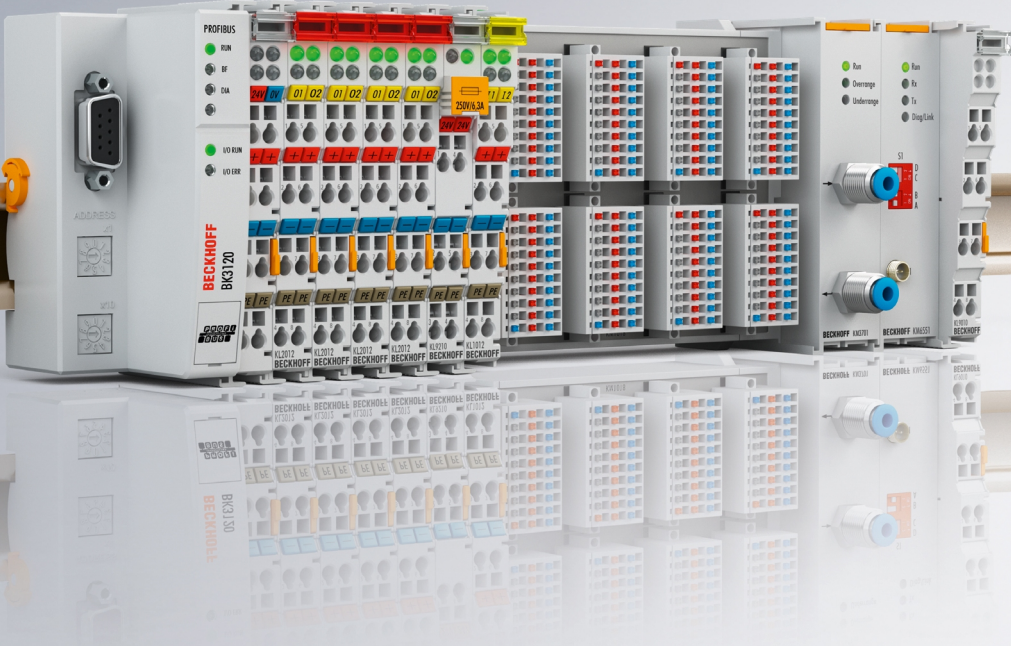


Documentation | EN

## KM6551

Terminal module for wireless data exchange





# Table of contents

<b>1 Foreword .....</b>	<b>5</b>
1.1 Notes on the documentation.....	5
1.2 Safety instructions .....	6
1.3 Documentation issue status .....	7
<b>2 Product overview.....</b>	<b>8</b>
2.1 Introduction.....	8
2.2 Technical data .....	9
2.3 Basic Function Principles .....	10
2.4 LED displays.....	12
2.5 DIP switch.....	13
<b>3 IEEE802.15.4 .....</b>	<b>14</b>
3.1 Introduction.....	14
3.2 Interference caused by other radio systems.....	15
<b>4 Mounting and wiring.....</b>	<b>17</b>
4.1 Recommended mounting rails.....	17
4.2 Mounting and demounting - terminals with traction lever unlocking .....	17
4.3 Dimensions .....	19
4.4 Connection .....	19
4.5 Antenna alignment.....	21
4.5.1 Directional characteristic.....	21
4.5.2 Alignment examples .....	23
4.5.3 Polarization .....	23
4.5.4 Placement of the antennas .....	24
4.6 Attenuation and range .....	25
4.6.1 Fresnel zone .....	25
4.6.2 Attenuation in practice .....	26
4.6.3 Range of different antenna combinations .....	26
4.7 Antennas .....	27
4.7.1 ZS6100-0900 .....	28
4.7.2 ZS6100-1800 .....	30
4.7.3 ZS6200-0400 .....	32
4.7.4 ZS6201-0410 .....	34
4.7.5 ZS6201-0500 .....	36
<b>5 Application examples - overview .....</b>	<b>38</b>
5.1 Peer to peer mode.....	38
5.2 Master-Slave mode .....	38
5.3 Broadcast mode .....	39
5.4 Energy scan.....	40
<b>6 TwinCAT .....</b>	<b>42</b>
6.1 TwinCAT libraries .....	44
6.2 TwinCAT examples .....	44
6.3 Function blocks.....	45
6.3.1 Function block FB_KM6551_MAIN.....	45

