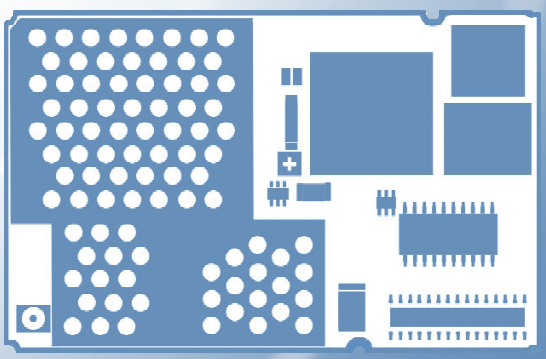
SIEMENS

Multiplexer User's Guide



Siemens Cellular Engines

Version: 02.00
DocID: Multiplexer_UG_V02.00

Wireless Modules

Document Name:	**Multiplexer User's Guide**
Version:	**02.00**
Date:	**February 14, 2003**
DocId:	**Multiplexer_UG_v02.00**
Status:	**Ready for distribution to customers**

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.

Trademark notice

MS Windows® is a registered trademark of Microsoft Corporation.

Contents

Figures

Tables

0 Version history

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

Preceding documents:
Muliplexer User's Guide Version 01.00

New document:
Multiplexer User's Guide Version **Multiplexer_UG_V02.00**

Chapter	Page	What is new
1.1	6	Added supported products
3.2	11	Revised Chapter

1 Introduction

This document describes how to use the multiplexer with your GSM / GPRS engine and to explain how to integrate an external multiplexer into your application. So this guide covers all information you need to design your own (external) multiplexer.
Multiplexer protocol sources (WinMux2k), provided by Siemens AG can be obtained on request from your local distributor. For more detailed information please refer to [5]
The GSM engine supports the basic option of the multiplex mode according to the ETSI TS 101 369, GSM 07.10 Multiplexer protocol. It allows a triple session over a serial link interface. In this document the multiplexer integration in the module is also referred to as internal multiplexer. With the multiplexer mode, AT commands and data are encapsulated into packets. Each packet has a channel identification and may vary in length.
Outside the module on the application side of the serial interface another multiplexer must be integrated, in order to demultiplex the signal and distribute it on the 3 virtual channels. The *external multiplexer* is to be provided by the user.

1.1 Supported products and related documents

Supported products

- AC43
- AC45
- MC35i
- MC45
- MC388
- TC35i
- TC45

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] Remote-SAT User's Guide
[5] Multiplexer Driver Developer's Guide for Windows 2000 and Windows XP
[6] 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.

Prior to using your GSM engine, be sure to carefully read and understand the latest product information provided in the Release Notes.

To visit the Siemens Website you can use the following link:
http://www.siemens.com/wm

1.2 References

[1] 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 Version 7.1.0, Release 1998

1.3 Term and abbreviations

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
PC Mux	Multiplexer Simulation
RTS	Request to Send
TE	Terminal Equipment
UART	Universal Asynchronous Receiver Transmitter

2 Multiplexer protocol - overview

2.1 Product concept and architecture

The multiplexer mode enables one serial interface to transmit data to three different customer applications. This is achieved by providing three virtual channels using a multiplexer (Mux).
This is especially advantageous when a fax/data/GPRS call is ongoing. Using the multiplexer features, e.g. controlling the module or using the SMS service can be done via the additional channels without disturbing the data flow; access to the second UART is not necessary.
Furthermore, several accesses to the module can be created with the multiplexer. This is of great advantage when several independent electronic devices or interfaces are used.
To access the three virtual interfaces available, both the GSM engine and the customer application must contain Mux components which communicate over the Multiplexer protocol.

Note
All statements regarding GPRS are valid for AC43, AC45, MC45 and TC45 only.

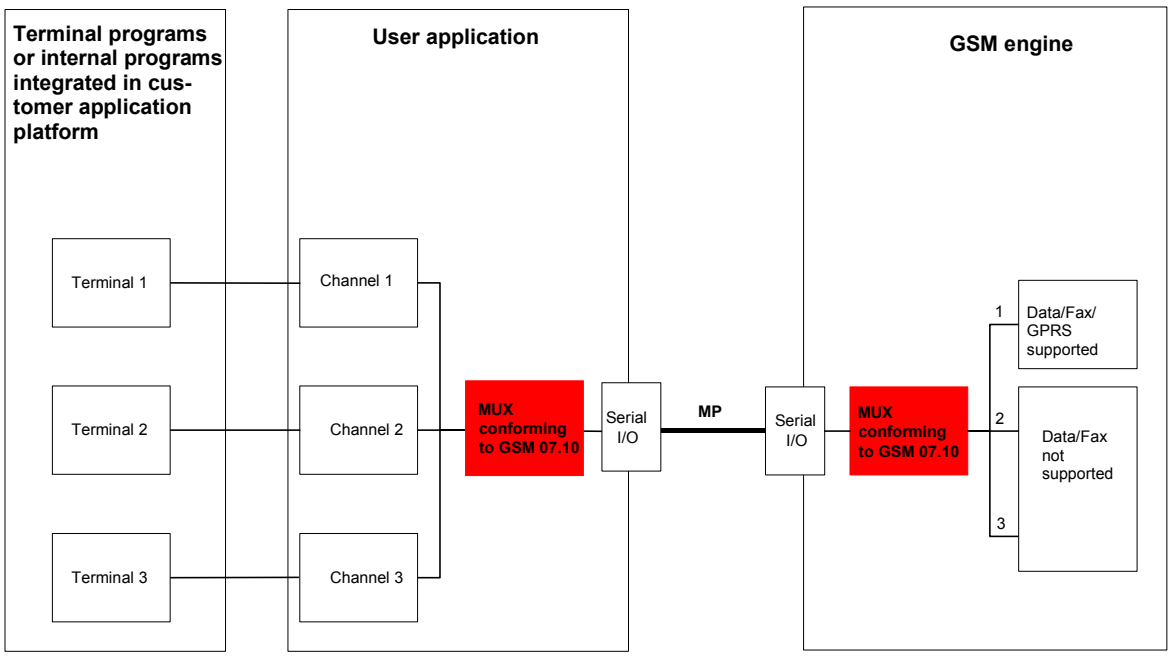


Figure 1: Multiplexer architecture

2.2 Virtual channels and AT commands

Please note, that a module including the multiplexer 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 multiplexer channels

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

● indicates that the functionality is available on the channel
--- indicates that the functionality is not available on the channel
1) except for AT commands related to data and fax calls
2) only two channels can be used parallel to transmit GPRS data

Examples

- While a data call is in progress on channel 1, you can send a SMS on channel 2 and edit the phonebook on channel 3. 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.
- When receiving a fax on channel 1, you are able to check the battery capacity using the appropriate AT command on channel 2 or 3.

