Protocol manual

PROFIBUS-FMS

Hilscher Gesellschaft für Systemautomation mbH
Rheinstrasse 78
D-65795 Hattersheim
Germany

Tel. +49 (0) 6190/9907-0
Fax. +49 (0) 6190/9907-50

Sales: +49 (0) 6190/9907-0
Hotline and Support: +49 (0) 6190/9907-99

E-Mail: hilscher@hilscher.com
Web: http://www.hilscher.com

Index	Date	Version	Chapter	Revision
4	19.02.96	V02.100	all	created
5	02.10.98	V3.000	all	PROFIBUS-FMS based on ASIC ASPC2

1 General

This manual describes the PROFIBUS-FMS and PROFIBUS-FDL. It presents important information about PROFIBUS functions, which you have to know if you want to use PROFIBUS.

The manual starts with **specification and protocol fundamentals**.

Special attention should be paid to the chapter PROFIBUS **data transfer**.

The next step is the configuration of the PROFIBUS device. Please read the chapter on **configuration**.

If you use the device in conjunction with an existing user program/driver, please read the manual on commissioning of that program after configuration.

The error handling and diagnostic are described in chapter **error numbers** and **diagnostic**.

2 Spezification and protocol fundamentals

The implementation is according to EN 50 170 part 2.

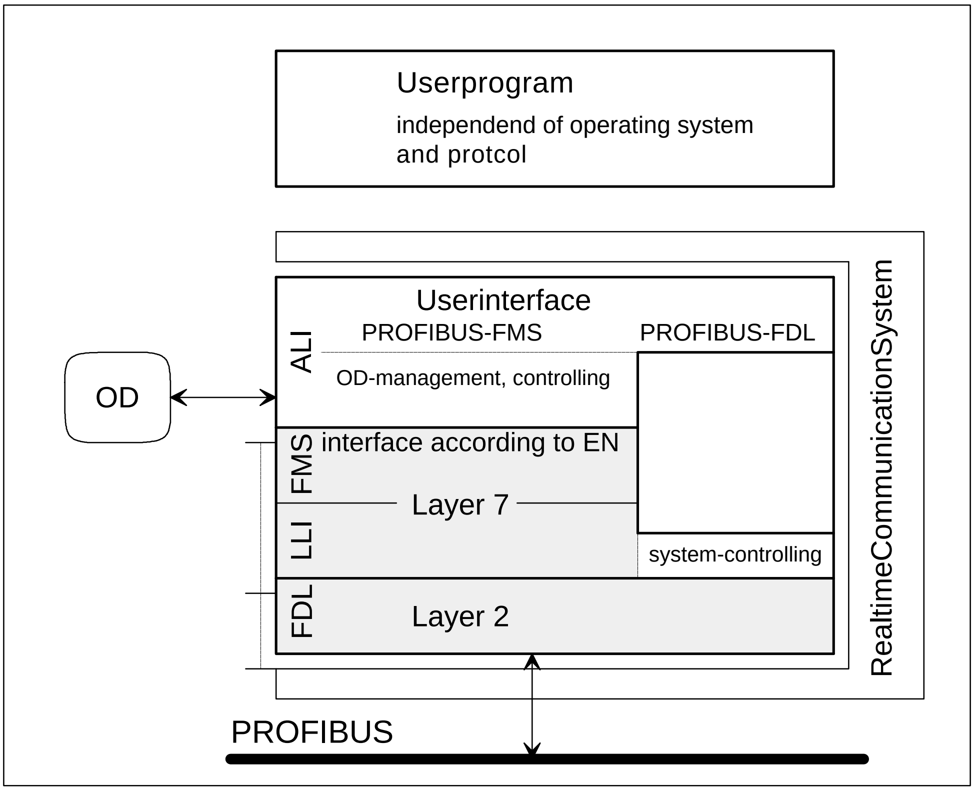
PROFIBUS is the abbreviation of PROcess FIeld BUS.

2.1 PROFIBUS-FMS

2.1.1 Structure

The function of data transfer is divided into individual layers with the PROFIBUS. Each layer has a function defined in EN 50 170 part 2. In our PROFIBUS implementation, each layer is implemented by a separate task, coordinated by a real-time operating system. The PROFIBUS interface proper to EN has been used to integrate the PROFIBUS application layer, the application layer interface (ALI), in the software The ALI manages the object directory and images the PROFIBUS EN functionality on a protocol independent user interface. The ALI interacts with the PROFIBUS Field bus Message Specification (FMS) interface to EN, or alternatively with the Field Data Link (FDL).

The illustration below shows the relationships between the individual layers (tasks). The PROFIBUS section to EN 50 170 is shown in grey.



Structure of the PROFIBUS software, links between the layers

Field bus Data Link Layer (FDL)

The function of the FDL is that of transferring the data from one device to another. This comprises the addressing and synchronisation of the stations on the bus (token passing). The FDL layer does not define any data contents or protocol sequences.

Lower Layer Interface (LLI)

The LLI regulates the sequence or data transfer. Among other functions, it decides how the software has to react in the case of a communication error or time-out. Furthermore, the LLI is the interface between the logical (communications reference) and physical addressing systems (station address, service access point).

Fieldbus Message Specification (FMS)

The FMS layer is the PROFIBUS application layer to EN 50 170 Part 2. It includes the user data in a PROFIBUS compatible message format and makes the services accessible to the user available.

The FMS also defines the format of the user data. A uniform data format is necessary for the communicating stations to be able to interpret the data they exchange. PROFIBUS FMS, for example, specifies the word format as high-byte/low-byte, whereas data words in a PC are stored as low-byte/high-byte (Intel format). Further information can be found in the chapter data format.

Application Layer Interface (ALI)

The ALI no longer belongs to the PROFIBUS layer to EN standard. The ALI itself is the application layer of the PROFIBUS. It interacts with the service made available by the PROFIBUS. The ALI in conjunction with the user program organises the PROFIBUS communications data, objects and functions.

The ALI is already integrated in our devices, in order to implement as simple a user interface as possible.

2.1.2 Scope of Functions of the PROFIBUS-FMS/FDL Implementation

In the PROFIBUS standard, the functions of the PROFIBUS are divided into those which every PROFIBUS station must be able to perform and those which may be available if so required of the device. As the scope of these functions may deviate from device to device, this chapter deals with the scope of functions of our implementation.

2.1.2.1 Scope of Functions of Layer 2-FDL

Scope of functions in accordance with EN 50 170 Part 2

- active station
- data rates of 9.6, 19.2, 93.75, 187.5, 500 kBit/s, 1.5, 3, 6 and 12 MBit/s.
- net data length up to 246 Bytes
- SDA, SRD, CSRD and SDN services
- management services

All transmission services and an adapted selection of the management services are supported. The FDL communications services are not directly available to the user.

2.1.2.2 Scope of Functions of Layer 7-LLI

The number of configurable communications references is 32. A maximum of 4 parallel services per communications reference is permissible.
The LLI provides the functions for

- acyclical master - master,
- acyclical master - slave,
- cyclical master - slave,
- Broadcast and
- Multicast

communications.

2.1.2.3 Scope of Functions of Layer 7-FMS

All the FMS services described as "mandatory" in EN 50 170 are currently available, supplemented by READ, WRITE, EVENT NOTIFICATION and INFORMATION REPORT. As a result of the modular software structure, further services can be added within a short time.

Available services:

- Initiate - Establishment of a connection
- Read - Reading of an object
- Write - Writing of an object
- Event Notification - Notification of an event
- Information Report - Unconfirmed write
- Abort - Shutdown of the connection
- Reject - Rejection of a service
- Status - Device status (as server only)
- Identify - Device identification (as server only)
- Get-OV - Reading of an OV entry (as server only)

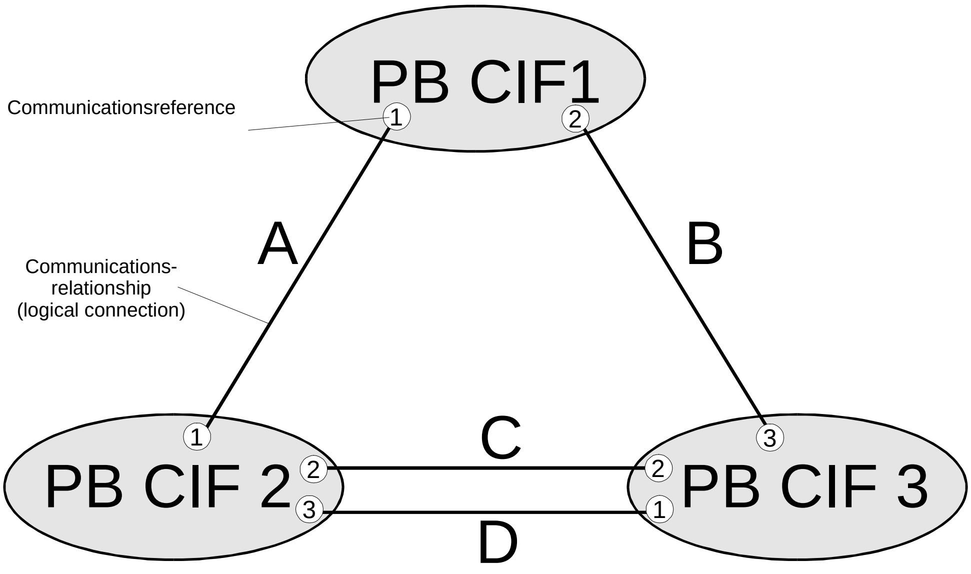
3 PROFIBUS Data Transfer

This chapter has general information about the data transfer on the PROFIBUS.

3.1 PROFIBUS-FMS

The devices on the PROFIBUS network are connected **physically** together. Only devices between which there is a **logical** connection can communicate together. We speak of communications relationships.

There can be several communications relationships between two PROFIBUS devices. Within a PROFIBUS device, a communications relationship is addressed by means of a unique reference number. This reference number is known as the **communications reference (CR)**.



Logical connections, communications references

Several jobs can be executed at the same time through each logical channel. Assignment of the returned responses to the transmitted jobs is effected by the application program and makes use of the job identifier, the so-called 'invoke ID'!

A communications channel can have three states:

- closed - inactive = basic state
- initiating - temporary, after sending the first job only
- open - data transfer

Before a message is sent, the PROFIBUS software checks the status of the connection. If necessary, the connection is established. The connection remains in the open state until the user program closes it again by sending an abort request. A connection can also be closed down independently by the PROFIBUS software in response to a communication error or the expiry of a watchdog timer.

Each time a job is sent in a logical channel, the PROFIBUS software increments a counter internally. With each response received on that channel, the counter is again decremented. If the counter reaches the value for the maximum permissible number of parallel services in the communications relationship, all following send jobs are acknowledged by the PROFIBUS software with an error message to the user program. The counter can be set back to zero by sending an abort request, i.e. resetting the link. All open jobs are thus discarded.

If the application program sends a job and this is not acknowledged by the communicating station, an open job is registered by the user program. The link therefore has to be reset by sending an abort request before the next job is transmitted. This job monitoring function can optionally be performed by the ALI. Please consult the chapter on configuration of the CRL for further details.

3.1.1 The Automatic Initiate

If required you can configure an automatic initiate. During defined periods an initiate telegram will be sent trough all configured, closed CRs, irrespective of sending data trough this CRs.

The field 'Task 1 State' in the dual port memory offers a row of bits, that show the actual state of the CRs. To every closed CR the value 0 is assigned, every open CR has the value 1.

The first bit in this state shows the actual state of the VFD. If this bit has value 1, the state 'OPERATIONAL' is active, so the PROFIBUS is initialized and taken in logical token ring - ready to transfer data. The state of the CR 1 is displaied in bit 1, the state of CR 2 in bit 2 etc.

Task 1 parameter

0 - VFD-state
1 - CR1 open (data transfer possible)
0 - CR2 closed (data transfer not possible)
1 - CR3 open
1 - CR4 open
0 - CR5 closed
1 - CR6 open
0 - CR7 closed
0 - CR8 closed
... - . . .