6.3.2	Energy scan .....	46
6.3.3	Master/Slave mode .....	47
6.3.4	Broadcast mode .....	49
<b>7</b>	<b>KS2000 Configuration Software .....</b>	<b>52</b>
7.1	KS2000 - Introduction .....	52
7.2	Parameterization with KS2000 .....	53
7.3	Settings .....	55
7.4	Register .....	56
7.5	Process data .....	57
<b>8</b>	<b>Access from the user program .....</b>	<b>59</b>
8.1	Process image .....	59
8.2	Control and Status Bytes .....	59
8.2.1	Process data mode .....	59
8.2.2	Register communication .....	61
8.3	Register overview .....	62
8.4	Register description .....	63
8.5	Examples of Register Communication .....	65
8.5.1	Example 1: reading the firmware version from Register 9 .....	65
8.5.2	Example 2: Writing to an user register .....	66
<b>9</b>	<b>Appendix .....</b>	<b>69</b>
9.1	General operating conditions .....	69
9.2	EC declaration of conformity .....	71
9.3	Calculating with decibels .....	72
9.4	Support and Service .....	72

# 1 Foreword

## 1.1 Notes on the documentation

### Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.

It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.

It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

### Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

### Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

### Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.

The logo for EtherCAT, featuring the word "EtherCAT" in a bold, black, sans-serif font. A red arrow points from the top of the "A" to the top of the "T". A registered trademark symbol (®) is located to the right of the "T".

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

### Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

## 1.2 Safety instructions

### Safety regulations

Please note the following safety instructions and explanations!  
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

### Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

### Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

### Description of instructions

In this documentation the following instructions are used.  
These instructions must be read carefully and followed without fail!

#### **DANGER**

##### **Serious risk of injury!**

Failure to follow this safety instruction directly endangers the life and health of persons.

#### **WARNING**

##### **Risk of injury!**

Failure to follow this safety instruction endangers the life and health of persons.

#### **CAUTION**

##### **Personal injuries!**

Failure to follow this safety instruction can lead to injuries to persons.

#### **NOTE**

##### **Damage to environment/equipment or data loss**

Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.



##### **Tip or pointer**

This symbol indicates information that contributes to better understanding.

### 1.3 Documentation issue status

Version	Comment
2.0.0	<ul style="list-style-type: none"> <li>• Migration</li> <li>• Structure update</li> </ul>
1.2.0	<ul style="list-style-type: none"> <li>• Included TwinCAT library [▶ 44] updated to version 1.7.0</li> <li>• Description of the KS2000 configuration software updated</li> </ul>
1.1.0	<ul style="list-style-type: none"> <li>• Register description extended</li> <li>• Notes on interference caused by other radio systems expanded</li> <li>• Notes on mounting expanded</li> <li>• Antenna ZS6201-0500 added</li> <li>• Antenna descriptions updated</li> </ul>
1.0.0	First release

#### Firmware and hardware versions

Documentation, version	Firmware version	Hardware version
2.0.0	1F	02
1.2.0	1E	01
1.1.0	1E	00
1.0.0	1B	00

The firmware and hardware versions (delivery state) can be taken from the serial number printed on the side of the terminal module.