Note

Based on the technical requirements of the multiplex mode, data and fax calls can only be set up on logical channel 1 while GPRS connections can be established on every channel. However, AT commands have a different behavior on channels 2 and 3, for details refer to chapter 3.2.
Additional information regarding restrictions and interferences between the channels refer to chapter 3.2.

3 Integrating the multiplexer into the customer application

If you wish to use your own design of the multiplexer, you may use the sources delivered by Siemens upon request, see [5]. In this case you receive a *.zip file, including data you can use to integrate the multiplexer in your application. Of course you may create your own implementation of the multiplexer.

3.1 Tips and Tricks

3.1.1 Timeout after starting the multiplexer

Please note that a 5 second timeout must be observed after starting the multiplexer. At this stage no AT commands should be entered. If the multiplexer fails to start, the normal AT interface will be available again.

3.2 Restrictions

After establishing the multiplexer protocol three logical channels are available. Please keep the following restrictions in mind:

- The GSM engine supports the basic option and UIH Framing according to GSM 07.10.
- MO and MT data and fax calls can only be set up on channel 1
- Unsolicited Result Codes will generally be transmitted to all logical channels (like “RING”)
- In charge-only and in alarm mode the multiplexer is not available
- See [2] for multiplexer characteristics with the commands AT&V, AT+CNMI, +CALA and +CMEE
- A voice call can be answered or ended on every channel. See AT commands like ATD, ATA, ATH
- ATH terminates every data call regardless on which logical channel ATH was executed, for details see [2]
- XON/XOFF flow control is not supported in multiplex mode (AT\Q1)
- Autobauding is not compatible with multiplex mode. It is neither possible to start Mux when autobauding is active, nor to set autobauding during multiplex mode.
- When switching on the module after a firmware update we recommend to wait for 5 seconds before entering the first AT command

Please note that the multiplexer protocol cannot be established if:

- Autobauding is activated
- XON/XOFF flow control is activated
- Character framing is not configured for 8 data bits, no parity and 1 stop bit (TC35i only)

Serveral customer software application 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-synchronizes all settings before using the multiplexer mode again.

The following table lists commands and features for which dependencies between the different instances on the 3 channels may exist

Table 2: Commands with possible channel dependencies

Access types valid for all channels
Call Control
Phonebook access
SIM Card access
RF settings
Time settings
ATZ, AT&F, AT&V, AT&W, AT+CEER, AT+CLCK
Network settings
Power save
Device locks
SMS read, write and delete
Baud rate settings

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 fax call will be ended as well.
- The table below is intended to remind the customer to be aware of possible dependencies when using the listed commands/features on different channels. One way of avoiding the problems may be to dedicate certain commands/features to certain channels or to assure that the application avoids conflicts.

Table 3: Commands/Features without channel dependencies

Access types for Channel 2 and 3 not disturbing other instances
URCs
RING voice calls
Read device information
Signal quality and monitoring

Table 3 gives an overview about events, which may be ongoing on different channels independently.

3.3 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.

3.3.1 Flow control

Logical flow control

The internal logical flow control (FC-BIT in MSC message, see Chapter 5.1.6.3) represents the existing flow control to the module. For example, if a data call is initiated and the customer application transmits data to the module on this channel, the module will stop the data transmission from time to time. This happens because the module operates with a bandwidth of 9k6 on air, but the customer application uses a larger width. In this case the module sends a MSC message with FC-BIT set. After all data stored in the internal buffer have been sent, the module will send a second MSC message with FC-BIT reset. As already pointed out, the logical flow control operates like RTS/CTS but with FC-BIT on every channel. The RTS/CTS 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.

RTS/CTS on the physical channels

Flow control (AT\Q3) must be used before the multiplexer is switched on. The customer application de - and encodes the data. It needs to respect the flow control given by the MP. The flow control is transmitted in the data flow and contains information on whether a channel is allowed to send or not. This is necessary for the cyclic power save modes, for detailed information see [1] and Chapter 3.4.
The customer application must set RTS (in the direction of the module) on channel 1. It shall neither be used nor switched off to prevent loss of data (control data cannot send in that case). If flow control is needed, it is recommended to use logical flow control on every channel.

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.