*Example of the bits in '**Task 1 State**'*

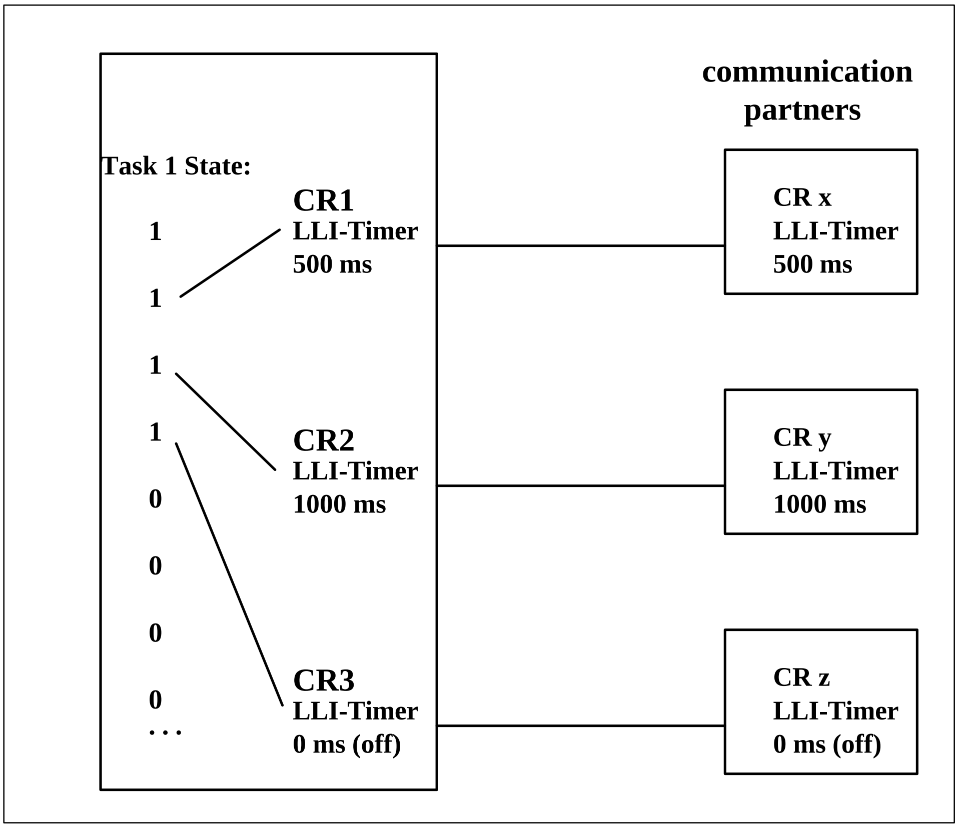
The state of the CR will stay open until an abort telegram is sent or received. The communication partner or the user of the device can send this abort or the ALI (application layer interface) sends the abort independent, when none or a not plausible response has been received after sending a request. If this happens, mostly the communicationpartner has been switched off.

You can activate the automatic initiate by configurating the parameters 'Auto Init Time' and 'Auto Init DTime' in the table ALI_SETUP. Please read the corresponding capture of configurating.

3.1.2 Connection Control

On PROFIBUS the communicationpartners can check their presence at the bus online by checking the time passing between receiving two telegrams of the partner. If the partner station has no data to send during this time, an IDLE-telegram will be sent.

This time interval can be configured for each CR with the parameter LLI-timer in the table KBL. Other manufacturer also call this parameter 'Control Interval' or 'Connection Interval'. Please read in the configuration chapter the part CRL.



Configuration of the LLI-timer and the display of the CRs on dual-port memory

3.1.3 Client and Server Behaviour

PROFIBUS communications identify the communications partners as clients and servers. The client (requester) sends a job to a server (responder), which has to execute that job. If, for example, it is a read job, the client issues the read job to the server, which must then return the requested data to the client in a response message.

In each PROFIBUS-FMS device, objects are defined which are available for access by other stations. A client can therefore only access those objects at the server which are defined in the server object directory.

When a job arrives, therefore, the ALI checks whether the object is defined in the object directory. If so, the request is passed on to the application (indication). If the object is not defined, the ALI rejects the request with the error OBJECT_NON_EXISTEND (Code=7) in error class ACCESS (Code=6). The server application notices none of this.

The following checks are performed by the ALI prior to signalling and indication to the application:

Check	Errorclasse	Errorcode
Object in ST_OV_0 defined ?	ACCESS	OBJECT_NON_EXISTEND
Subindex value ok?	ACCESS	OBJECT_ATTRIBUTE_INCONSISTEND
Length of receiving write-data is according to length of defined object?	ACCESS	TYPE_CONFLICT
Error in server-application MSG->F <> 0	ACCESS	OTHER

The error codes or error classes have the following values as defined in the standard:

```
ACCESS                             6
OTHER                              0
OBJECT_ATTRIBUTE_INCONSISTEND      5
OBJECT_NON_EXISTEND                7
TYPE_CONFLICT                      8
```

Further information can be found in the chapter error numbers.

3.1.4 The PROFIBUS-FMS Data Format

The data format of the PROFIBUS-FMS data is described in the PROFIBUS standard. A uniform data format is a precondition for all communications devices to be able to interpret the data exchanged. The norm defines on PROFIBUS-FMS the motorola-format (MSB-LSB). For our devices you can configure the data format of the user-program as intel (LSB-MSB) or motorola (MSB-LSB). The converting of data will be made on the device.

By using the from us defined **FDL-defined** communication, the partner-station has to send the data in the on PROFIBUS usual motorola-format (MSB-LSB). So the user program will receive the data in the expected format.

3.1.5 The PROFIBUS-FMS Data Types

The PROFIBUS data types supported are listed in the following table:

Type index	Index code	Data type	Range of data type	**Type length** Length of one element in byte
Boolean	1	Boolean	TRUE(=255), FALSE(0)	1
Int8	2	Integer 8	-128..127	1
Int16	3	Integer 16	-32768..32767	2
Int32	4	Integer 32	-2^31-1 .. 2^31	4
UInt8	5	Unsigned Integer 8	0..255	1
UInt16	6	Unsigned Integer 16	0..65535	2
UInt32	7	Unsigned Integer 32	0..2^32	4
Float	8	Float	IEEE-Float	4
Octet String	10	Octet String (Byte string)	Byte string: interpretation by user	Length of byte string

The supported PROFIBUS data types

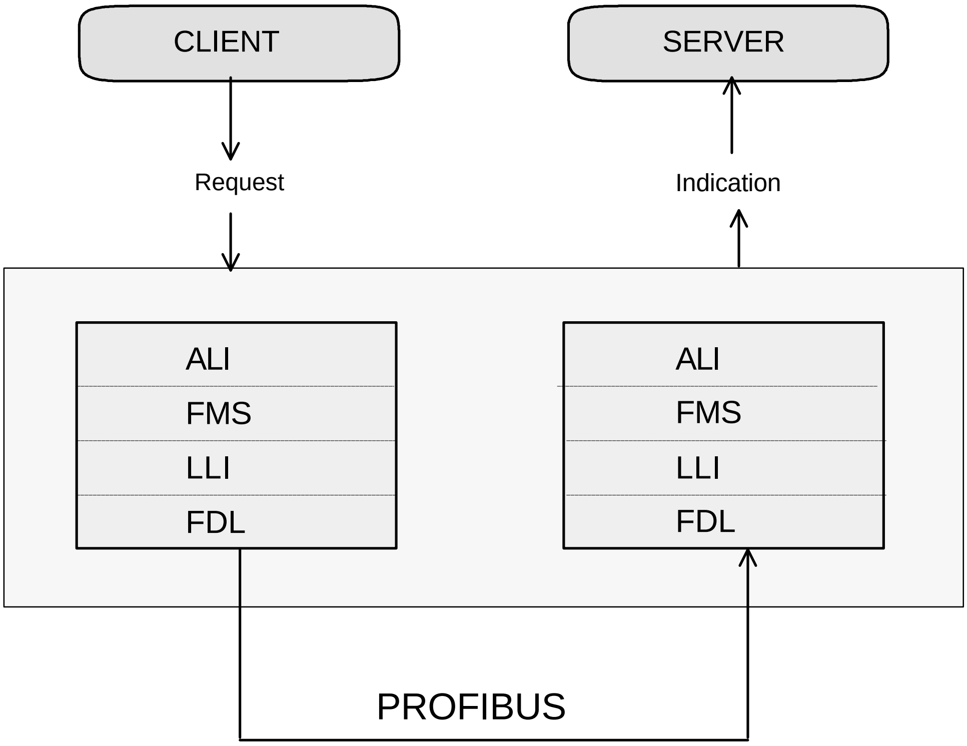
3.1.6 PROFIBUS-FMS Data Transfer

In the context of communications with PROFIBUS-FMS, two types of service are fundamentally distinguished, and these are explained below:

3.1.6.1 Unconfirmed or Unacknowledged Services

Unconfirmed services are not acknowledged via the bus on the user level. In this context, a client sends a request to a server, where the incoming message is described as an indication. On receipt of the indication, the data cycle on the user level is concluded.

The expression "Acknowledge" means that an acknowledgement takes place on the bus, but this is only used in the SPL data transfer layer and not passed on to the user. Reliable data transfer is thus ensured.



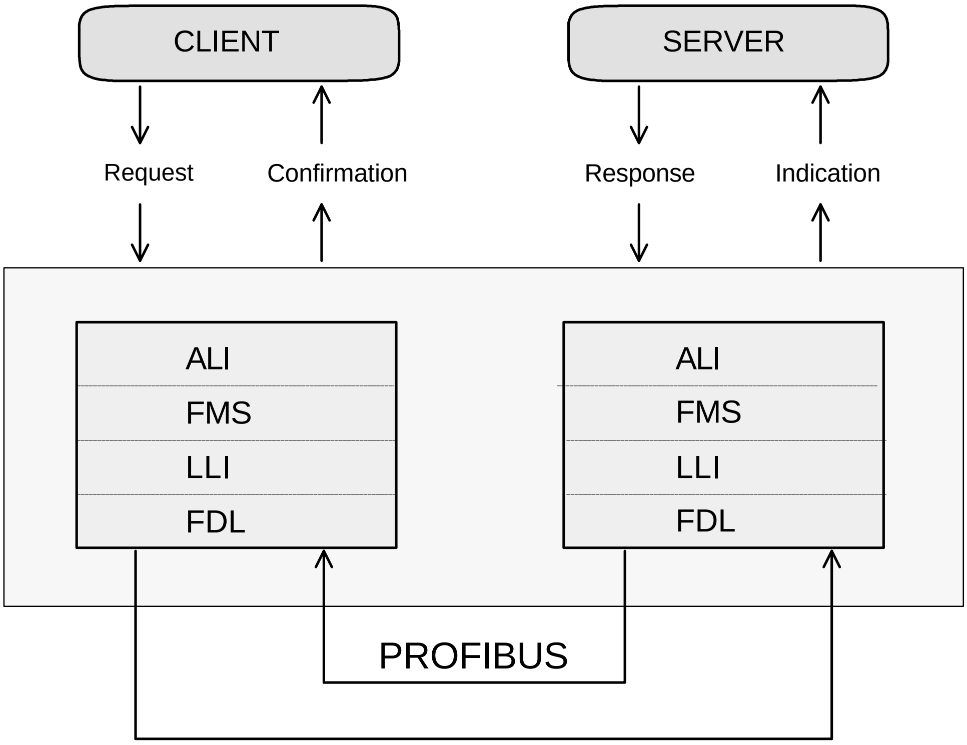
Transmission of an unconfirmed service

3.1.6.2 Confirmed Services

Confirmed services are acknowledged via the bus on the user level. In this case - as with unconfirmed services - a client sends a request to a server, at which the incoming message is described as an indication.

The server processes this request, but then compiles a response, which it sends to the client. Arriving at the client, the message is designated a confirmation. The confirmation contains, for example, the data read for read services.

Only when the confirmation has been received is the cycle of a confirmed service concluded.



Transmission of a confirmed service

Our devices support the confirmed services read and write.

