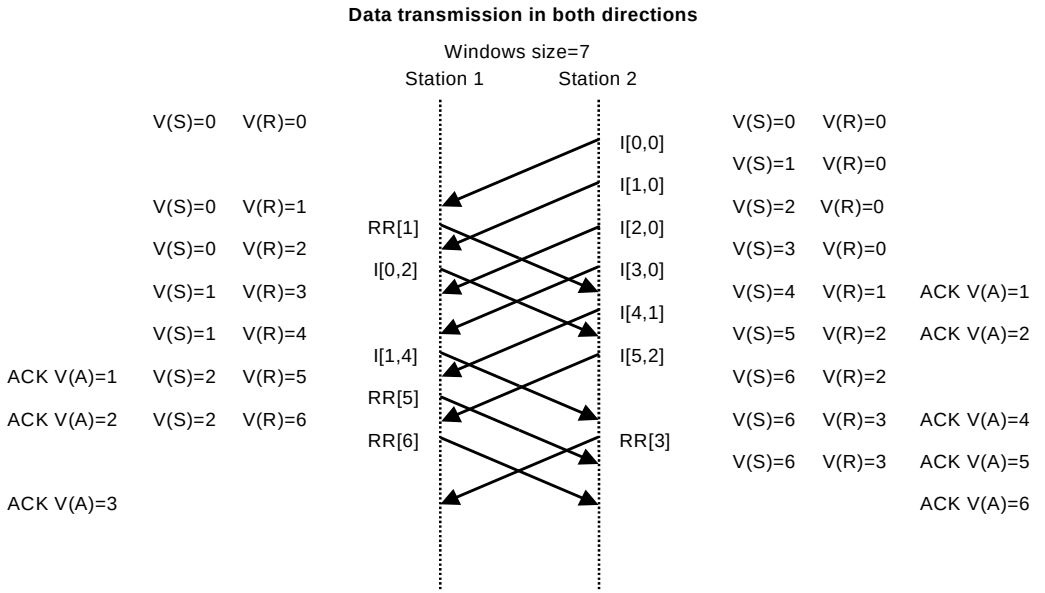


Figure 17: Data transmission in both directions

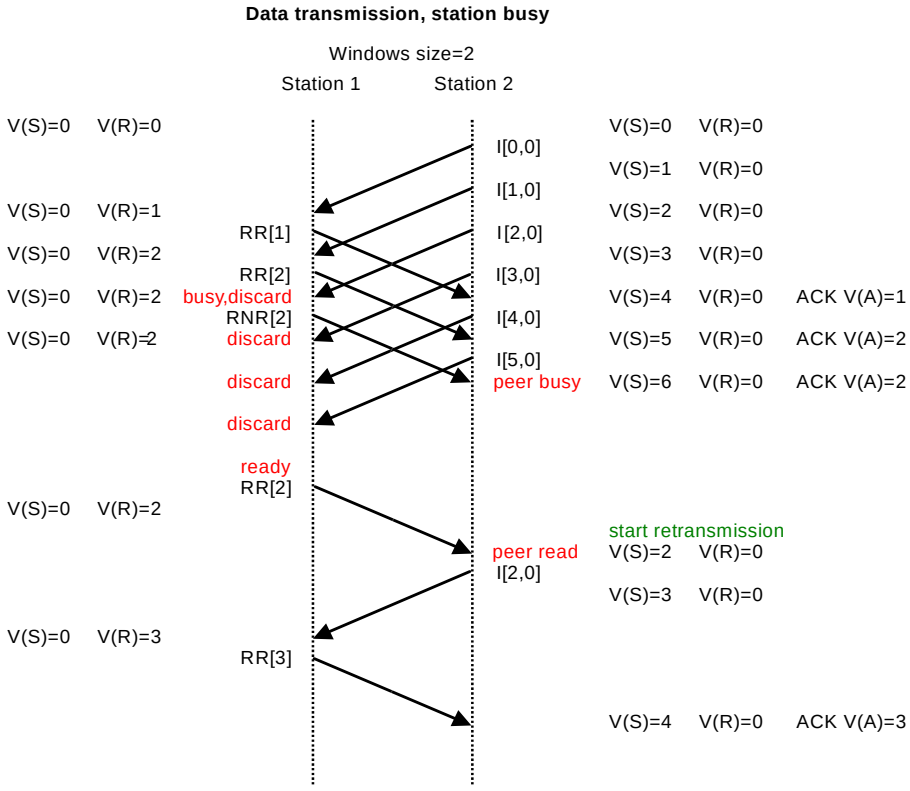


Figure 18: Data transmission to busy station

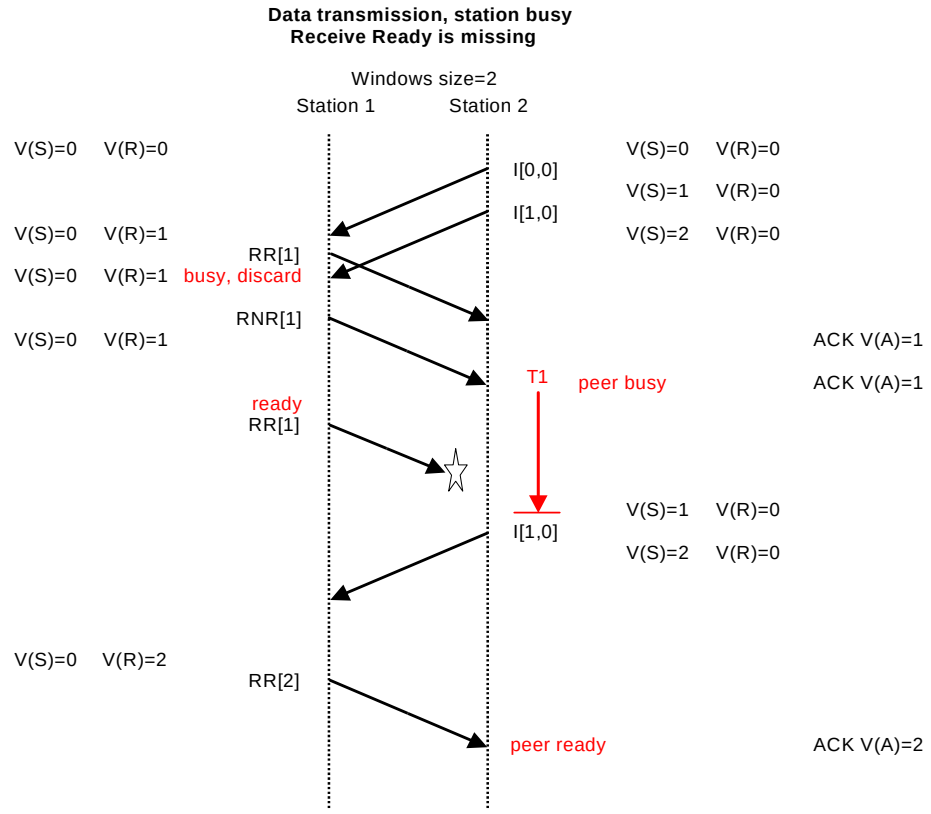


Figure 19: Data transmission to busy station with Receive Ready missing