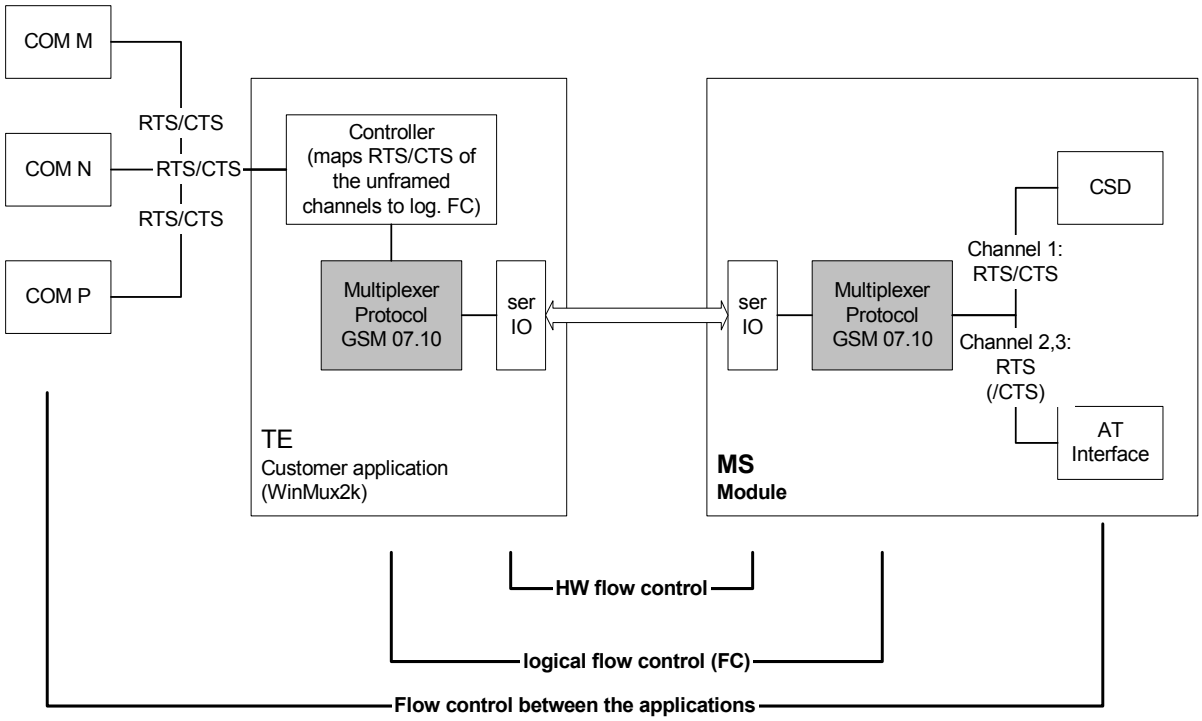


Figure 2: Logical flow control and RTS/CTS signaling behind the decoder

Ring/DCD

Contrarily to all other lines DCD and Ring are transmitted additionally on the UART directly by the module. These signals are logical OR's from the three logical channel status lines. However, the customer application must carefully decide how to handle these lines and it should ensure, that no conflicts occur between the different channels. E.g. in some situations it may be advisable to only display RING on channel 1.

Please keep in mind that a call can only be accepted once. Therefore some kind of mutual locking mechanism must be used.

3.3.2 Escape sequence

When the MP 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. Please 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 the „+++“ on the customer multiplex application and transport this information via the MSC signal to the module (see Chapter 5.1.6.3).
- As an alternative, ATH may be sent on one of the other channels, for more detailed information please refer to [2].

3.4 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 is set with the AT+CFUN command which provides the choice of the functionality levels <fun>=0, 1, 5, 6, 7 or 8, all explained below. Further instructions you need to use AT+CFUN can be retrieved in [1] and [2].

3.4.1 AT+CFUN=1,1 (Reset procedure)

After executing the AT command AT+CFUN=1,1 the module resets and the multiplexer must be restarted. Please note, that this procedure causes the loss of all multiplexer settings, which have not been stored.

3.5 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. 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. Due to the varying length of the transmitted data packets no fixed header can be sent.
To summarize, if the module is set to 115200 bps and an incoming GPRS call requires 5 kByte, the two other channels have to operate within the range of the remaining 5 kByte.
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.

4 Structure of the multiplexer protocol

4.1 Introduction of the multiplexer protocol

The supported multiplexer protocol confirms the GSM 07.10 Multiplexer Protocol. The non-error recovery mode was implemented with the basic option.
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 calls can be established simultaneously on every channel.
The multiplexer protocol must be started and the logical channels opened in compliance with specified procedures.

The chapter also gives you suggestions for the following aspects:

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

4.2 Data link layer

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

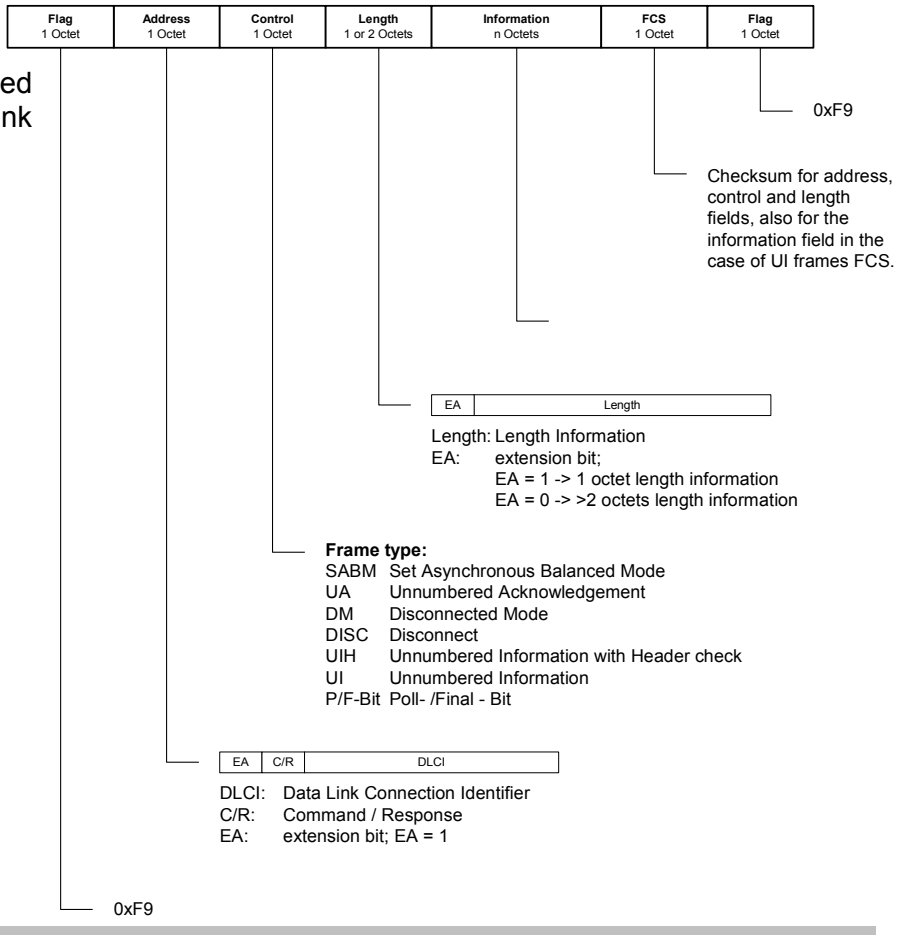


Figure 3: Data link layer

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 4: Address field

EA	C/R	DLCI

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

Table 5: Assignment of the DLCI

	DLCI number (decimal)	Priority
Multiplexer control channel (see chapter 5.1.6)	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 6: 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 7: 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.
The default length is 31 bytes, but can be extended with the E/A bit.