3.1.7 PROFIBUS-FDL Data Transfer

In contrast to FMS data transfer, no message contents are defined in FDL data transfer. The FDL cycle describes only the transfer of data from one device to another and the acknowledge of correct receiving. This is realized with the **FDL service 'SDA'** (Send Data with Acknowledge). There are no functions like read or write, only send. Therefore our PROFIBUS-firmware offers two interfaces for FDL-communication, which can be configured for each communications reference. So it is possible to run at the same time on one device data transmission on FMS, FDL-transparent and FDL-defined. The last ones will be described next chapters.

3.1.7.1 Transparent FDL Data Transmission

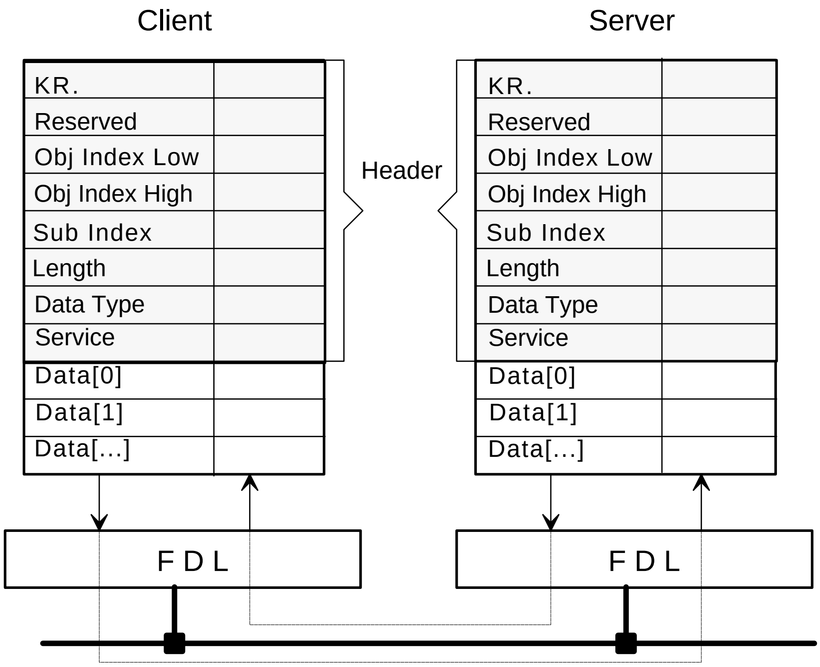
The transparent FDL data transmission sends a data-block from one PROFIBUS device to another. The contents of the data does not matter. If the partner is a PLC, mostly the data arrives in a receive data block (DB). If the device should receive data from a PLC, the PLC has to invoke the data transmission. Mostly the PLC puts the data in a datablock and sends it by using a SEND function block (SEND-FB). The device receives this data as user data.

3.1.7.2 Defined FDL Data Transmission

Because there is a difference between responding to the FMS- and FDL transparent - interfaces, we implemented the FDL-defined interface with the aim of a uniform user interface for FMS and FDL. These concern on the one hand the sequence of FDL communications and on the other hand a uniform message header data structure. The communicating station, for example a PLC S5-95U, has to construct an image of this sequence and evaluate the header data structure in the manner described.

Our FDL-defined user interface corresponds to that of the FMS. All data from the "logical address" parameter onwards are transferred to the communicating station.

The following table shows the general structure of the FDL message data. The header data must be present in every message!



Message data in FDL-defined data transfer

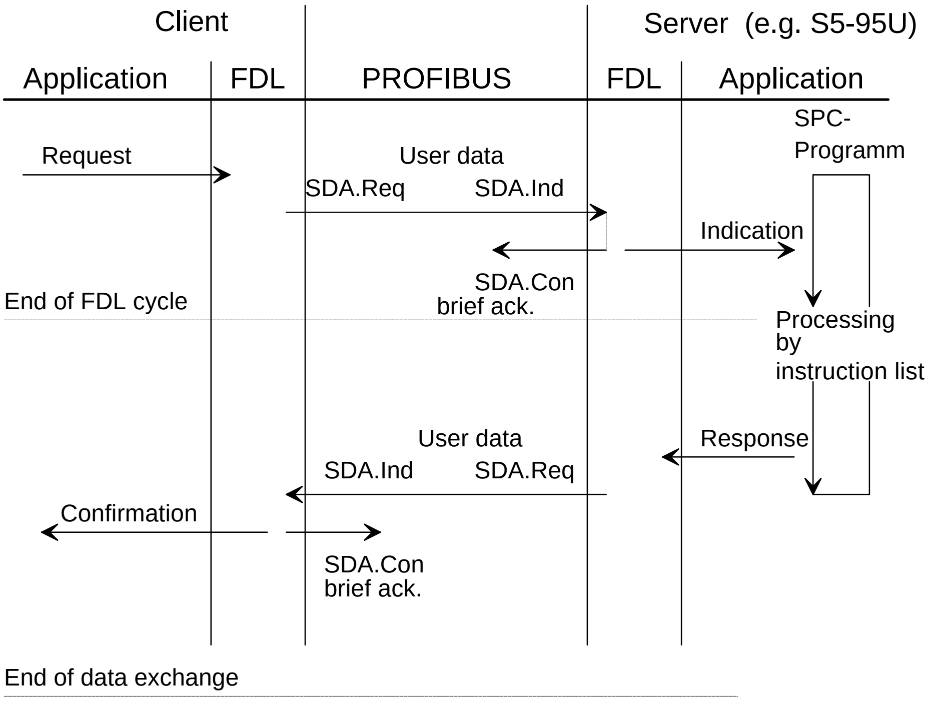
The functions of the parameters and values are shown in the following table:

Parameter	Description
KR	Communications Reference
Reserved	Reserved (0)
Obj Index Low	Data Block (DB)
Obj Index High	Reserved (0)
Sub Index	First Data Word (DW) to Read/Write
Length	Number of Data to Read/Write (Counted in Data Types)
Data Type	Data Type from PROFIBUS Convention
Service*	PROFIBUS Service: 1 - Read 2 - Write
Data[ ]	User Data

Functions of the FDL-defined data
**In the response the bit 6 (0x40) of the parameter 'service' has to be set!*

Notice: In the response the bit 6 (0x40) of the parameter 'Service' has to be set.

The time sequence for FDL data transfer looks as follows:



Time sequence of FDL data transfer

In the FDL - link between Siemens devices, an FDL send cycle is terminated on arrival of a brief acknowledgement from the remote FDL. This brief acknowledgement indicates that the data have been successfully transferred from A to B. From the point of view of the PROFIBUS-FMS/FDL user, however, the acknowledgement on the "higher" level, i.e. the confirmation, is missing. The confirmation is transmitted by the remote partner in a new FDL cycle. The message data for the confirmation/response are generated by the server from the indication received.

In order to facilitate the distinction between a request and a response, Bit 6 in the service byte for the response is set. Bit 6 is automatically operated by the ALI in our PROFIBUS software, whereas it has to be operated by the FDL application program in the PLC or other PROFIBUS-FDL partners. See chapter Example of FDL cummunications with PLC S5-95U.

3.1.7.3 Error Handling

If an error occurs in the application program, the value of the data type in the response is set to 0xff. So the userprogram will receive the error 0x81, 129 (Error in application of remote-partner).The user data then contain data of the following structure:

Parameter	Type	Description
ErrorClass	Byte	Error class
ErrorCode	Byte	Error
AdditionalDetail	Word	Additional information to this error

The contents of these parameters should be in harmony with the FMS error messages defined in the EN standard. See the chapters below.

3.1.7.4 Example of FDL Communications with PLC S5-95U

The FDL server must be able to analyze and process the incoming message data. In connection with an PLC S5-95U, we propose the following interpretation of the message header:

Parameter	Interpretation in S5-95U
KR	Unused
Reserved	Unused
Object Index, Low Byte	Data Block DB
Object Index, High Byte	0
Sub Index	Data Word DW
Data Length	1..233 (= max.241 - 8 byte header) Counted in data types
Data Type	1 .. 8, 10
Service	Read / Write
Data[]	

The directory "STEP5" on your diskette contains the example programs for complete frame for layer 2 communications with an PLC S5-95U as server.

For actual filenames and functions please read the README.ASC - file.

4 Configuration

4.1 Parameter

4.1.1 Station Address

Each device on the PROFIBUS can be addressed by its unique **station address** (= device address) (cf. 'Highest station address'). The region of the station address is 0 to highest station address. The highest station address is in the region 2 to 126.

4.1.2 The Bus Parameters

In accordance with the PROFIBUS standard, all bus times are given as bit times! Conversion from milliseconds to bit times can be effected with the following formula:

*Bits [bit time] = Time [milliseconds] * Baud rate*

Example: 5ms * 187,5kB => Bit time = 937.5 => 938

The following table includes the parameters for a FMS master system with two stations.

Parameter		Baud rate	12,0 MB	6,0 MB	3,0 MB	1,5 MB	500kB	187,5kB	93,75kB	19,2kB	9,6kB
Min. Station Delay Responder	min T_{SDR}	t_{BIT}	11	11	11	11	11	11	11	11	11
Max. Station Delay Responder	max T_{SDR}	t_{BIT}	800	450	250	150	100	60	60	60	60
Slot Time	T_{SL}	t_{BIT}	1.000	600	400	300	200	100	100	100	100
Quiet Time	T_{QUI}	t_{BIT}	9	6	3	0	0	0	0	0	0
Setup Time	T_{SET}	t_{BIT}	16	8	4	1	1	1	1	1	1
Target Rotation Time	T_{TR}	t_{BIT}	8.192	8.192	8.192	6.656	6.656	6.656	6.656	6.656	6.656
GAP Actualization Factor	G		10	10	10	10	10	10	10	10	10
Max Retry Limit			4	3	2	1	1	1	1	1	1
Highest Station Address	HSA		2	2	2	2	2	2	2	2	2

Parameters has to be changed if devices are in the network that need other parameters, for example slower devices. The Tartget Rotaion Time has to be changed to higher values if there are more than two masters.

The bus parameters must be configured identically at all devices!

- **Baud rate**

The baud rate can be set in the following stages:

Baud rate	Bit time	Max. cable length (type A)
9,6 kB	104,2 usec	1200 m
19,2 kB	52,1 usec	1200 m
93,75 kB	10,7 usec	1200 m
187,5 kB	5,3 usec	1000 m
500 kB	2 usec	400 m
1,5 MB	666,7 nsec	200 m
3 MB	333,3 nsec	100 m
6 MB	166,7 nsec	100 m
12 MB	83,3 nsec	100 m

Please note that the maximum cable length depends on the baud rate.

- **Minimum Station Delay of Responders - min T_{SDR}**

The Minimum Station Delay Time is the minimum period the responder (slave) has to wait before it is allowed to send its reply or acknowledgement.

The Minimum Station Delay Time must be longer than the Ready Time T_{RDY} *of the initiator. This is the time within a master station is ready to receive an acknowledgement or response from the responder (slave). This time is defined to be less or equal eleven bit times for a DP master.*

- **Maximum Station Delay of Responders - max T_{SDR}**

The responder (slave) has to answer before the Maximum Station Delay Time has passed. The initiator (master) has to wait at least this time after an unconfirmed telegram before sending a new telegram.

The Maximum Station Delay Time must be longer or equal than the maximum of all Station Delay Times of all responders.

- **Slot Time - T_{SL}**

The Slot Time is the maximum time the initiator waits for the first character of the receipt or reply from the remote station (responder) after sending a request message or transmitting a token frame. After that period, the request may be repeated before an error is signalled (cf. 'Number of repeats').

The Slot Time must be greater than the maximum of the T_{SDR} *of all responders. Because the Slot Time is only supervisor time it decrease not the performance if it is set to a large value.*

- **Quiet Time - T_{QUI}**

Transmitter fall time and/or repeater switch time. The time a transmitting station must wait after the end of a frame before enabling its line receiver.

The Quit Time must be smaller than the Minimum Station Delay Time.

- **Setup Time - T_{SET}**

The time which expires from the occurrence of an event e.g. last data send until the necessary reaction is performed e.g. to start the Syn Time.