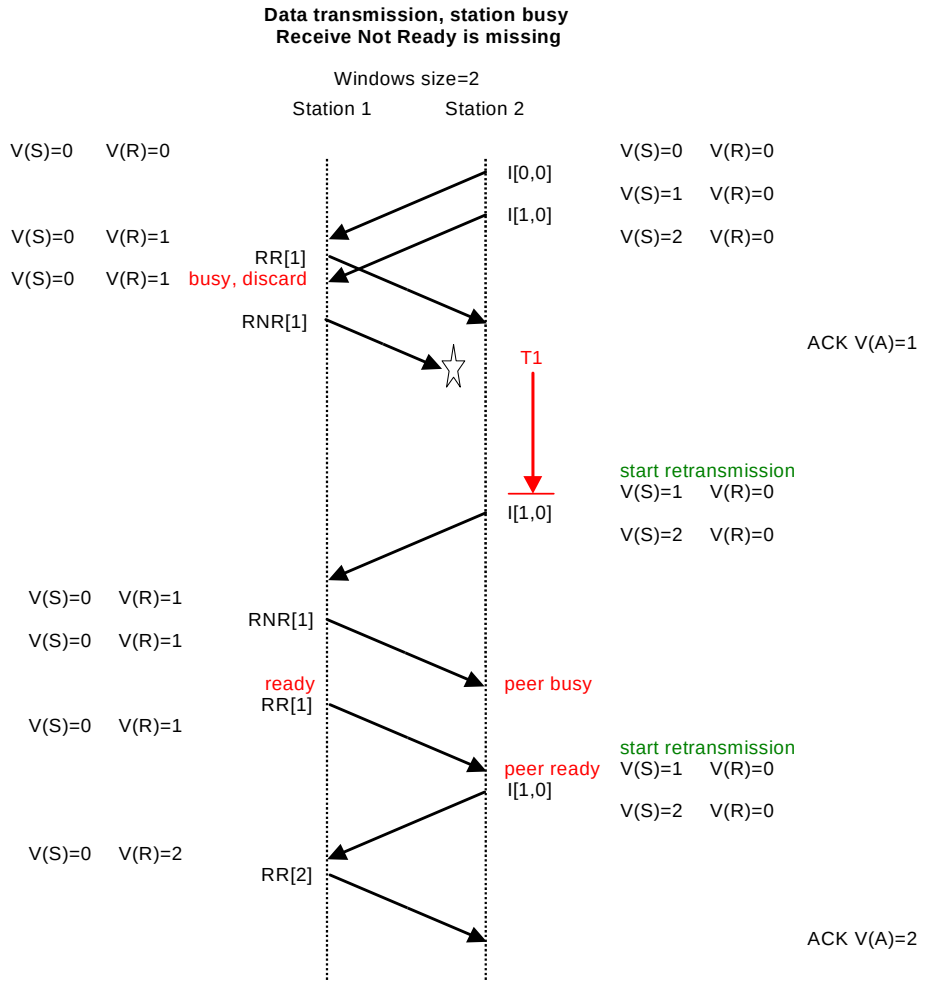


Figure 20: Data transmission to busy station with Receive Not Ready missing

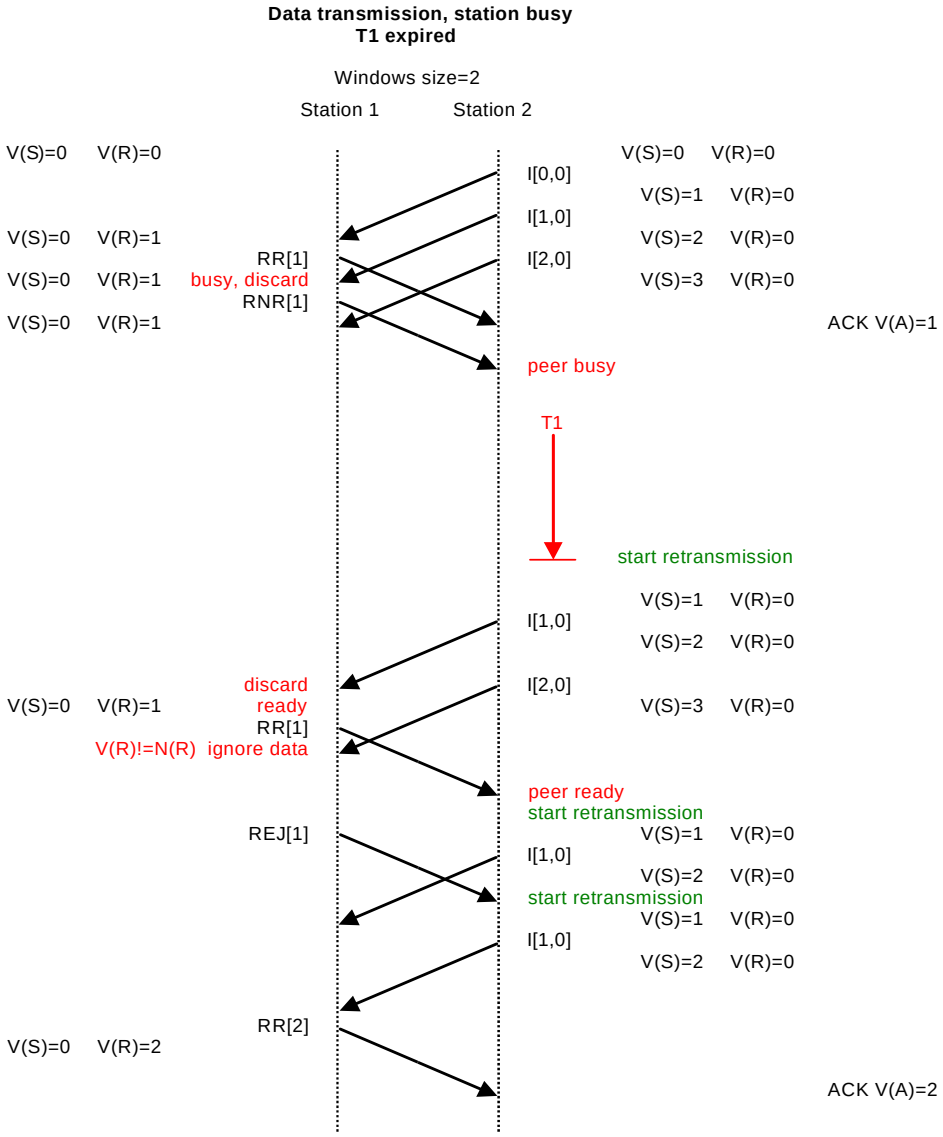


Figure 21: Data transmission to busy station with T1 timer expired

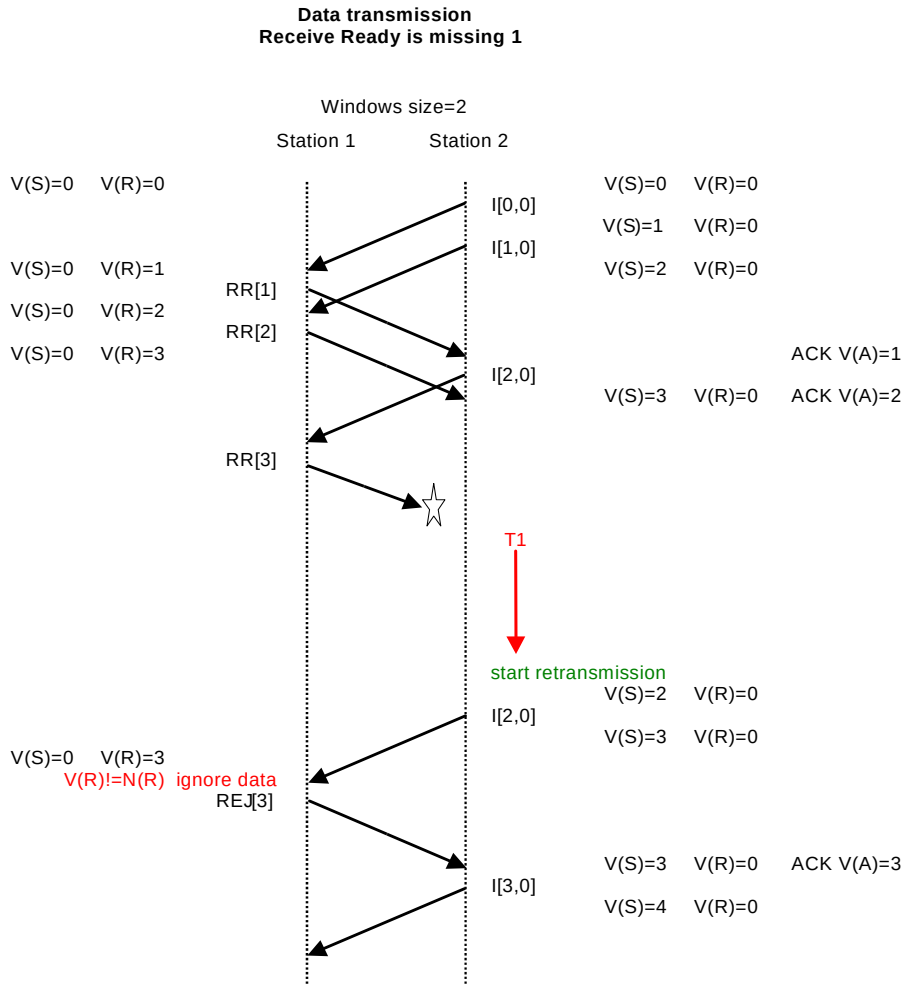