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:

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 source computer. The receiving computer performs the same calculation. If the receiving computer's calculation does not match the result sent by the source computer, the packet is judged corrupt and discarded. An FCS calculation is made for each packet.

5 State diagrams

The multiplexer protocol is based on two state machines (see Figure 4).
One state machine initiates the setup of the logical channels, while the other responds to the requests.
The GSM engine 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. In the case 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 switches back.

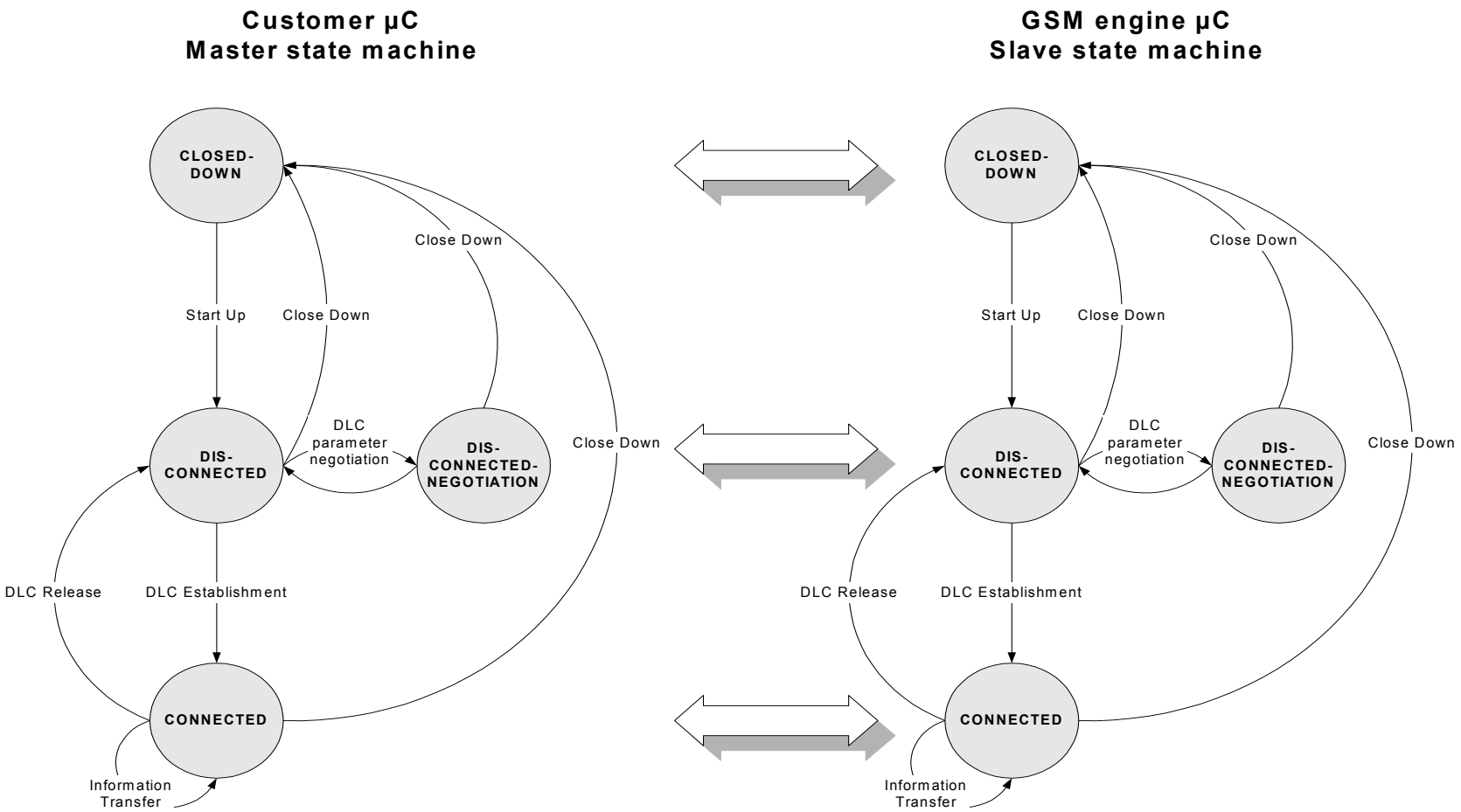


Figure 4: Relationship between the customer µC and the GSM engine µC

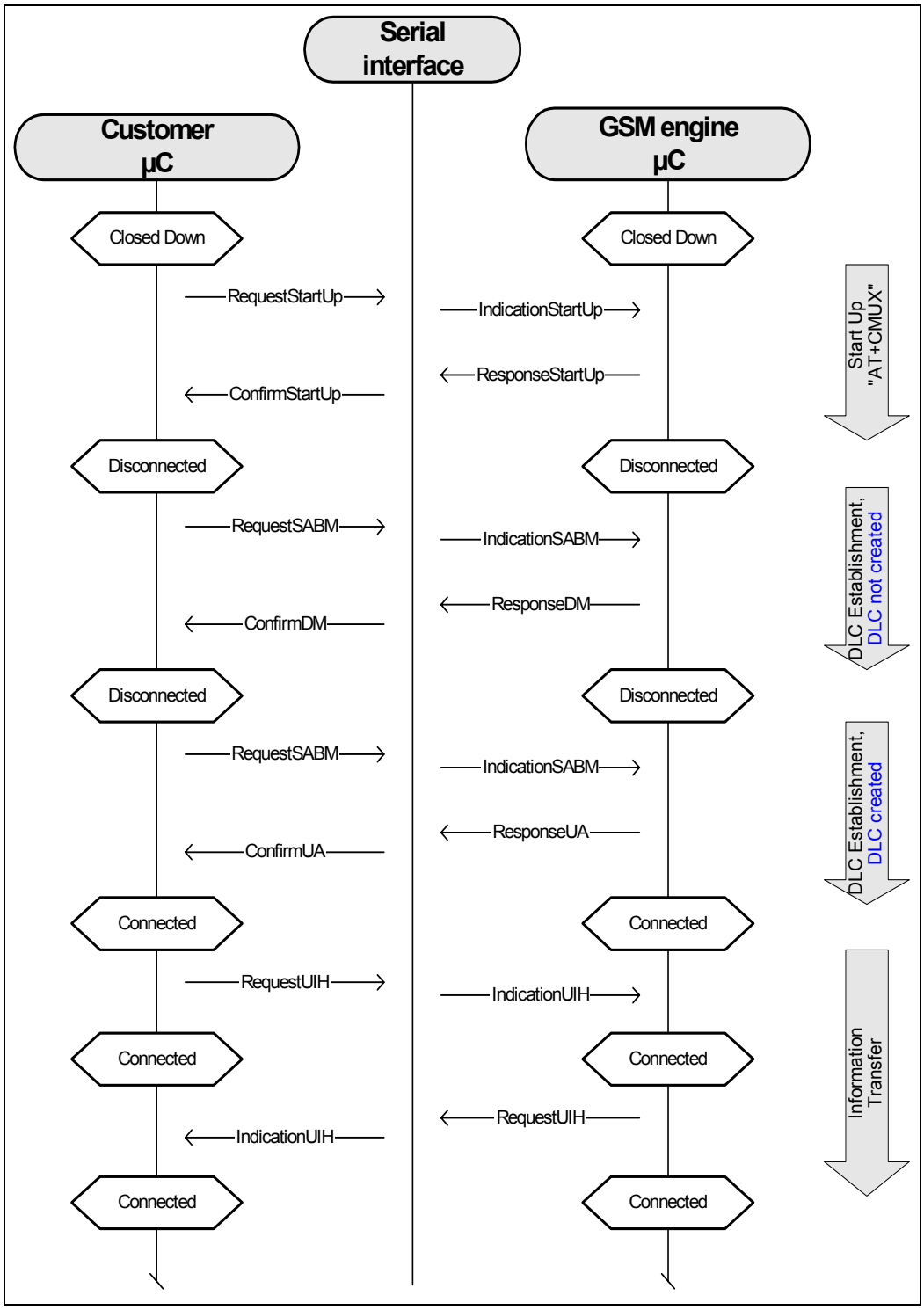