#### Syntax of the serial number

Structure of the serial number: WW YY FF HH

- WW - week of production (calendar week)
- YY - year of production
- FF - firmware version
- HH - hardware version

Example with serial number 35 05 00 01:

- 35 - week of production 35
- 05 - year of production 2005
- 00 - firmware version 00
- 01 - hardware version 01

## 2 Product overview

### 2.1 Introduction

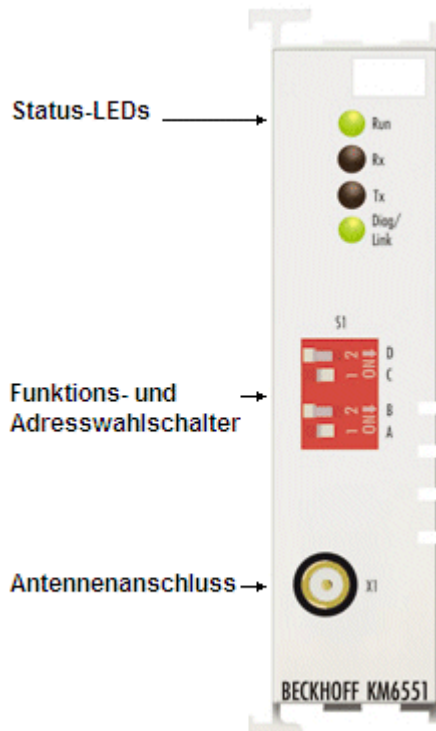


Fig. 1: KM6551-0000 - Terminal module for radio transmission

The KM6551-0000 terminal module is a data exchange unit based on radio technology. It uses the IEEE 802.15.4 standard. Data is exchanged or transmitted between two independent controllers via radio, independent of the higher-level fieldbus. The free-field distance between two KM6551-0000 units can be up to 300 m.

The KM6551-0000 terminal module has a reverse SMA plug (Straight Medium Adapter), to which various [radio antennas](#) [► 27] can be connected, which are to be procured from Beckhoff. The directional characteristic can be adapted to the surroundings by means of specifically selecting the antenna. Status and data exchange are displayed via LEDs, thereby offering fast and simple diagnostics. A [TwinCAT library](#) [► 44] is available for the use of the KM6551-0000 terminal module with TwinCAT.



## 2.2 Technical data

Technical data	KM6551-0000
Frequency band	2.4 GHz
Data transfer rates	250 kbit
Output power	0 dBm (1 mW)
Reception sensitivity	-87 dBm
Protocol	IEEE 802.15.4
Antenna connection	reverse SMA plug (RP_SMA)
Power supply for the electronics	via the K-bus
Current consumption via K-bus	typically 135 mA
Width of a bus terminal block	Maximum 64 standard Bus Terminals or 80 cm (one KM6551-0000 corresponds to 2 standard Bus Terminals here)
Data width in the input process image	12 bytes
Data width in the output process image	12 bytes
Dimensions without antenna (W x H x D)	approx. 26.5 mm x 100 mm x 55 mm (width aligned: 24 mm)
Weight	app. 100 g
Permissible ambient temperature range during operation	0°C ... + 55°C
Permissible ambient temperature range during storage	-25°C ... + 85°C
Permissible relative air humidity	95%, no condensation
<a href="#">Mounting</a> [ <a href="#">▶ 17</a> ]	on a 35 mm <a href="#">mounting rail</a> [ <a href="#">▶ 17</a> ] (e.g. DIN rail TH 35-7.5 conforming to EN 60715)
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, EN 60068-2-29
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP20
Installation position	variable
Approval	CE