Figure 22: Data transmission with Receive Ready missing, scenario 1

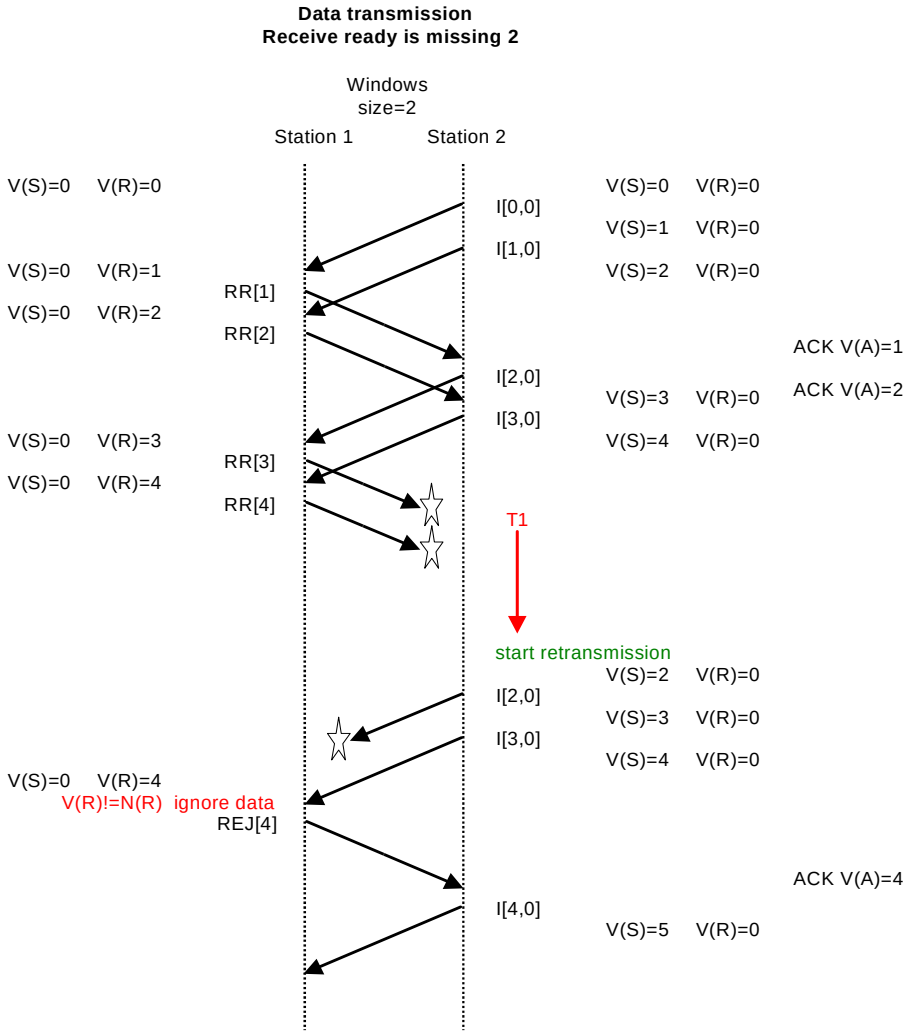


Figure 23: Data transmission with Receive Ready missing, scenario 2

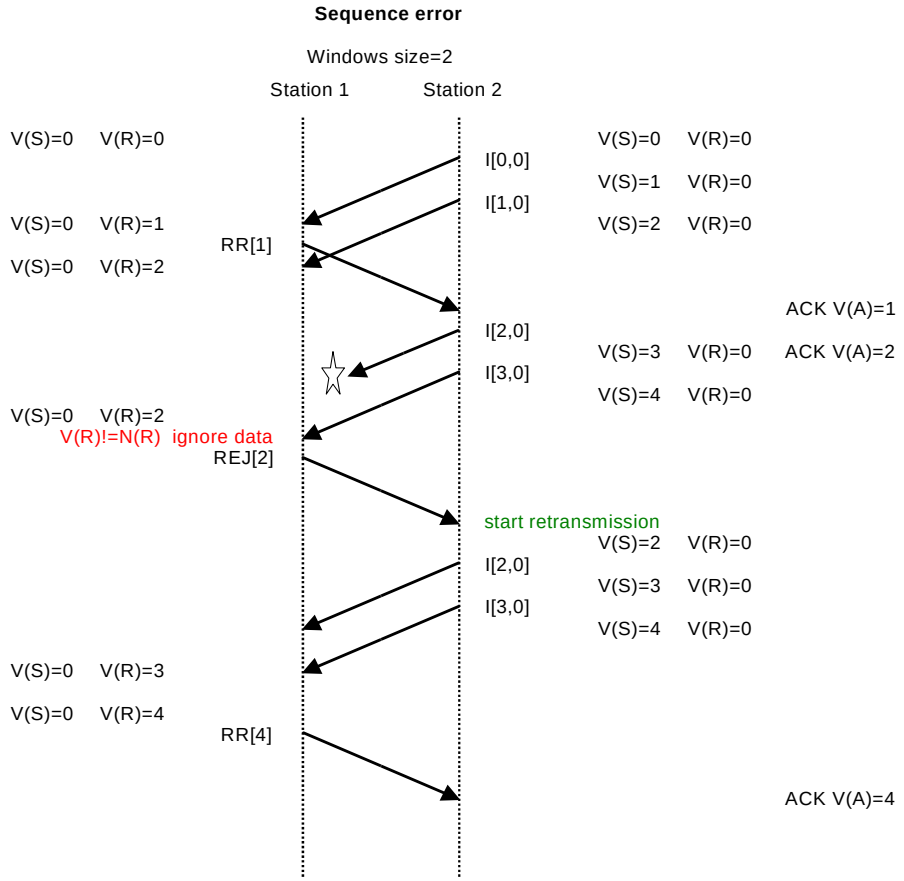


Figure 24: Sequence error

5.5.4 State / Event Machine for Error Recovery Mode

The following state and event machine diagrams represent the workflow on the data channels (DLCI 1…4) when in CONNECTED state and in error recovery mode.

See also the general multiplex state machine diagrams provided in Section 4.3, especially for explanations on the CONNECTED state of the channels.