Figure 5: MPI – Startup, DLC establishment and information transfer

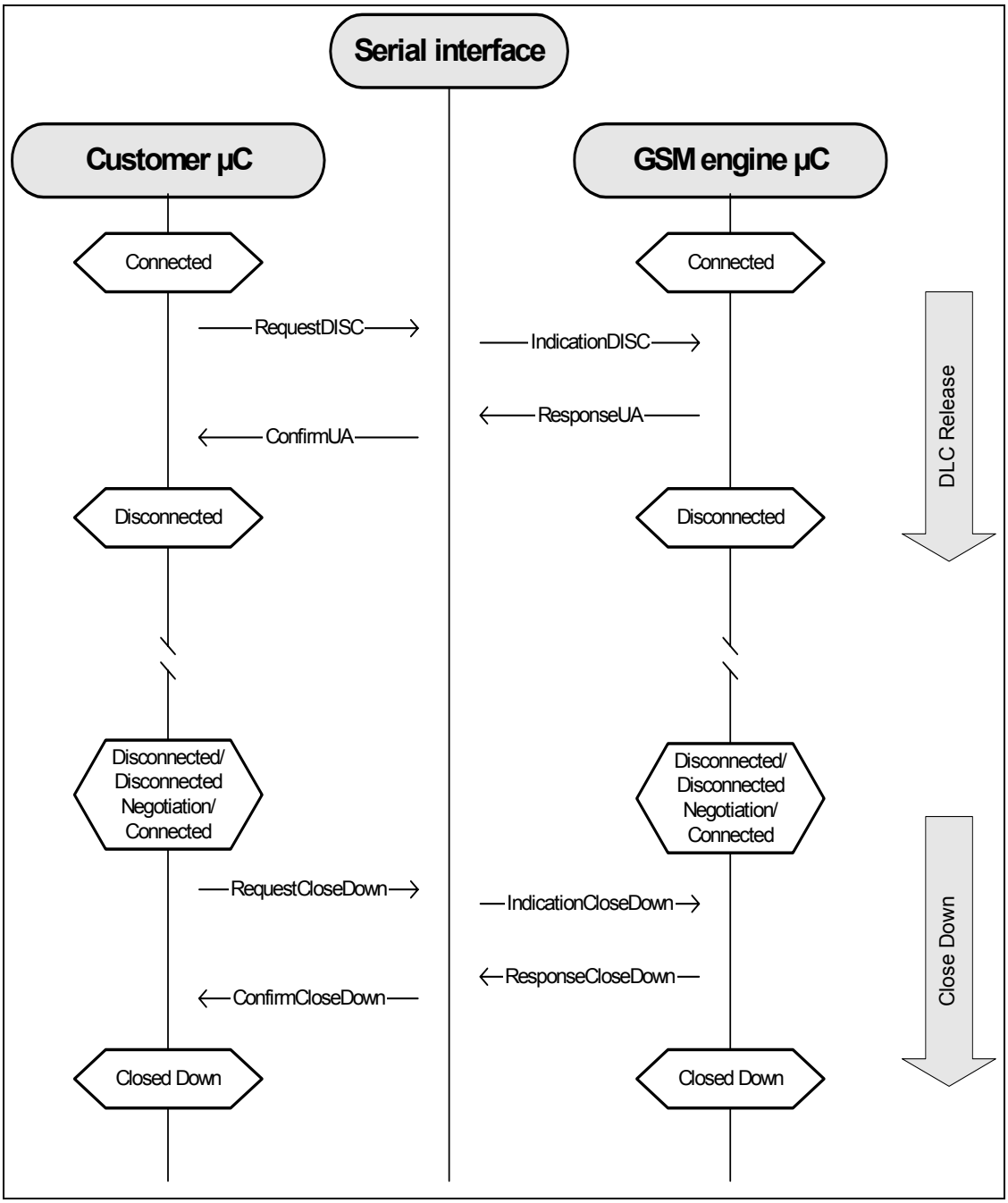


Figure 6: MP - DLC release and close down

5.1.1 Start up procedure

The only approach to activate the multiplexer protocol is the entering of the appropriate AT command AT+CMUX=0. It enables the multiplexer control channel. Please keep in mind that as the next step the multiplexer control channel must be set up, see Chapter 5.1.2.

5.1.2 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 Chapter 4.2.3) must be sent to the GSM engine.
The module responds with either a UA frame – i.e., 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".

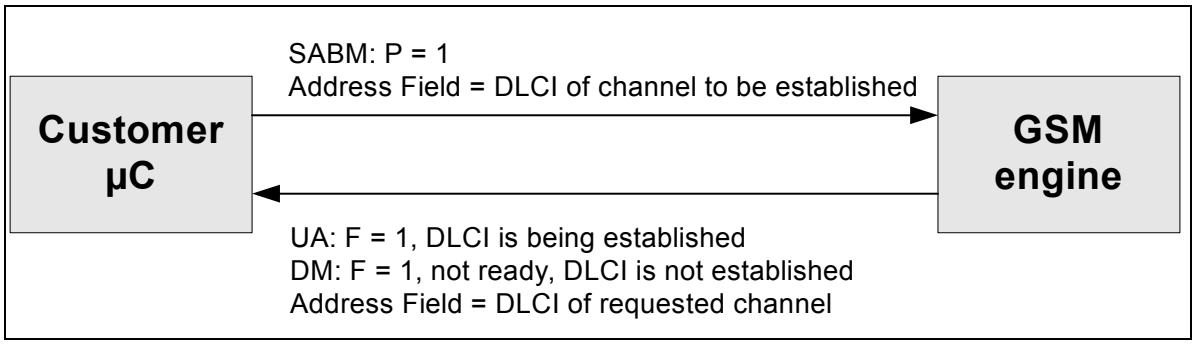


Figure 7: DLC establishment

5.1.3 Information transfer

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

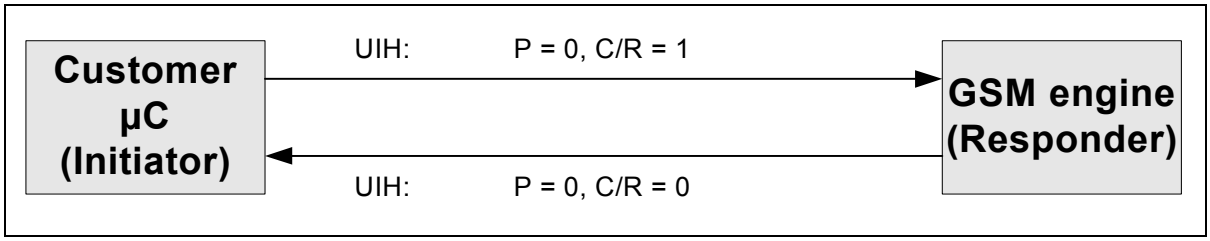