The Setup Up Time T_{SET} is normally used to prolong the time after the initiator received the last character from the responder until it sends the next request telegram. This time is named Idle Time T_{ID} which is not direct accessible.
Depending on the type of the telegram there are two different Idle Times:

T_{ID1} starts after the initiator had received an acknowledge, response or token telegram.
The Idle Time 1 is calculated with the following equation:

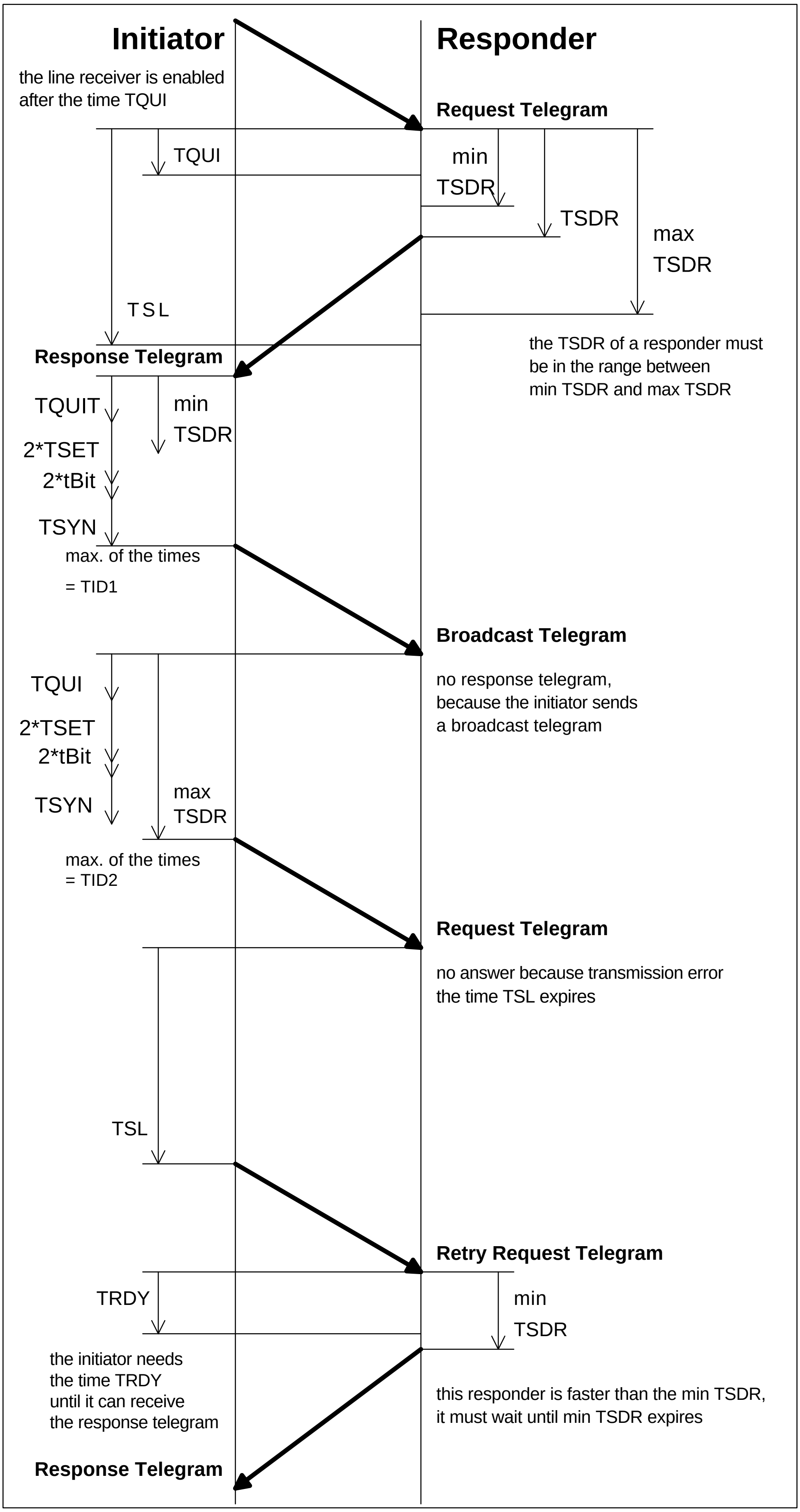
T_{ID1} = maximum (T_{QUI} + 2*T_{SET} + 2 + T_{SYN}, min T_{SDR})

T_{ID2} starts after the initiator had send a telegram which is not to be acknowledged.
The Idle Time 2 is calculated with the following equation:

T_{ID2} = maximum (T_{QUI} + 2*T_{SET} + 2 + T_{SYN}, max T_{SDR})

T_{SYN} is defined to 33 bit times.

This assumes that the Initiator is fast enough to realise these times, which is the case with the ASIC ASPC2.


Initiator
Responder
the line receiver is enabled after the time TQUI
Request Telegram
TQUI
min TSDR
TSDR
max TSDR
TSL
Response Telegram
the TSDR of a responder must be in the range between min TSDR and max TSDR
TQUIT
min TSDR
2*TSET
2*tBit
TSYN
max. of the times = TID1
Broadcast Telegram
no response telegram, because the initiator sends a broadcast telegram
TQUI
2*TSET
2*tBit
TSYN
max TSDR
max. of the times = TID2
Request Telegram
no answer because transmission error the time TSL expires
TSL
Retry Request Telegram
TRDY
min TSDR
the initiator needs the time TRDY until it can receive the response telegram
this responder is faster than the min TSDR, it must wait until min TSDR expires
Response Telegram

- **Target Rotation Time - T_{TR}**

Within the Target Rotation Time T_{TR} the token must have moved completely round the logical ring of active stations once. For passive stations which are all DP and FMS slaves this value is irrelevant.

This time must be calculated that every active station on the bus has enough time to send its telegrams.
A DP master has to have a Target Rotation Time large enough, so that it can poll each associated DP slave once per token cycle.
In a multimaster system the cycle time of each DP master has to be extended by the token hold time of all other masters.

- **Gap update factor - G**

The GAP update factor multiplied by the Token Rotation Time determines the time for which an **active** station checks whether any new devices have come onto the bus. If a very short period is selected, GAP tests are performed frequently, which reduces the net data transfer rate on the bus. New masters on the bus are then however included in the token ring more rapidly.

- **Max Retry Limits**

This factor indicates how often layer 2 has to repeat a request telegram when it does not receive any response message from a known station within the Slot Time.

- **Highest Station Address - HSA**

The Highest Station Address for an active station on the bus. Active devices on the bus are only recognised when their station address is lower or equal to the Highest Station Address on the bus. This value must be configured the same on all masters.

4.2 PROFIBUS-FMS

This chapter provides a general description of the PROFIBUS-FMS configuration of our devices. Various configuration parameters can be adapted for specific projects or specific devices. It is therefore possible for certain parameters not to be available on your device.

The data of the configuration is kept in a database. This database can be edited with the configurating and diagnosis tool system configurator SyConPB.

The following pages describe the PROFIBUS-FMS configuration which can be edited by the user.

4.2.1 FMS Settings

- **Data Format**

The **data format** of the user data can be configurated with this parameter LSB-MSB format or MSB-LSB format.

- **Operating Mode**

The setting for **Compatible mode** is Standard.

- **Wait until Conclusion in Token Ring**

After a warm start, a number of applications encounter problems in that the PROFIBUS software accepts send jobs from the user, but does not transmit them on the bus because the FDL is not yet in the token ring. The first send job remains in the FDL until it can be transmitted!

From PROFIBUS version 1.2x onwards, the time for inclusion in the ring has increased (it is also dependent on the configuration). For this reason, time outs in the application may occur, because the first acknowledgement takes an unusually long time to arrive.

For this reason, the FDL signals inclusion in the token ring to the ALI. The ALI enters this in the VFD status.

If the parameter **Wait until inclusion in token ring** = checked, the ALI uses the VFD status to reject all send jobs from the user with an error message as long as the FDL is not in the token ring.

- **Access / Protection**

For access and protoection the **password** and the **access groups** is set here.

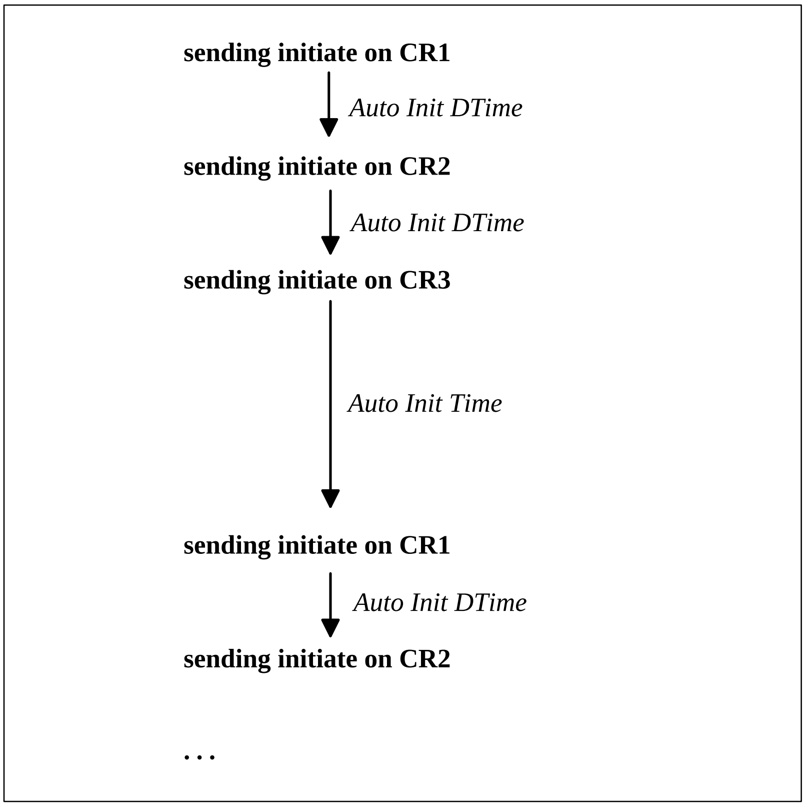
- **Automatic CR initiating**

Time interval of the automatic CR initiating is 10ms.

The parameter **Auto Init Time** and **Auto Init Difference Time** configure the timer for the auto initiate function. The value zero indicates, that the auto initiate function is switched off.

The parameter Auto Init Difference Time stipulates the time passing between sending two initiates. The application layer interface (ALI) will start with the first CR and go on in an increasing sequence. Wise values for this parameter are 2 secconds or more.

After sending the initiate on all closed, configured CRs the ALI waits for a time, configurated in the parameter 'Auto Init Time'. After that an initiate will be sent to the first, closed CR. Wise values are 10 secconds or more, values lower than 2 secconds should not be selected.



timer controlled auto-initiate

4.2.2 The communications Reference List (CRL)

To exchange data between two devices a communication relation has to be defined. A communication relation specifies how these devices communicate to each other.

The communications reference list (CRL in brief) contains the descriptions of all possible communications relations for a device independently of the time of use. Each communications relation of a device is stored as an entry in the CRL. Communications relations 1 - N are numbered consecutively. The number of a communications relation is known as the communications reference - abbreviated as CR. The communications reference is used by the application program to address the communications channel.

Two PROFIBUS devices can of course maintain several communications links with each other.

In our devices, numbering of the communications references starts with one. As the **communications references** are local device parameters, this does not result in any restrictions.

In order to exchange data between two communications partners, corresponding entries must be available in the communications relationship lists of both devices, although the number of the communications relationship (CR) must not be identical in the communicating devices. A communications reference in a device is unequivocally defined by:

Chapter	Parameter
SAPs	Local SAP
	Remote SAP
	Remote Station
Type	Communication Type
	LLI-User
Counter	SCC
	RCC
	SAC
	RAC
Services	Client
	Server
Timer	LLI Timer (Control Interval)
	ALI Timer
PDU Length	PDU Length

Overview parameter of the communication reference list

4.2.2.1 SAP - Local SAP, Remote SAP, Remote Station

To reach the other station these three parameters are neccessary.

An SAP is a fictitious point (auxiliary point) used to define logical connections. A logical connection (**communications reference**) is unequivocally defined by a **local SAP**, a **target device address** (physical bus address) and an **SAP in the remote device** (remote SAP).