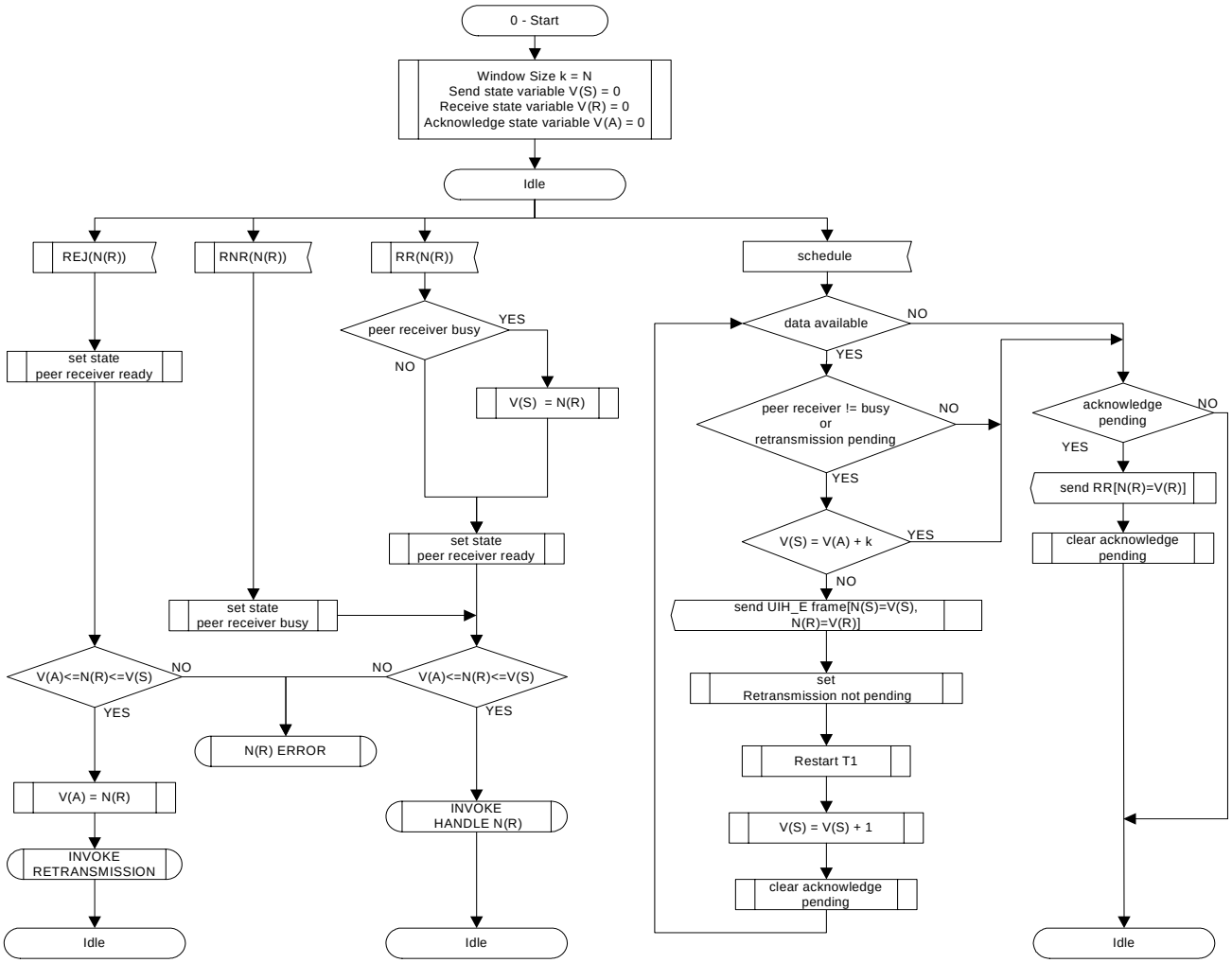


Figure 25: State / event machine (1)