Figure 8: Information transfer

5.1.4 DLC release

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

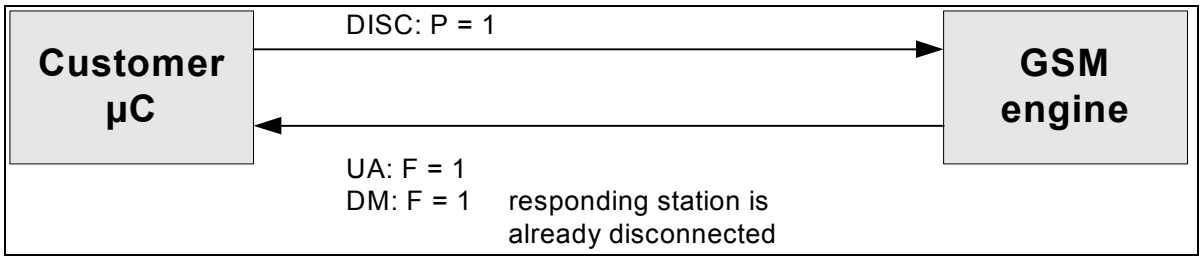


Figure 9: DLC release

5.1.5 Turn off procedure

To switch the multiplexer off the following procedures may be used:

- Disconnecting all DLCIs (DLCI command): The last disconnection is for DLCI = 0
- Multiplexer close down command (CLD): The multiplexer controls the close down; all DLCIs are closed first and finally DLCI = 0

From this moment on, both stations returning to AT command mode.

5.1.6 Multiplexer control channel

DLCI = 0

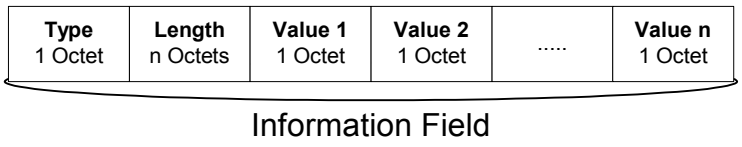


Figure 10: Multiplexer control channel

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.

5.1.6.1 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

5.1.6.2 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 6.