- **Local SAP**

Local service acces point. In master-slave connections, the master is characterized by the fact that its local SAP is identical to the poll list SAP (SAP 58). On the slave, in consequence, the poll list SAP must never appear as the local SAP.

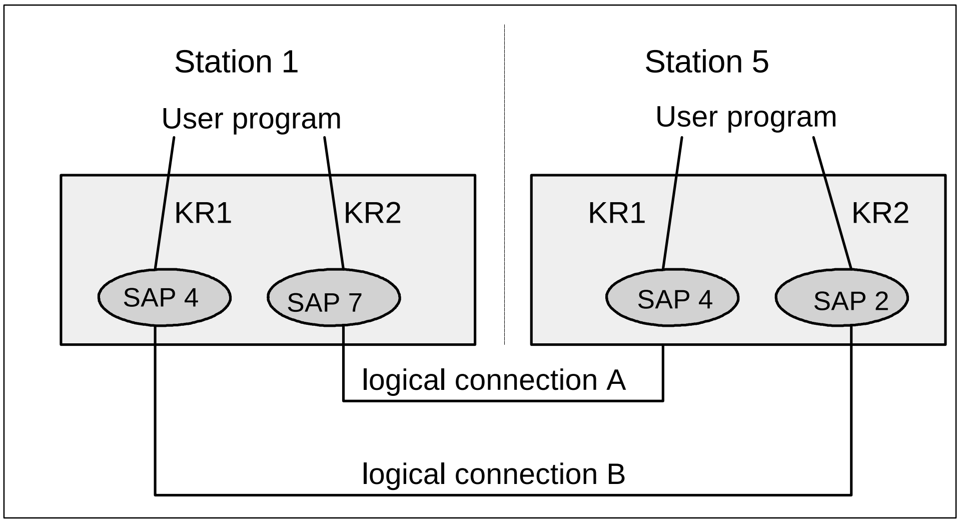
- **Remote SAP**

Service access point at the other station.

- **Remote Station Address**

This value indicates the station address of the other station. Possible Values are 0 to 125 to one station and 127 to all stations.

The following diagram should serve to illustrate the relationships:



In the constellation shown, communications references CR1 and CR2 in the devices must be defined as follows:

Station 1:

CR1: Local SAP = 4 Remote SAP = 2 Remote Address = 5

CR2: Local SAP = 7 Remote SAP = 4 Remote Address = 5

Station 5:

CR1: Local SAP = 4 Remote SAP = 7 Remote Address = 1

CR2: Local SAP = 2 Remote SAP = 4 Remote Address = 1

4.2.2.2 Type

- **Communication Type**

Communication Type	Meaning
MMAC	Master/Master Acyclic
MSAC	Master/Slave Acyclic
MSCY	Master/Slave Cyclic
MULTI	Multicast (One to Many)
BRTC	Broadcast (One to All)

- **LLI-User**

This entry sets the communication type to PROFIBUS

- FMS (Standard)
  The communication is a PROFIBUS-FMS communication. Example: Communication to Siemens S5 CP5431 (FMS).
- FDL defined (Read and Write)
  PROFIBUS-FDL communication using predefined data structures. This type of communication makes it possible to use 'read' and 'write' telegrams. This type uses requests and confirmation telegrams. Example: Communication to Siemens S5-95U. A PLC program is needed.
- FDL transparent (Send and Receive)
  PROFIBUS-FDL is used to send data from device A to device B. This type is used to send or to receice unstructured data.

4.2.2.3 Counters

For better understanding of the following parameters, please read the chapter entitled PROFIBUS-FMS data transfer.

- **Send Confirmed Request Counter SCC**,
  Receive Confirmed Request Counter RCC

Maximum number of parallel confirmed send jobs and confirmed received jobs.

The following applies:
Local SCC <= RCC at communicating device
Local RCC >= SCC at communicating device

The values must be configured in accordance with the relationship outlined above at both communications partners. The consistency is verified by the server LLI when the connection is established. If there is an inconsistency, the establishment of a link is rejected.

Confirmed services are READ and WRITE, i.e. services which expect a confirmation.

For MSZY, MULTICAST and BROADCAST, SCC and RCC must be configured with 0!

- **Send Acknowledged Request Counter SAC**,
  Receive Acknowledged Request Counter RAC

Maximum number of parallel unconfirmed (acknowledged) send jobs and unconfirmed received jobs. Acknowledge in this context means that an acknowledgement of the transmission takes place on the FDL data transfer layer, but not on the user level.

The following applies:
Local SAC <= RAC at communicating device
Local RAC >= SAC at communicating device

The values must be configured in accordance with the relationship outlined above at both communications partners. The consistency is verified by the server LLI when the connection is established. If there is an inconsistency, the establishment of a link is rejected.

Unconfirmed services are EVENT NOTIFICATION and INFORMATION REPORT.

For MSZY, MULTICAST and BROADCAST, SAC and RAC must be configured with 0!

4.2.2.4 Services

- **Client Services**

The services which can be initiated by the client on this communications reference must be configured in these parameters. For each service a bit is available that has to be set or not.

At present, the confirmed services READ and WRITE are supported.

At present, the unconfirmed services EVENT NOTIFICATION and INFORMATION REPORT are supported.

- **Server Services**

The services supported as a server on this communications reference must be configured in these parameters. For each service a bit is available that has to be set or not.

At present, the confirmed services READ and WRITE are supported.

At present, the unconfirmed services EVENT NOTIFICATION and INFORMATION REPORT are supported.

The following applies: *The server must support at least those services which can be required by the client.*

4.2.2.5 Timer

- **LLI Timer (Control Interval)**

Connection monitoring interval in 10 ms.

This timer is used to verify whether the communicating station is still ready to send or receive. The LLI timer is reset on each transmission or receipt of a message-. It can therefore only trip with open connections through which nothing is being transmitted or received. When the timer runs out, an idle message is sent, and this has to be acknowledged by the partner station. If there is no such acknowledgement, the connection is interrupted. Otherwise, the time is restarted

This timer has no effect on open connections where data are regularly transmitted. Because this is mostly the case, the connection monitoring interval is best configured to zero, i.e. not used.

The LLI timer is also referred to as the monitoring interval, connection monitor or control interval.

The LLI timer must be configured in the same manner on both communications partners. The consistency is verified by the server LLI when the connection is established. If there is inconsistency, establishment of the link is rejected.

- **ALI Timer**

Job monitoring time in 10 ms

This timer is used for time monitoring of outstanding jobs for the user program. On transmission of a job on a communications relationship, the ALI timer is reinitialised to the value set there. The timer is stopped when all the outstanding jobs have been acknowledged, i.e. there is no further outstanding job. If an acknowledgement from the opposite station is not received, an abort is triggered on the communications relationship on expiry of the timer.
This timer is a special feature of our software. It is not described in the PROFIBUS standard!

4.2.2.6 PDU Length

- **PDU Length (Send PDU Length Low)**

Indicates the maximum length of a send message.

PDU - Protocol Data Unit
In brief, a message.

The following applies: *Send PDU-Length* <= *Receive PDU-Length*

In some PROFIBUS devices, mostly simple slaves, the maximum permissible received message length cannot be changed and is smaller than 241 bytes. The maximum send message length on the master must therefore be configurable.

The maximum receive data length for our link is fixed at 241 bytes. Messages with higher priority are not supported at present, and so the lengths of this message type is specified firmly at zero.

The values for the send and receive PDU length are compared by the FMS on establishment of the connection. If an error occurs, establishment of the connection receives a negative acknowledgement.

4.2.3 The Local Object Directory (OD)

The local communications objects are defined in this part of the OD. These are the objects which the PROFIBUS device makes available to the opposite station. The opposite station can only access the objects defined here. Access to a non-defined object is refused by ALI (negative acknowledgement), which does not affect the user program.

- **Object Index**

Index number of the object. The object index is used by the opposite station to reference the object.

- **Description**

Symbolic name of the object.

- **Object Type**

The object type determines which object is involved. The following values are currently supported:

• Standard (Simple)	A single variable.
• Event	An event variable.
• Array	Serveral single variables of the same type.

Elements: The number of elements in this array object.

- **Data Type, Type Length**

Here, the type of variable or the type of array element is defined. Reference is made to the static data type directory. The following values are permissible:

Type-code	Datatype	Description	Value range for the data type	Type length Size of an element in bytes
1	Boolean	Boolean	TRUE(=255), FALSE(0)	1
2	Int8	Integer 8	-128..127	1
3	Int16	Integer 16	-32768..32767	2
4	Int32	Integer 32	-2^31-1 .. 2^31	4
5	UInt8	Unsigned Integer 8	0..255	1
6	UInt16	Unsigned Integer 16	0..65535	2
7	UInt32	Unsigned Integer 32	0..2^32 -1	4
8	Float	Float	IEEE-Float	4
10	Octet String	Octet String (Bytestring)	Byte string: interpretable by the user	Length of the byte string

Only in case of data type Octet String the Type Length has to be entered. In all other cases the Type Length is set automatically.

4.2.4 Optimization

If the performance of your PROFIBUS-FMS system is insufficient for your purposes, you may consider the following points for optimisation.

4.2.4.1 Communications Via Several Communications References

As a rule, transmission of several messages to one station is implemented in such a way that a new request is only transmitted to the opposite station when a confirmation has been received.

In order to reduce idle times which arise at station A while station B is processing the message, communications may take place via several communications references when there is a large volume of message traffic. If a message is "underway" on the first CR, the second CR can be used to transmit.

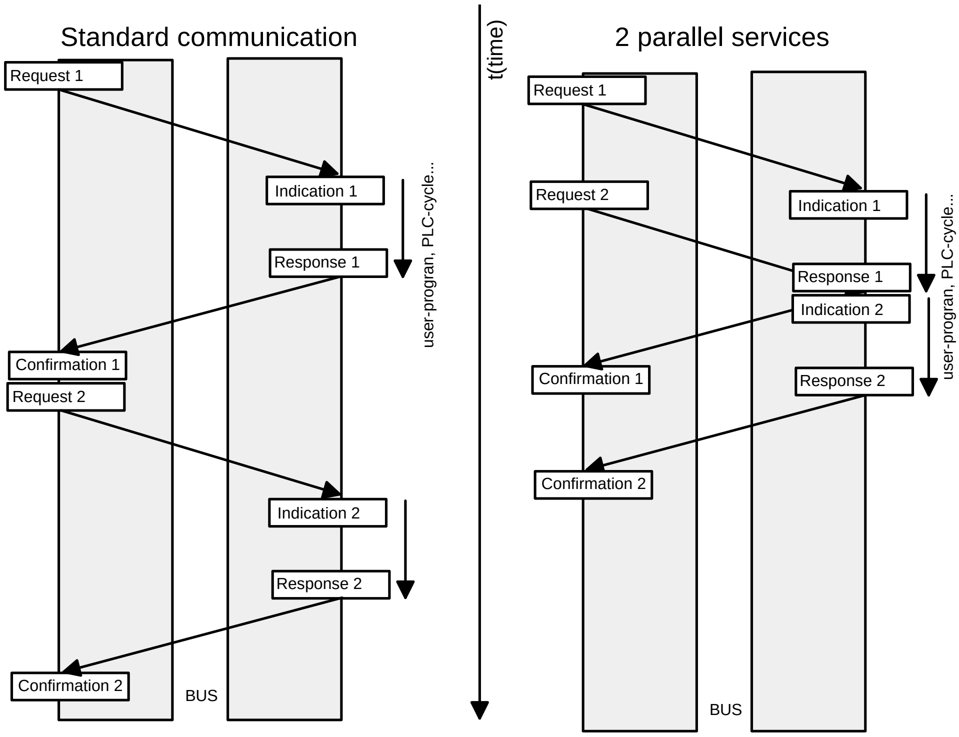
The user has to decide which CR is to be used for transmission. Customary arrangements involve, for example, the transmission of alarm values through a different CR than that for measured values or commands.

Please note that the user program has to be adapted for this method of optimization.