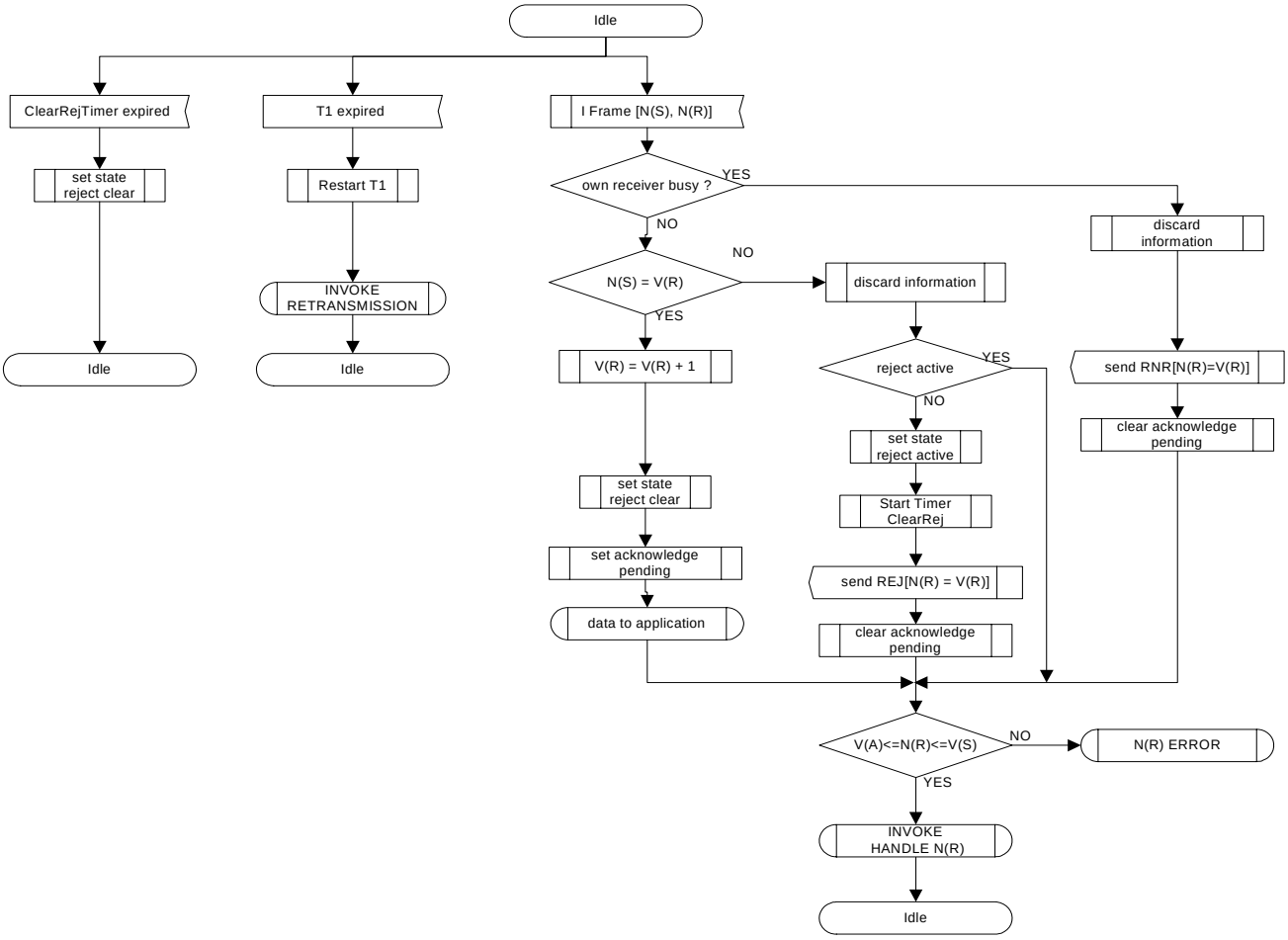


Figure 26: State / event machine (2)

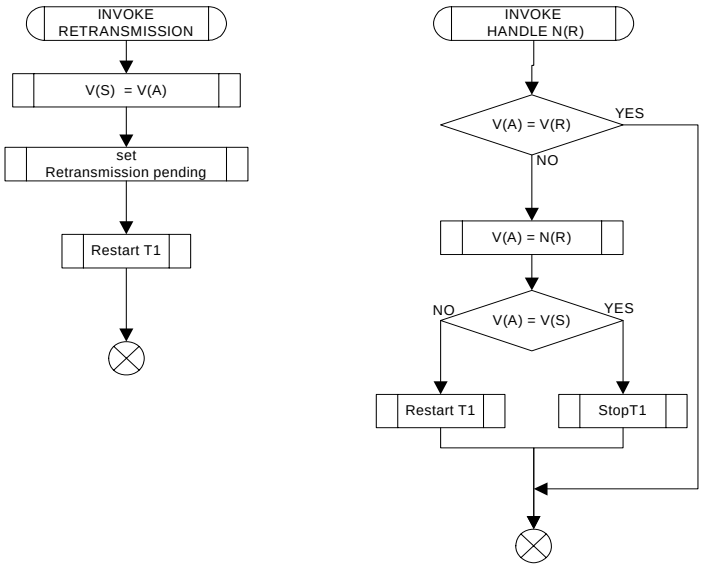


Figure 27: State / event machine (3)

5.5.5 Trace of a Standard Multiplex Protocol Start-up with Siemens Multiplexer Version 4

The following trace shows the normal startup procedure for the multiplex protocol and the standard handshake for activating multiplexer protocol version 4. The example shows that two channels are opened, and a simple AT command is executed.

Table 12: Typographical conventions used in the trace example

Format	Meaning	Example
Courier bold	Command sequence	**41 54 5C 51 33 0D 0A**
Courier bold, highlighted magenta, underlined	Highlights the result of the protocol negotiation on host and module side	**34 00 00**
Courier normal	Command sequence decoded	AT\Q3 command
# mark, Arial italic	Comment	*# Enable hardware handshake*
Uppercase, Arial bold, italic	Important note	***NOTE: FROM NOW….***

Host application	Module
41 54 5C 51 33 0D 0A AT\Q3 command *# Enable hardware handshake*	**41 54 5C 51 33 0D 0D 0A 4F 4B 0D 0A** Echo and OK
41 54 2B 49 50 52 3D 31 31 35 21 30 30 0D 0A AT+IPR=115200 command *# Use fixed baudrate for the multiplex protocol only*	**41 54 2B 49 50 52 3D 31 31 35 21 30 30 0D 0D 0A 4F 4B 0D 0A** Echo and OK *# Module has now a fixed baudrate*