5.1.6.3 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 V24 status lines as described in this chapter are included in multiplexer protocol version 3 only. However, if you wish to use the advantages of this version it is absolutely necessary to switch on the version 3, otherwise version 1 will be used, see Chapter 6.2.

Direction host application→ module (for request only) MUX V3:
RTC: mapped to DTR
RTR: mapped to RTS
Bit 5, 6, 7, 8 are not valid.

Direction module → host application (for request only) MUX 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 for version 3 and an MSC request only. Descriptions of all other versions are available in Chapter 6.
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.
Remember that the bits described above are valid in Mux version 3 only, switched on by a version control handshake (see Chapter 6). More detailed information on older multiplexer versions are available in Chapter 6.2.

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.

5.1.6.4 Power saving control (PSC)

The power saving control message use 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

Figure 12: Power Saving Control (PSC)

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

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.

5.1.6.5 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 a NSC-frame.

5.2 Samples of establishing logical channels

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

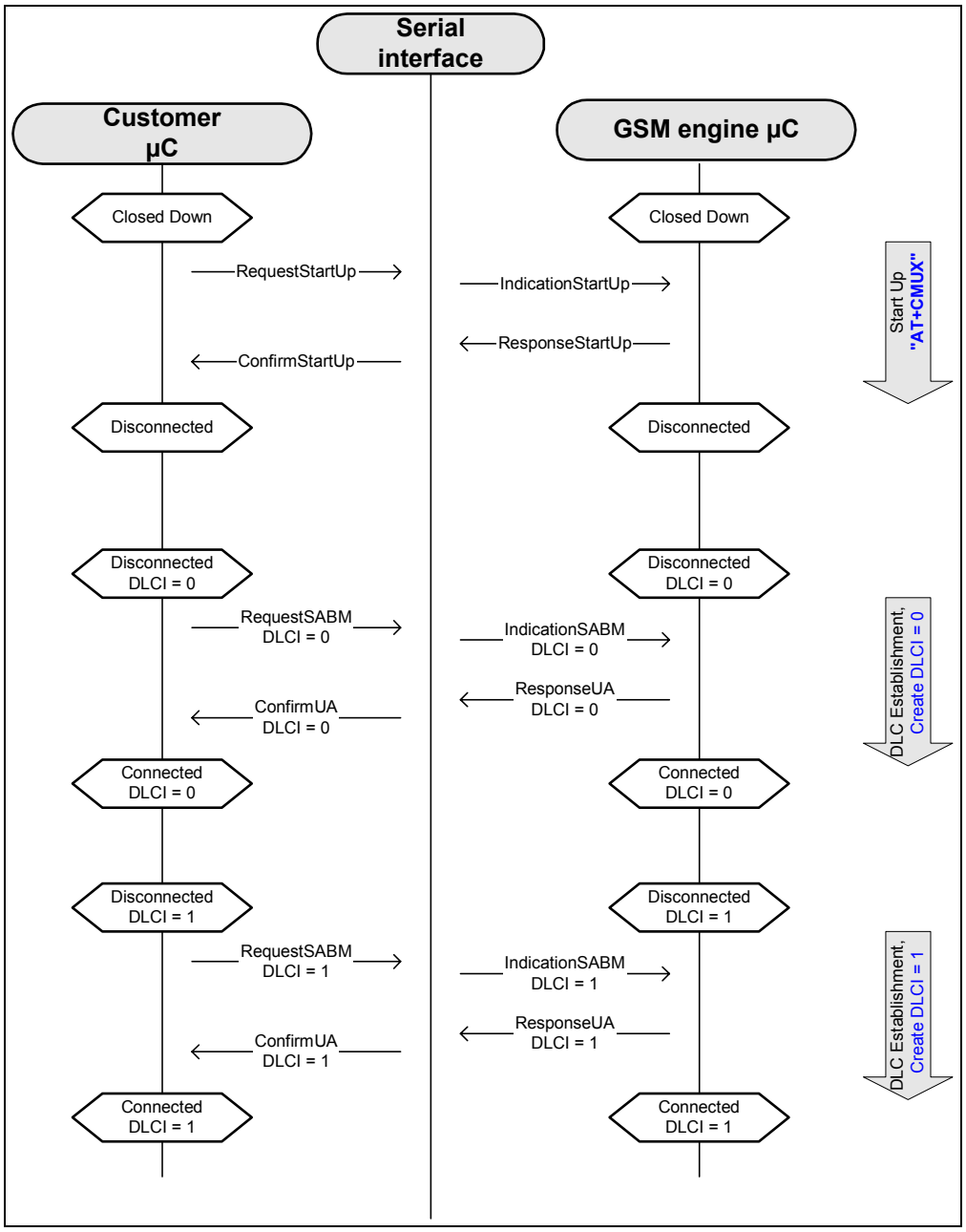


Figure 13: Establishing the multiplexer control channel and the logical channel

5.2.2 Closing the multiplexer protocol and returning to AT mode

The best and safest approach to power down the multiplexer is to issue to appropriate AT command for details see [2]. This procedure sends a CLOSE DOWN request of the multiplexer control channel and allows the interface to return into AT command mode.

6 Version control

6.1 Introduction

The multiplexer protocol offers a version control which ensures, that TE and ME side supporting the same state of functionality.
If you wish to implement the multiplexer protocol yourself, please keep in mind that it is absolutely necessary to implement the version check in the case, you wish to use features included in later versions. The implementation is a subset of the GSM 07.10 standards.
The version control also ensures the upward and downward compatibility of the GSM engine. When the multiplexer is started, the MS and the application negotiate which MP version to use.

6.2 Multiplexer protocol versions

This chapter briefly summarizes the various existing multiplexer protocol versions.