4.2.4.2 Communications Via Several Parallel Services

In the example above, it can nevertheless occur that measured values are most frequently transmitted and the CR through which the measured values are transferred is overloaded. In this case, it is useful to configure several parallel services on this CR.

In such a case, new requests are sent before the confirmations of previously requests have arrived. "Idle times" on the individual levels are thus used.



Principle of data transmission by means of two parallel services

With our PROFIBUS-FMS devices, you can process up to four parallel services per communications reference.

Parallel services are configured with the parameters SCC/RCC or SAC/RAC.

Configuration is to be performed on both PROFIBUS stations which are linked by this CR!

Please note that with this method of optimisation the user program has to be adapted.

5 Error Numbers and Trouble Shooting

5.1 Error Numbers of PROFIBUS-FMS

No. hex	No. dez	Description
0x00	0	No error
0x42	66	Connection has been aborted A local PROFIBUS-layer (FDL, LLI, FMS, ALI) or the remote partner has closed the connection. All requests by the client will be confirmed with this error number. All indications by the server will be lost. By sending a new request on this CR, the firmware will try to open the connection automatically by sending an initiate-telegram. **See additional details at message data to get more information, descriped in 'Connection aborted'**
0x43	67	Too many parallel services on one CR ALI received Reject service with Reject Code for Max-Service-Overflow. This means confirmed Service.Request received from the user and Out-standing Services Counter Client **≥** Outstanding Services Client. Check setting of ´SCC´ or 'SAC' in CRL. On FDL-communication there can be generally sended only one open service, independend of the SCC/SAC parameters in the CRL.
0x44	68	Invoke-ID in use for running service Check the consistence of your user-program according to the messagenumber. When you send a seccond telegram with the same messagenumber before receiving the answer of the first one, this error will be received.
0x45	69	Telegram too long to send ALI received Reject service with Reject Code for PDU-Size. This means PDU length exceeds maximum PDU length. Check telegram length or value of ´PDU-Length´ in CRL.
0x46	70	Client-Service not supported ALI received Reject service with Reject Code for Feature-Not-Supported-Connection-Oriented or Feature-Not-Supported-Connectionless. This means Service Request received from the user and service or option not supported as a client. Check client services in CRL.
0x47	71	Telegram cannot be evaluated by local FMS ALI received Reject service with Reject Code for Other.
0x48	72	Negative acknowledgement, resource of the remote FDL controller not available or not sufficent and reply data not available (RR error from FDL) The remote-FDL will send this error-number, when there is no segment or receivebuffer free to receive this telegram. It's a temporary error. By receiving this error, the local LLI will disconnect the connection.
0x49	73	No or no plausible reaction of remote FDL (NA error from FDL) There is no connection between the local and the remote FDL. Possibilities are, that the stations are not connected, switched off or the bus-parameters are invalid or inconsistent. Check with default-BUS-parameters.
0x4A	74	FMS service received at closed Communication Reference

PROFIBUS-FMS Error Numbers

No. hex	No. dez	Description
0x00	0	No error
0x50	80	Parallel service during FDL defined FDL defined don't support parallel services.
0x55	85	Abort is send from user during FDL defined
0x56	86	Invoke Id already exists ALI received Reject service with Reject Code for Invoke-Id-Exists. This means confirmed Service Request received from the user and Invoke ID already exits.
0x60	96	FDL error received This range containes the most used FDL errors. All other FDL errors without the following.
0x61	97	UE Negative acknowledgement, remote User interface error.
0x62	98	RR Negative acknowledgement, no remote resources available.
0x63	99	RS No service or no remote address activated at Remote Service Access point.
0x64	100	RA Negative acknowledgement, access-point blocked
0x68	104	DL Positive acknowledgement for sent data, reply data with low priority available.
0x69	105	NR Positive acknowledgement for sent data, negative acknowledgement for reply data, as not available to the remote FDL controller
0x6A	106	DH Positive acknowledgement for sent data, reply data with high priority available.
0x6C	108	RD Negative acknowledgement for sent data, resource of the remote FDL controller not available or not sufficent, reply data with low priority available.
0x6D	109	RDH Negative acknowledgement for sent data, resource of the remote FDL controller not available or not sufficent, reply data with high priority available.
0x70	112	LS Service at local SAP or local SAP not activated.
0x71	113	NA No reaction (ACK/RES) from remote station
0x72	114	DS Local FDL/PHY controller not in logical token ring or disconnected from line.
0x73	115	NO No reply data transmitted.
0x74	116	LR Resource of the local FDL controller not available or not sifficient.
0x75	117	IV Invalid parameter in request.
0x76	118	BR
0x77	119	NM
0x78	120	NC
0x79	121	NP
0x7A	122	NL
0x7B	123	RD
0x7C	124	SV

PROFIBUS-FMS Error Numbers

No. hex	No. dez	Description
0x80	128	Connection could not be opened By the first request the connection has to be opened by sending an initiate-telegram. If the remote-partner confirmed this initiate negative, the connection could not be opened and the request will be reject with this error. Please check the local and remote configuration of this CR in KBL. **See error code at message data to get more information,** **described in chapter 'Connection could not be established'.**
0x81	129	Error in application of remote-partner The communication partner (server) has reject the request with an error. Possible reasons: a) Access on an non existing object b) Data-length of sending data is not consistent to data-length of object c) The user-program has reject the service **See error type at message data to get more information,** **described in chapter 'Negative response from remote station - Error-Type'**.

PROFIBUS-FMS Error Numbers

No. hex	No. dez	Description
0x87	135	Local state does not allow to send You can configure, that there can only be send telegrams after be taken up in the logical token-ring. Before taken up in the logical token-ring, all commissions will be rejected with this error.
0x88	136	FMS reply contains error number at msg.f
0x89	137	The ALI-task is not initialized Check configuration and download.
0x8A	138	Invalid datatype Check datatype-unsign char of your commission. On FDL-transparent there is only allowed the OCTET-STRING
0x8C	140	Invalid data count Check consistence of msg.ln-unsign char to len-unsign char.
0x8D	141	Invalid FMS service
0x8E	142	Unknown Msg.nr / Invoke ID get from user
0x8F	143	Non expected reply received from user
0x90	144	Unknown bAccessSpec
0x91	145	Unknown AccessMode
0x92	146	OV_NEXT without OV_FIRST
0x93	147	Access error for local object
0x94	148	Objet non exsistent on local station
0x95	149	To less data in reply from user
0x96	150	Invalid Invoke ID from FMS
0x97	151	Invalid Communication Reference
0x98	152	Message length is less than telegramm header
0x9A	154	No free segments available
0x9B	155	Invalid PROFIBUS service Check service-unsign char of your commission
0x9C	156	Invalid command from user Check msg.b.
0x9D	157	Invalid answer from user Check Msg.a.
0xA0	160	Unknown service for FDL defined
0xA1	161	Invalid bDataType for FDL defined
0xA2	162	Invalid bDataCnt for FDL defined
0xA3	163	Local SAP non existing on actual CR for FDL communication
0xA4	164	Unknown FDL reply
0xA5	165	Error at reply from FDL defined
0xA6	166	Reply from FDL defined containes to less data
0xB0	176	Invalid logical state
0xB1	177	Invalid physical state
0xB2	178	Invalid DetailLen by setting VFD state
0xC1	193	Database error during initialisation
0xC2	194	Not supported object code at object directory
0xC4	196	To many Communication References
0xC6	198	Error during access at the database

PROFIBUS-FMS Error Numbers

5.1.1 Connection could not be Established - Error 0x80, 128

If the connection is not established then the ALI generates with the first service received from the user an Initiate service on this communication reference. If the remote station answeres with a negative result, then the following error message will be send back to the user.

If this error occur the connection is not established.

The Error Codes are defined in accordance EN 50170-6-2:1996, page 438 and are described in the following table.

Error code	Description
0	Other Reason other than any of those identified below.
1	Max PDU size insufficient The remote-FMS will generate this error, when the configured max. receive-PDU-length is lower then the local max send-PDU-length
2	Feature not supported The requested service or option is not supported by the server. Check client- and server-services in the Communication Reference list.
3	The OD incompatible The versions of the Object Dictionaries (local OD and remote OD) of the communication partners are not compatibel.
4	User Initiate denied The FMS user from the remote station refuses to establish the connection
5	Password error There is already a communication relationship established with the same password.
6	Profile number incompatible The client's profile is not supported by the server.

Error Codes for connection could not be established

5.1.2 Negative response from Remote Station - Error 0x81, 129

This is the most frequently error which is generated if the server detects an error during the execution of a service and answers mit a negativ result.

You will find the additional error information in the PROFIBUS standard as Error Type in the Result(-) of all confirmed services.

This error is return for the services:

- Read
- Write
- Status
- Identify
- GetOD

In case of this error the connection will not be closed.

The errors are divided in Error classes with own Error Codes. These are defined in accordance EN 50170-6-2:1996, page 432 and are described in the following table.

Error class	Error code	Description
1		This error class is returned whenever the state of the VFD is such that the request service my not be executed.
	0	Other error as described below.
2		This error class is related to the communication relationship on which the service is executed.
	0	Other error as described below
	1	The application process is not reachable.
3		This Error Class is returned when there are problems with object definitions.
	0	Other error as described below
	1	The required objects do not exist.
	2	The indicated objects are defined with inconsistent attributes.
	3	The name exists already.
4		This Error Class is returned when available resources are exceeded.
	0	Other error as described below
	1	There is no more memory for the execution of this service.
5		This Error Class is returned whenever there are problems with the service itself.
	0	Other error as described below
	1	The current state of the object does not permit execution of the service.
	2	Response-PDU > Max-PDU-Sending-Low-PRIO.
	3	The execution of the service is not possible at this time.
	4	The service contains inconsistent parameters.
	5	A parameter has an illegal value.
6		This Error Class is returned whenever an access is faulty.
	0	Other error as described below
	1	The access refers to a defined object with an undefined Reference attribute.
	2	The access to the object failed because of a hardware error.
	3	The FMS client has not sufficient Access Rights for the object.
	4	The indicated Address is out of the legal range.
	5	The attributes of the objects are inconsistent.
	6	Object has not been defined for this access.
	7	The object does not exist.
	8	The access is rejected because of an incorrect Data Type.
	9	The access with names is not supported.
7		This Eror Class is returned whenever an incorrect change to the OD is made.
	0	Other error as described below
	1	The legal length of the name is exceeded.
	2	The legal length of the OD is exceeded.
	3	The OD is write protected.
	4	The legal length of the Extension is exceeded.
	5	The legal length of a single Object Description is exceeded.
	6	The currently loaded OD is incorrect.
8		This Error Class is returned to a reason other than any of those listed above.
	0	Other error as described below

Error class / Error Code of negative response

5.1.3 Connection Aborted - Error 0x42, 66

If the remote station sends an Abort on an established communication relationship the local ALI create a confirmation with the error 0x42, 66 for all outstanding services. The reason for the Abort is signalled in the message data as described in the PROFIBUS standard for the Abort indication.

After receiving an Abort the Communication relationship is closed.
The Abort Identifier defines which Layer has generated the Abort and the Reason Code defines the error itself in accordance EN 50170-5-2:1996, page 308 and EN 50170-6-2:1996, page 684. The Abort can be generated from the local or the remote station, which is signaled in the parameter 'Abort Generated'.
In some case there are additional details available which are defined by the user.

With error 0x42, 66 additional infomation will come in the following structure:

Parameter	Information	Value
Invoke ID	ID	0x00..0xFF 0..255
Abort Generated	Local Remote	0xFF, 255 0x00, 0
Abort Identifier	ALI / User	0
	FMS	1
	LLI	2
	FDL	3
Reason Code	Error Code	See following table
Additional Information	Additional Information	Specified by user

Additional Error Information

5.1.3.1 ALI / User Reason Codes

Reason Code	Description
0	ALI-timer overrun on this communication reference The ALI-timer controls the time between sending a request and receiving the confirmation of this request. Check configuration in KBL.
1	Version OD incompatible The versions of the OD (source OD and remote OD) of the communication partners are not compatible.
2	Password Error There is already a communication relationship established with the same password.
3	Profile Number incompatible The server's profile is not supported by the client.
4	Limited Services Permitted The VFD is in the Logical Status LIMITED-SERVICES-PERMITTED.
5	OD loading interacting The PutOD Service which is not free of interaction is currently active.