41 54 2B 43 4D 55 58 3D 30 0D 0A AT+CMUX=0 command *# Activate multiplexer protocol on selected UART*	**41 54 2B 43 4D 55 58 3D 30 0D 0D 0A 4F 4B 0D 0A** Echo and OK *# Module switches to multiplex mode. Within the next 5 seconds, the protocol has to be opened.*
F9 03 3F 01 1C F9 dlci=0 MP_IndSABM *# Start up multiplexer protocol, open control channel*	**F9 03 73 01 D7 F9** dlci=0 MP_ConfUA_SABM *# Module confirms that control channel was opened.*
F9 03 EF 25 23 21 04 54 45 4D 55 58 56 45 52 53 49 4F 4E 34 00 00 CA F9 dlci=0 MP_IndTest *# Version control handshake:* *Host sends its version* *Host supports up to multiplex version 4*	**F9 01 EF 25 21 21 04 54 45 4D 55 58 56 00 00 00 00 00 00 00 00 00 AB F9** dlci=0 MP_ConfTest *# Module replies to the test signal*
01 EF 25 21 21 08 4D 53 4D 55 58 56 45 52 53 49 4F 4E 34 00 00 AB F9 dlci=0 MP_ConfTest *# Host replies to the test signal*	**F9 01 EF 25 23 21 08 4D 53 4D 55 58 56 45 52 53 49 4F 4E 34 00 00 AB F9** dlci=0 MP_IndTest # Version control handshake: *Module sends its version.* *Module supports up to multiplex version 4.*