1. No version check
   - No break signal is sent

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

3. Advanced version included in all modules mentioned in 1.1
   - All features from 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

The table below summarizes the differences in handling the MSC using the various multiplexer protocol versions:

Modem status command (MSC)

Command 1 octet	**Length** 1 octet	**DLCI** 1 octet	**V.24 signals** 1 octet	**Break Signals (optional)** 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 14: MSC as used in version 3

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

Table 8: Version differences for MSC

Version no	RTC RTC host application - module	RTC RTC RING DCD module - host application
1	1 1 if 0 is indicated, all calls are terminated	not used
2	DTR RTS	not used
3	DTR RTS	DSR CTS RING DCD

Please note that it is strongly recommended to use the latest version, switched on by version control.

Examples for handling different versions from the mux:

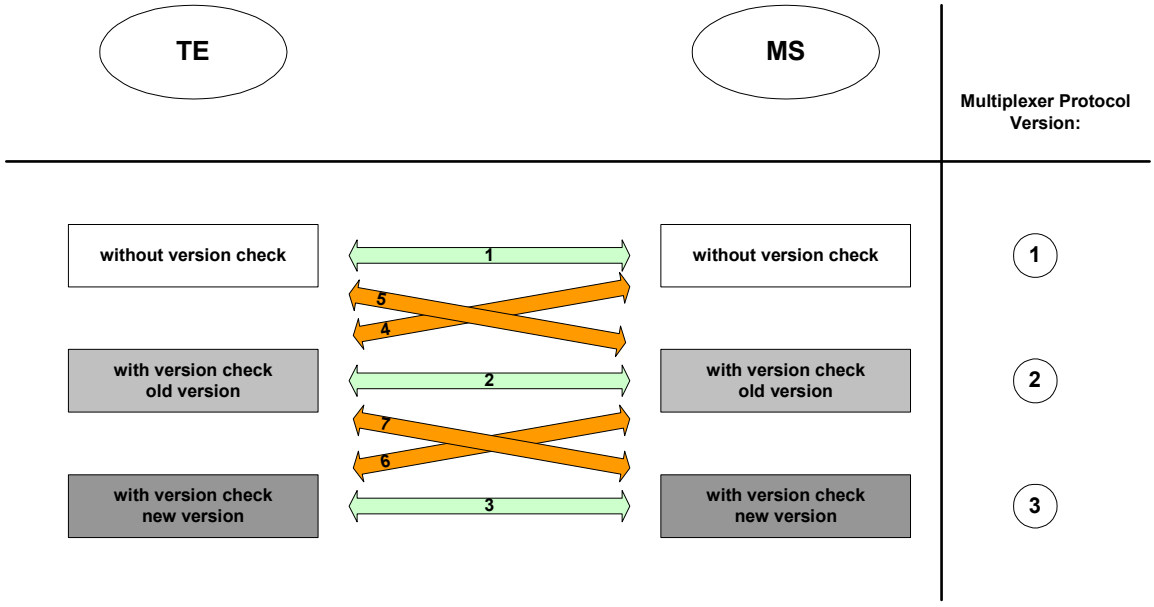


Figure 15: Overview of combinations for version check

The TE and MS multiplexer version numbers can be traced on the serial interface. They appear as follows:

- TE version (e.g. version 1): `TEMUXVERSION0001`
- MS version (e.g. version 2): `MSMUXVERSION0002`

As already mentioned before during the initialization process the valid multiplexer version is negotiated. If TE and MS do not include the same version, the lower version will be used.

Example 1, 2 and 3 (see Figure 15)
This example describes the case if TE and MS support the same multiplexer version without version check.

Example 4 and 6
Vice versa if the MS does not include version check the TE uses the older protocol version. This can be done in various ways:

- (WinMux2k): DLLs for all versions are available and have to be chosen according to the negotiated version.

- Customer application with Siemens AG sources:
  The TE switches back to the older version without version check. Since all new protocol features can be used only if they are implemented on the MS side the switch back happens automatically. Please ensure that new multiplexer protocol features provided by Siemens AG should be implemented into the MS side of the customer application.

- Customer application with proprietary multiplexer implementation:
  See Customer application with Siemens AG sources.

Example 5 and 7
In this case TE and MS do not support the same multiplexer protocol version. On TE side a version check is not included while the MS version already includes it. Due to the implemented features the older protocol version will be used automatically.

6.3 Implementation

The TE initiates the version check by sending the test command message on 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 should reply the test command response with the same contents for the verification pattern. Hereafter the MS must send a test command message (with MSMUX_Version) to the TE which should be responded with the same contents. After sending the response a comparison will be performed.
Please note that both sides should use the same multiplexer protocol version

6.3.1 Troubleshooting

In the case the MS realizes the implemented software but the TE does not response correctly, the following errors might occur:

- ´Request Test Message´ is not sent from TE side:

  No Version check takes place. No retransmission for ´Request Test Message´ is triggered. The multiplexer is started with the lower version.

- ´Response Test Message´ is not sent from TE side:

  No timer has been implemented for the non responding cases. If the response message is not received as expected, the multiplexer stays in the state DLC_CONNECTEDWAIT4RESPONSE until another multiplexing related action takes place.

However it is possible to send test commands with “any contents” (with exception of 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.

6.3.2 Coding of “TestCommand” message

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 9.
(see GSM recommendation 07.10 / Chapter 5.4.6.3.4)

Table 9: 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. So it is possible 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.

6.3.3 Example of “TestCommand” message

An example for coding a “TestCommand” message is illustrated in Table 10.

Table 10: Coding of “TestCommand” (Example)

IEI coding								Information element name
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	0x	
1	0	0	1	1	1	1	1	F9	START Flag
0	0	0	0	0	0	1	1	03	Address Field DLCI=0,C/R=0,EA=0
1	1	1	0	1	1	1	1	EF	Control Field UIH Frame, P/F=0
0	0	0	0	1	0	1	1	25	Length LENGTH=18, EA=1
0	0	1	0	0	0	1	1	23	Type Field **TestCommand**, C/R=1, EA=1
0	0	0	0	0	1	1	1	21	Length Lenth=16, EA=1
0	0	0	0	0	1	0	0	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	
0	0	0	0	0	0	0	1	39	
X	X	X	X	X	X	X	X	XX	FCS (is calculated)
1	0	0	1	1	1	1	1	F9	END Flag