ALI / User Reason Codes

5.1.3.2 FMS Reason Codes

Reason Code	Description
0	FMS CRL Error Faulty FMS CRL Entry.
1	User Error Improper, unknown or faulty service primitive received from the FMS user.
2	FMS PDU Error Unknown or faulty FMS PDU received from the LLI.
3	Connection State Conflict LLI Improper LLI service primitive.
4	LLI Error Unknown or faulty LLI service primitive.
5	PDU Size PDU length exceeds maximum PDU length.
6	Feature not supported SERVICE_REQ_PDU received from LLI and service or option not supported as a server.
7	Invoke ID Error Response Confirmed service.res received from the FMS user and Invoke ID does not exist.
8	Max Service Overflow CONFIRMED-SERVICE_REQ_PDU received from the LLI and Outstanding Services Counter Server ≥ Outstanding Services Servcer.
9	Connection State Conflict FMS INITIATE_REQ_PDU or INITIATE_RES_PDU received from the LLI.
10	Service Error The service in the response does not match the service in the indication or the service in the confimation does not match the service in the request.
11	Invoke ID Error Request CONFIRMED-SERVICE_REQ_PDU received from LLI and Invoke ID already exists.
12	FMS disabled FMS is not ready for data transmission.

FMS Reason Codes

5.1.3.3 LLI Reason Codes

Reason Code	Description
0	LLI-LLI context-check negative The configuration concerning the CRL of the two partners is inconsistent.
1	Unallowed LLI PDU received in the connection establishment phase or release phase This error happens often, when the communication-partner has been switched off/on during datatransfer.
2	Unallowed LLI PDU received in the data transfer phase This error happens often, when the communication-partner has been switched off/on during datatransfer.
3	Unknown or faulty LLI PDU received
4	DTA_ACK_PDU received and SAC = 0 Check parameter SAC/RAC in KBL.
5	Number of parallel services exceeded
6	Invoke ID unknown
7	Priority error
8	Local error in the remote station Read manual of communication-partner. If our firmware sends this error, there is a system-error. Please contact your distributor.
9	Timer T1 expired (connection establishment) Remote-partner did not answer during a fixed time. If this time is too short, please contact our hotline.
10	Timer T3 expired (LLI-timer control interval, connection monitoring) No answer received for LLI-IDLE-telegram from remote-partner. Check local and remote LLI-timer in CRL.
11	RTimer expired (LLI-receive-timer control interval) No LLI-IDLE-telegram from remote-partner received. Check local and remote LLI-timer in CRL.
12	Error in the LSAP actication
13	Unallowed FDL primitive in the connection establishment phase or release phase
14	Unallowed FDL primitive in the data transfer phase
15	Unknown FDL primitive
16	Unknown LLI primitive
17	Unallowed LLI primitive in the connection establishment phase or release phase
18	Unallowed LLI primitive in the data transfer phase
19	CRL entry not OK
20	Conflict case in the connection establishment phhase; local address > remote address This error can only happen on MMAZ-connections. If both masters send at the same time initiate-telegrams, the master with the higher address has to send an abort.
21	Execution error in cyclic data transfer
22	Number of parallel services exceeded
23	CRL is being loaded by FMA7, LLI disabled
24	Confirm / indication mode error
25	Unallowed FMA1/2 primitive received
26	Unallowed FMS service on connection for cyclic data transfer
27	FSM PDU size exceeded on connection for cyclic data transfer

LLI Reason Codes

5.1.3.4 FDL Reason Codes

Reason Code	Description
0	No error occured
1	Remote user (LLI) or interface-error
2	Recources of Remote-FDL-not available or exhausted The remote-FDL will send this error-number, when there is no segment or receivebuffer free to receive this telegram. It's a temporary error. By receiving this error, the local LLI will disconnect the connection.
3	Service, remote-address or remote-SAP is by remote-FDL not active
12	Recources of the remote-FDL-controller/user not available or exhausted
13	Recources of the remote-FDL-controller/user not available or exhausted
16	Service or local SAP not active
17	No or no logical reaction of remote FDL There is no connection between the local and the remote FDL. Possibilities are, that the stations are not connected, switched off or the bus-parameters are invalid or inconsistent. Check with default-BUS-parameters.
18	Local FDL not in logical token-ring or disconnected from bus
19	No reply data transmitted
20	Recources of the remote-FDL-controller/user not available or exhausted
21	Invalid parameter in request

FDL Reason Codes

5.2 Trouble Shooting on PROFIBUS-FMS

Experience shows that problems repeatedly occur with the commissioning of PROFIBUS devices. This chapter is intended to assist you in diagnosting and r-ectifying faults.

Checking the configuration during start up does not guaranteee error free communications! The configuration of the communications partner also has to be in line with the local configuration!

The first step towards functioning PROFIBUS communications is carefully planned configuratoin. The configuration data are then downloaded with the system configurator SyCon and transferred to the device.
The PROFIBUS device checks on start up whether the local configuration data are plausible. If the configuration is correct, the run and ready LEDs on the device light up.

If the PROFIBUS device does not start up, check which task has signalled an initialisation error.

If the device has been successfully initialised, check whether all the stations connected to the bus have been detected. This is accomplished by selecting the menu online - live list in the system configurator.

Before the actual data can be transmitted, a connection to the communications partner first has to be established. If establishment of the connection fails, the reason is usually an incorrect entry in the communications relationship list. You should therefore check whether the entries in the communications relationship list are compatible with the communications partner.

a) The local SAP must correspond to the remote SAP at the partner station - and vice versa.

b) The target device address must be correct.

c) The send counters SCC/SAC must be greater or equal to the receive counters RCC/RAC at the partner.

d) The monitoring time LLI-Timer (Control Interval) must be equal.

e) The communications type MMAZ/MSAZ/... must be identical.

f) The receive PDU length must be equal to or greater than the send PDU length of the partner station.

g) The server must support at least those services which the client can request.

After the successful establishment of a connection, the user data can be transmitted. An access error to an object is always reported when ...

a) ... the object is not defined at the server.

b) ... the object is incorrectly defined at the server.
Check in particular the data length of the object and the length of the data transmitted for agreement.

c) ... the server cannot access the object.
This occurs for example in an PLC when the program in the PLC is not running or the CP is stopped. Also check that the data module assigned to the object exists and is of sufficient length.

5.2.1 Initialization Errors

The initialisation errors provide information on the device status after initialisation. These errors are used to evaluate inconsistent configuration data.

5.2.1.1 The Initialisation Errors for the FDL Task

Definition	Description
17	user not available
19	database not available
20	database cannot be readed
21	invalid value of configured parameter
22	not enough segments available
23	busaddress invalid

Initialisation errors for the FDL task

5.2.1.2 The Initialisation Errors for the LLI Task

Definition	Description
4	FDL-connection with MSZY configure only MMAZ-CRs by FDL-communication
8	BRTC-receiver and local SAP different of 63 the local SAP of the broadcast receiver has to be SAP 63
11	local SAP = poll-SAP, but no master-slave CR the poll-SAP is only can be used in master-slave CRs as local SAP for the master - and in no other case!
13	too many CRs confogured the max. number of CRs is 32 at this time
18	too many parallel services configured SAC+RAC <= 4 and SAC+RAC <=4
19	Local SAP > 63 oder reserved Byte != 0 check local SAP number
20	unexpected message from FDL in this case the FDL-task is not initialized
22	MSAZ (master): RCC > 0 oder RAC > 0 the receive-counter of the master by MSAZ have to be zero
23	MSAZ (slave): SCC > 0 oder SAC > 0 the send-counter of the slave by MSAZ have to be zero
24	MSZY/BRTC/MULT: SCC or RCC or SAC or RAC are not zero set on these CRs the counters SCC/RCC/SAC/RAC to zero
25	BRTC responder and remote SAP is 63 or remote address is 127 responder on broadcast relationships, because the local SAP== 63. Therefore the remote SAP cannot be 63 and the remote address cannot be 127
26	BRTC requester and remote address is not 127 requester on broadcast relationships, because the remote SAP is 63. Therefore the remote address cannot be 127
28	invalid remote SAP
29	invalid remote address

Initialisation errors for the LLI Task

5.2.1.3 The Initialisation Errors for the ALI Task

Definition	Description
128	error by initializing object-directory
129	error by initializing commission-controlling
130	error by initializing immagefunctions check th consistence of the configuration, specially the tables ALI_CLIENT/ALI_SERVER
131	too many communication references

Initialisation errors for the ALI Task

6 Diagnostic Functions

For diagnostic the diagnostic LEDs on the device and over the serial diagnostic interface the possibilities with the system configurator SyCon-PB can be used.

6.1 Diagnostic LEDs

To diagnostic the device the diagnostic LEDs on the front panel can be used when the device has finished its initialization-sequence. The upper yellow LED RDY shines continous after the operation system itself has initialized without an error and no hardware defect has determined.
The green LED RUN blinks cyclic, if the protocol tasks haven't determined nor an hardware error nor an parameterization error. In this case the tasks are ready to start the communication. The LED shines continous, if the device has a data exchange communication with at least one projected slave participant.
The LED ERR shines additionally, if a bus error occured. This can be for example possible, if an projected slave can't found on bus.
The yellow LED STA represent the hold-token of the master. As long as the device doesn't recognize any other master systems on the bus, the device retain the right to send. Therefore the LED shines continuous.

6.2 Task State

The debug mode of the system configurator offers to display task states.

6.2.1 FDL_ASPC Task

State table 'Common variables':

variable	signification
Task state	task state
Last FDL error state	last FDL error state
Last FDL init error	initialization error of FDL
Last FDL runtime error	runtime error code
Last FDL message error	last message error number
ASPC2 bus short circuits	Count of 'bus syncon error' of the ASPC2
ASPC2 bus error	Count of 'bus error' of the ASPC2
Free application blocks	free application blocks of the software
Free SAP blocks	free SAP blocks of the software
Free CLASS2 blocks	free CLASS2 blocks of the software

While the system is running the condition 'task state' changes between the values of '10, 11, 21.' The FDL error values are changed only at runtime, if an error occurs between the main-processor and ASPC2-PROFIBUS-Controller. Is an error value displayed, please inform our hotline.

State table 'act. bus parameters':

variable		signification
Highest station address	HSA	highest station-address the master polls
Bus address of this master station	TS	the master bus address
Transmission rate		transmission rate, definition range of ASPC2
Maximum request retry in errorcase		retry-number of requestor
Slot time (Bit)	TSL	slot time
Transmitter fall time (Bit)	TQUI	transmitter fall time
Setup time (Bit)	TSET	setup time
Smallest station delay (Bit)	MIN-TSDR	minimum STATION-DELAY
Largest station delay (Bit)	MAX-TSDR	maximum STATION-DELAY
Target rotation time (Bit)	TTR	target rotation time
GAP update factor	G	gap update factor

The activated master parameters are shown in this table.

State table 'DDLM requests cl.1'

variable	signification
Set bus parameters	counter of 'set-bus-par' requests
Set slave address list	counter of 'set-slave-list' requests
Slave diagnostic request	counter of 'slave-diag' requests
Set parameters request	counter of 'set-prm' requests
Check configuration request	counter of 'check-cfg' requests
global control request	counter of 'global-control' requests
Data exchange request	counter of 'data-exchange' requests
Data exchange conf. pos.	counter of 'data-exchange' confirmation, positiv
Data exchange conf. neg.	counter of 'data-exchange' confirmation, negativ
Data exchange all request	counter of 'data-exchange-all' requests
Data exchange all conf. pos.	counter of 'data-exchange-all' confirmation, positiv
Data exchange all conf. neg.	counter of 'data-exchange-all' confirmation, negativ

The counter for master services of the PROFIBUS-DP are listed in this table. Basically the 'data exchange all' counter should increment at a faultless network only. The services 'set-slave-list', 'set-prm', 'chk-cfg', 'data-exchange' or 'slave-diag' might be incremented at faultless network for each projected slave only once. If an bus error occurs all these services are also increased.