Host application	Module
NOTE: FROM NOW, UIH IS NO LONGER SUPPORTED ON DATA CHANNELS; ALL DATA CHANNELS ACCEPT UIH_E FRAMES ONLY; CONTROL CHANNEL STILL ACCEPTS UIH FRAMES AS USED IN V3	
F9 03 EF 15 83 11 01 00 01 FF F4 00 00 04 EE F9 MP_IndNegotiation dlci=0 used dlci=1 frame type=0 prio=1 T1 ack timer=255 frame len=244 retrans=0 windows size for recover mode=4 *# Parameter negotiation on control channel:* *Host sends parameter set for dlci 1.* *Valid values are frame length and windows size.*	**F9 01 EF 15 81 11 01 85 01 C8 F4 00 00 04 8F F9** MP_ConfNegotiation dlci=0 used dlci=1 frame type=133 prio=1 T1 ack timer=200 frame len=244 retrans=0 windows size for recover mode=4 *# Parameter negotiation on control channel:* *Module responds with its used parameters.* *Valid values are frame type, T1 timer, frame length and windows size.* *The host should use these negotiated values as shown in this example.*
F9 07 3F 01 DE F9 MP_IndSABM dlci=1 *# Open channel 1*	**F9 07 73 01 15 F9** MP_ConfUA_SABM dlci=1 *# Module confirms that channel 1 was opened.*
NOTE: HOST AND MODULE SHOULD NEGOTIATE THEIR V.24 CONTROL LINES. THE MODULE ALWAYS SENDS ITS V.24 LINES AFTER OPENING A CHANNEL. THE HOSTS SHOULD DO IT TOO.	
F9 01 EF 09 E1 05 07 0D 9A F9 MP_ConfMSC dlci=0 ModemStatus used dlci=0x01 value=0x0d *# Host responds. This response should be a copy of MP_IndMSC.*	**F9 01 EF 09 E3 05 07 0D 9A F9** MP_IndMSC dlci=0 ModemStatus used dlci=0x01 value=0x0d flow control = 0 DSR = 1 CTS = 1 RING = 0 DCD = 0 *# Module sends its current V.24 control lines to the host.*

Host application	Module
F9 03 EF 09 E3 05 07 0D FB F9 MP_IndMSC dlci=0 ModemStatus used dlci=0x01 value=0x0d flow control = 0 DTR = 1 RTS = 1 *# Host sends its current v24 control lines to the module*	**F9 01 EF 09 E1 05 07 0D 9A F9** MP_ConfMSC dlci=0 ModemStatus used dlci=0x01 value=0x0d *# Module responds. This response should be a copy of MP_ConfMSC*
F9 03 EF 15 83 11 02 00 02 FF F4 00 00 04 EE F9 MP_IndNegotiation dlci=0 used dlci=2 frame type=0 prio=2 T1 ack timer=255 frame len=244 retrans=0 windows size for recover mode=4 *# Parameter negotiation on control channel:* *Host sends parameter set for dlci 2* *Valid values are frame length and windows size*	**F9 01 EF 15 81 11 02 85 02 C8 F4 00 00 04 8F F9** MP_ConfNegotiation dlci=0 used dlci=2 frame type=133 prio=2 T1 ack timer=200 frame len=244 retrans=0 windows size for recover mode=4 *# Parameter negotiation on control channel:* *Module responds with its used parameter* *Valid values are frame type, T1 timer, frame length and windows size* *The host should use these negotiated values as shown in this example.*
F9 0B 3F 01 59 F9 MP_IndSABM dlci=2 *# Open channel 2*	**F9 0B 73 01 92 F9** MP_ConfUA_SABM dlci=2 *# Module confirms that channel 2 was opened.*
# … status line handshake as on channel 1 …	
	# … status line handshake as on channel 1…
# … open all other needed channels …	

Host application	Module
NOTE: HOST SENDS AT <CR>, MODULE ANSWERS WITH OK AND SOME \R\N FRAMING. THIS EXAMPLE SHOWS THE USAGE OF UIH_E FRAMES.	
F9 07 00 07 61 74 0D 2D F9 UIH_E frame n_r=0, n_s=0 dlci=1 Data(TXT): at <CR> *# Send AT*	**F9 07 21 7F F9** hdlc RR n_r=1 Receiver Ready dlci=1 *# Frame successfully received*
	F9 07 20 07 61 74 0D 07 F9 UIH_E frame n_r=1, n_s=0 dlci=1 Data(TXT): at (echo) *# Echo AT command (ATE1 in the user profile stored with AT&W)*
	F9 07 22 0D 0D 0A 4F 4B 0D 0A 30 F9 UIH_E frame n_r=1, n_s=1 dlci=1 Data(TXT): (\r\r\n) OK (\r\n) *# Execute AT command, send answer*
F9 07 41 37 F9 hdlc RR n_r=2 Receiver Ready dlci=1 *# Both frames (echo and OK) successfully received*	