## 2.3 Basic Function Principles

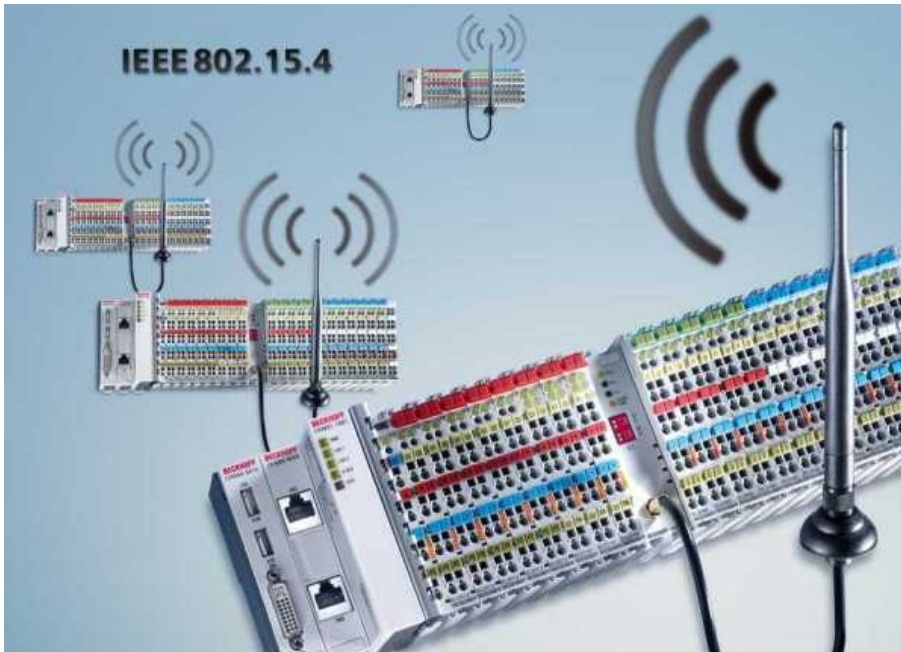


Fig. 2: Basic Function Principles

The KM6551-0000 data exchange module enables the wireless exchange of data between two or more controllers. It uses the IEEE 802.15.4 standard as its basis with a Beckhoff-specific protocol. 10 bytes of user data are transmitted per data packet. The DIP switch is used to set the operating mode of the KM6551-0000, i.e. whether the module functions as a master or slave and which communication mode is to be used.

The data is exchanged in the peer to peer and master-slave modes using the polling method. In broadcast mode, one module is the broadcast master that sends the data and all other modules are broadcast slaves that receive the data but cannot send data to the broadcast master themselves. Hence, they listen only to data from the broadcast master.

In master-slave mode you can decide via the software with which slave data should be exchanged. Up to 7 slaves can be addressed.

The data exchange module KM6551-0000 supports 16 channels, which are freely selectable and can be used, for example, to establish several radio networks or for placement outside WLANs or other radio systems that also use 2.4 GHz.

The KM6551-0000 can scan the possible 16 channels. The energy in the frequency range is measured to ascertain in advance whether other systems are active and on which frequencies. Furthermore, the so-called LQI (Link of Quality Index) is transmitted with each data telegram. This makes it possible to determine the quality of the signal. A high LQI value indicates a good connection, a low value a poor connection. In order to improve the LQI value, a larger antenna or an antenna with a correspondingly larger transmission factor can be used.

### Communication mode

The KM6551-0000 data exchange module supports three different communication modes.

#### Mode 1: Peer to peer – data exchange between two modules

Enables the exchange of data between two KM6551-0000. A maximum of 10 bytes of data can be transmitted per cycle. In one cycle (typically < 20 ms), module 1 sends data to module 2 and module 2 sends data back to module 1.