If for example the service 'slave-diag' increases sporadic, maybe the bus cable has a defect or the conclusion-resistor in the cable is missing. Furthermore it can be possible that a slave reports an error in his diagnostic data, so that the master automatically starts the service 'slave_diag'.

State table 'DDLM requests cl.2'

variable	signification
Get configuration request	counter of 'Get_Cfg' requests
Get configuration conf. pos.	counter of 'Get_Cfg' confirmations positiv
Get configuration conf. neg.	counter of 'Get_Cfg' confirmations negativ
Read input request	counter of 'RD_Inp' requests
Read Input conf. pos.	counter of 'RD_Inp' confirmations positiv
Read Input conf. neg.	counter of 'RD_Inp' confirmations negativ
Read output request	counter of 'RD_Outp' requests
Read output conf. pos.	counter of 'RD_Outp' confirmations positiv
Read output conf. neg.	counter of 'RD_Outp' confirmations negativ
Set slave address request	counter of 'Set_Slave_Add' requests
Set slave address conf. pos.	counter of 'Set_Slave_Add' confirmations positiv
Set slave address conf. neg.	counter of 'Set_Slave_Add' confirmations negativ
Get master diag request	counter of 'Get_Master_Diag' requests
Get master diag conf. pos.	counter of 'Get_Master_Diag' confirmations positiv
Get master diag conf. neg.	counter of 'Get_Master_Diag' confirmations negativ
Get master diag indication	counter of 'Get_Master_Diag' indications
Get master diag response	counter of 'Get_Master_Diag' responses
Download request	counter of 'Download' requests
Download conf. pos.	counter of 'Download' confirmations positiv
Download conf. neg.	counter of 'Download' confirmations negativ
Download indication	counter of 'Download' indications
Download response	counter of 'Download' responses
Upload request	counter of 'Upload' requests
Upload conf. pos.	counter of 'Upload' confirmations positiv
Upload conf. neg.	counter of 'Upload' confirmations negativ
Upload indication	counter of 'Upload' indications
Upload response	counter of 'Upload' responses
Start sequence request	counter of 'Start_Seq' requests
Start sequence conf. pos.	counter of 'Start_Seq' confirmations positiv
Start sequence conf. neg.	counter of 'Start_Seq' confirmations negativ
Start sequence indication	counter of 'Start_Seq' indications
Start sequence response	counter of 'Start_Seq' responses
End sequence request	counter of 'End_Seq' requests
End sequence conf. pos.	counter of 'End_Seq' confirmations positiv
End sequence conf. neg.	counter of 'End_Seq' confirmations negativ
End sequence indication	counter of 'End_Seq' indications
End sequence response	counter of 'End_Seq' responses
Act.param. brct. request	counter of 'Act_Param_Brct' requests
Act.param. brct. conf. pos.	counter of 'Act_Param_Brct' confirmations positiv
Act.param. brct. conf. neg.	counter of 'Act_Param_Brct' confirmations negative
Act.param brct. indication	counter of 'Act_Param_Brct' indications
Act.param. brct. response	counter of 'Act_Param_Brct' responses
Act.param. request	counter of 'Act_Param' requests

Act.param. conf. pos.	counter of 'Act_Param' confirmations positiv
Act.param. conf. neg.	counter of 'Act_Param' confirmations negativ
Act.param. indication	counter of 'Act_Param' indications
Act.param. response	counter of 'Act_Param' responses

The counter for Class 2 services of the PROFIBUS-DP are listed in this table. Here, request (out going) and indications (in coming) are counted.

Not all functions are supported in this implementation.

State table 'FDL-requests'

variable	signification
SDA Request	counter of 'SDA' requests
SDA Confirmation pos.	counter of 'SDA' confirmations, positiv
SDA Confirmation neg.	counter of 'SDA' confirmations, negativ
SDA Indication	counter of 'SDA' indications
SDN Request	counter of 'SDN' requests
SDN Confirmation pos.	counter of 'SDN' confirmations, positiv
SDN Confirmation neg.	counter of 'SDN' confirmations, negativ
SDN Indication	counter of 'SDN' indications
SRD Request	counter of 'SRD' requests
SRD Confirmation pos.	counter of 'SRD' confirmations, positiv
SRD Confirmation neg.	counter of 'SRD' confirmations, negativ
SRD Indication	counter of 'SRD' indications
SRD-Update Request	counter of 'SRD' update requests
SRD-Update Con. pos.	counter of 'SRD' update confirmations, positiv
SRD-Update Con. neg.	counter of 'SRD' update confirmations, negativ

The counter for FMS services of the PROFIBUS are listed in this table. Here, requests (out going), its confirmation (positiv or negativ) and indications (in coming) are counted.

Not all functions are supported in this implementation.

State table 'FMA requests'

variable	signification
SAP act. request	counter of 'SAP Aktivate' requests
SAP act. confirmation pos.	counter of 'SAP Aktivate' confirmations, positiv
SAP act. confirmation neg.	counter of 'SAP Aktivate' confirmations, negativ
RSAP act. request	counter of 'RSAP Aktivate' requests
RSAP act. confirmation pos.	counter of 'RSAP Aktivate' confirmations, positiv
RSAP act. confirmation neg.	counter of 'RSAP Aktivate' confirmations, negativ
SAP deact. request	counter of 'SAP Deaktivate' requests
SAP deact. confirmation pos.	counter of 'SAP Deaktivate' confirmations, positiv
SAP deact. confirmation neg.	counter of 'SAP Deaktivate' confirmations, negativ
LiveList request	counter of 'LiveList' requests
LiveList confirmation pos.	counter of 'LiveList' confirmations, positiv
LiveList confirmation neg.	counter of 'LiveList' confirmations, negativ

The counter for FMA services of the PROFIBUS are listed in this table. Here, requests (out going), its confirmation (positiv or negativ) and indications (in coming) are counted.

Not all functions are supported in this implementation.

7 Examples

7.1 PROFIBUS-FMS

7.1.1 Typical MMAC Configuration

The following two tables show an entry in the CRL at station with station address 1 and an entry in the CRL at station with station address 2 that defines a logical connection between these two devices. This example shows a typical Master-Master Acyclic FMS connection.

Local SAP	2	2
Remote SAP	2	2
Remote Station	2	1
Communication Type	MMAC	MMAC
LLI-User	FMS	FMS
SCC	1	1
RCC	1	1
SAC	0	0
RAC	0	0
Client	Read/Write	Read/Write
Server	Read/Write	Read/Write
LLI Timer (Control Interval)	0	0
ALI Timer	0	0
PDU Length	241	241
	Entry in CRL at station with station address 1 that defines an logical connection to station address 2	Entry in CRL at station with station address 2 that defines an logical connection to station address 1

CRL entry at station address 1 and station address 2

7.1.2 Typical MSAC Configuration

The following two tables show an entry in the CRL at station with station address 1 and an entry in the CRL at station with station address 2 that defines a logical connection between these two devices. This example shows a typical Master-Slave Acyclic FMS connection.

Local SAP	58	2
Remote SAP	2	58
Remote Station	2	1
Communication Type	MSAC	MSAC
LLI-User	FMS	FMS
SCC	1	0
RCC	0	1
SAC	0	0
RAC	0	0
Client	Read/Write	-
Server	-	Read/Write
LLI Timer (Control Interval)	0	0
ALI Timer	0	0
PDU Length	241	241
	Entry in CRL at station with station address 1 that defines an logical connection to station address 2	Entry in CRL at station with station address 2 that defines an logical connection to station address 1

CRL entry at station address 1 and station address 2

7.1.3 FMS Starterkit

The following two tables show the entry in the CRL at station with station address 1 (Client) and the entry in the CRL at station with station address 2 (Server) that defines a logical connection between these two devices.

	Client	Server
Local SAP	2	2
Remote SAP	2	2
Remote Station	2	1
Communication Type	MMAC	MMAC
LLI-User	FMS	FMS
SCC	1	1
RCC	1	1
SAC	0	0
RAC	0	0
Client	Read/Write	Read/Write
Server	Read/Write	Read/Write
LLI Timer (Control Interval)	0	0
ALI Timer	0	0
PDU Length	241	241

Entry in CRL at station with station address 1 that defines an logical connection to station address 2	Entry in CRL at station with station address 2 that defines an logical connection to station address 1

The data base FMS_SKIT.PB already has these settings.

You need on the Server PC an application. For Windows 95, Windows 3.11 and DOS you can use the program SRV_DOS.EXE. To start the server application use SRV_APP.BAT.

This program first read out the object directory and displays these information on the screen.

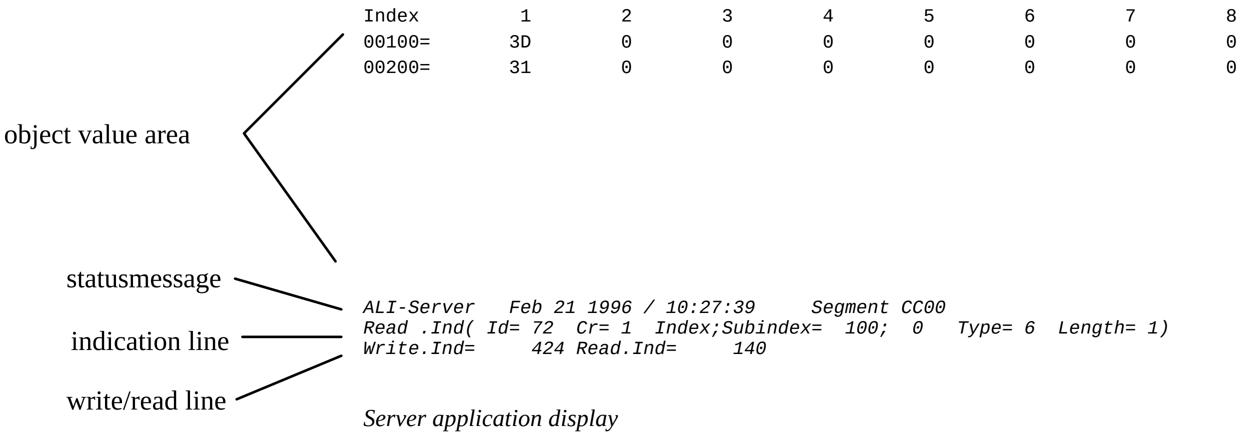
```
Extended ALI-Server, compiled on ... Feb 21 1996 / 10:27:39

CIF-Base segment = CC00

Driver opened without error
CIF initialization without error

Reading objectdirectory ...
-----------------------------
Index   Name             OC       Cnt   Type          Len
  100   Obj 100          SIMPLE     1   6(Uint16   )   2
  200   Obj 200          ARRAY     10   6(Uint16   )   2
Press <RETURN> to continue ...
```

After confirm with <ENTER> you will see the status-screen.



```
Index         1        2        3        4        5        6        7        8
00100=       3D        0        0        0        0        0        0        0
00200=       31        0        0        0        0        0        0        0








ALI-Server   Feb 21 1996 / 10:27:39     Segment CC00
Read .Ind( Id= 72  Cr= 1  Index;Subindex=  100;  0   Type= 6  Length= 1)
Write.Ind=     424 Read.Ind=     140
```

Server application display

After the start the object value area, the indication- and the write/read line are empty. These areas will be written during communication.

Object value area

In this area, all the objects managed by the server and their values are displayed. A maximum of 20 objects can be managed by this example server. The figures one to eight in the first line indicate the sub-index of the object. Only the first eight elements of an object are therefore **displayed**, but all the elements of the object are managed internally.

Note: The communications interface checks on receipt of an indication whether an object which is to be accessed is available in the local object directory. If that is not the case, the CIF refuses access. The server application can therefore only receive reads and writes to objects which are defined in the local object directory.

Status message

Fixed message comprising the CIF-address and the last compilation date.

Indication line

The incoming indications (enquiries) with the most important parameters are displayed in this line.

Write/Read counter

Two read and write counters which display the number of incoming and evaluated reads and writes are displayed in the last line.

Use the FMS Monitor in the system configurator (Menu Online - FMS Monitor) to send read and write telegrams from the client to the server.