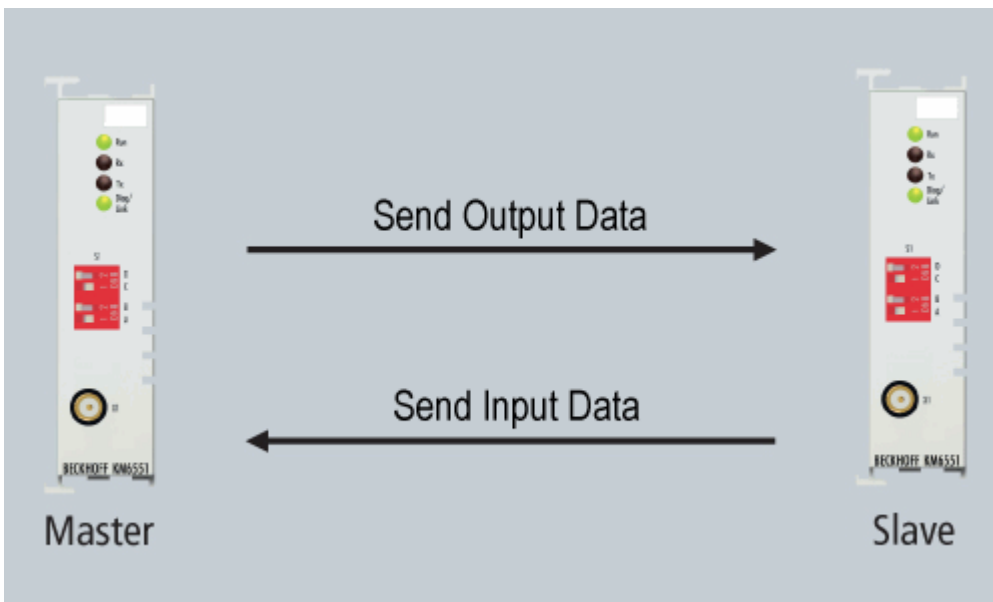


Fig. 3: Peer to Peer

**Mode 2: Master-slave – data exchange between a master and up to 7 slaves**

In master-slave mode the master can communicate with up to seven KM6551-0000 using the polling method. To do this, set the corresponding slave addresses using the DIP switch. From the PLC you can inform the master which slave it should communicate with, how often and for how long. Approx. 20 ms are required per slave. For seven slaves this results in a minimum cycle time of 140 ms which the master requires in order to address all 7 slaves once each.

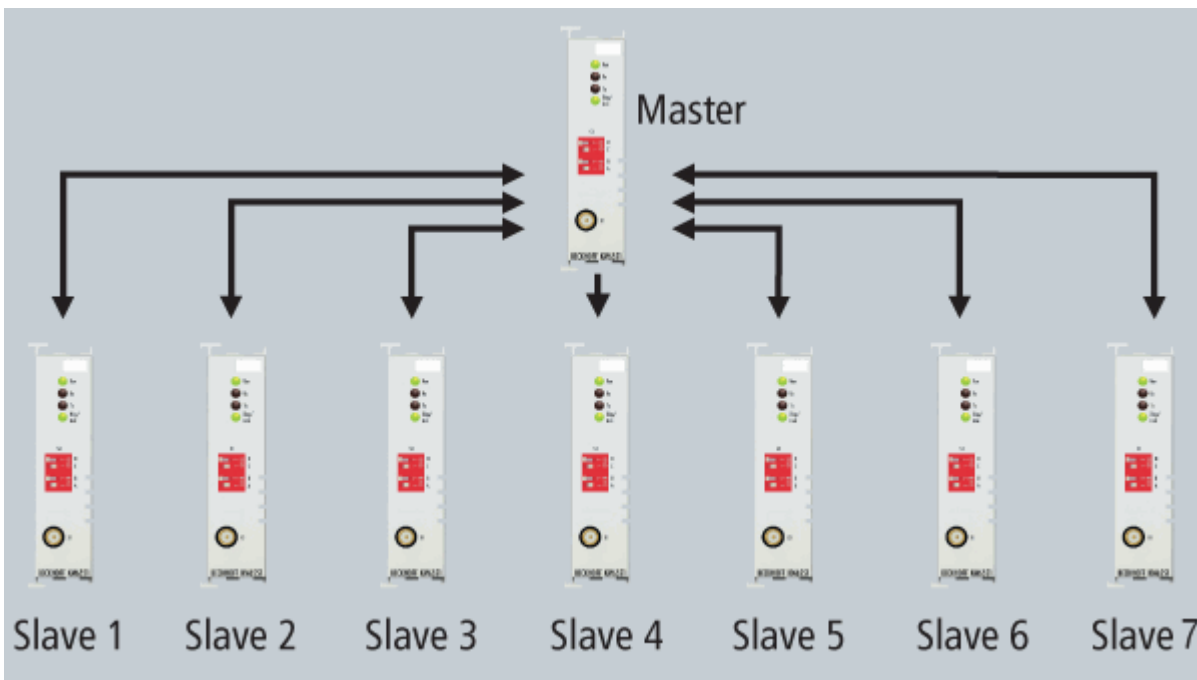


Fig. 4: Master-Slave mode

**Mode 3: Broadcast to any number of slaves**

In broadcast mode, only the broadcast master transmits. All other modules (broadcast slaves) can only receive data but cannot send data themselves. Any number (x) of broadcast slaves can listen in.

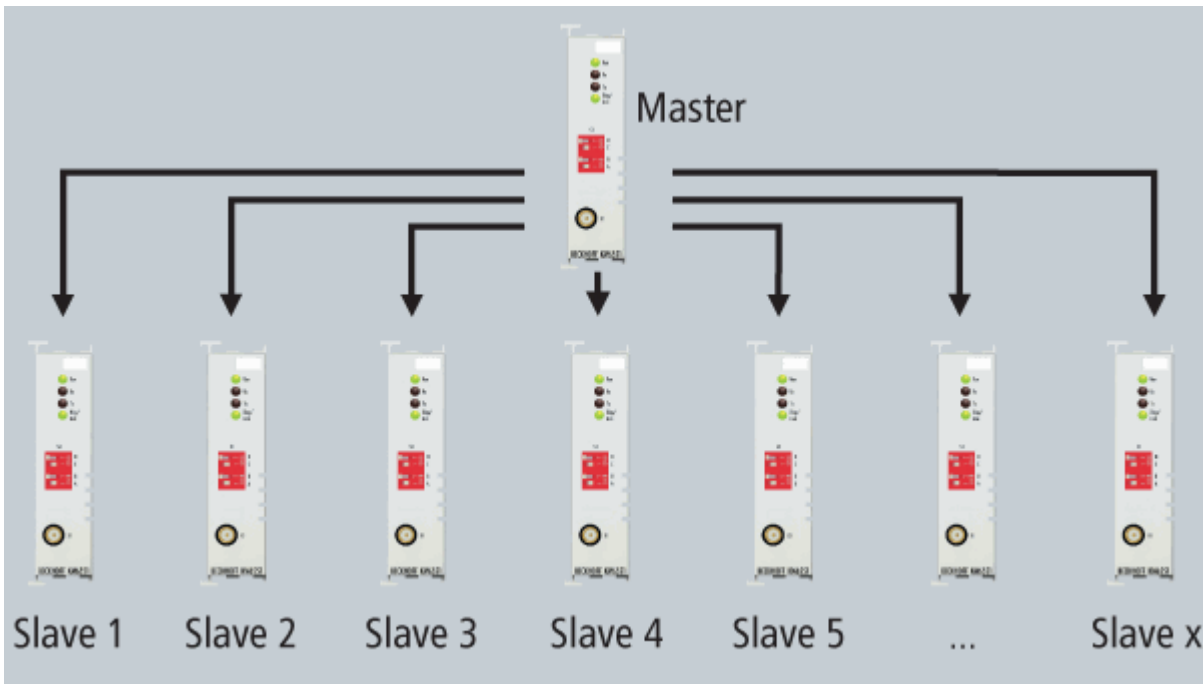


Fig. 5: Broadcast mode

**i**

**Support of the KM6551-0000 using Bus Couplers, Bus Terminal Controllers and TwinCAT**

The KM6551-0000 is supported from TwinCAT 2.10 Build 1326 onwards. The following Bus Couplers are supported: BK1120, BK1250, BK2020, BK3120, BK3150, BK9000, BK9050. (Further Bus Couplers on request). All Bus Terminal Controllers from the BCxxxx, BXxxxx and BXxxxx series are supported.

**NOTE**

**CE conformity**

The CE conformity of the KM6551-0000 is only guaranteed if it is operated with original Beckhoff accessories ([antennas | 27](#)), [coaxial cable | 19](#))!

**2.4 LED displays**

**KM6551-0000**

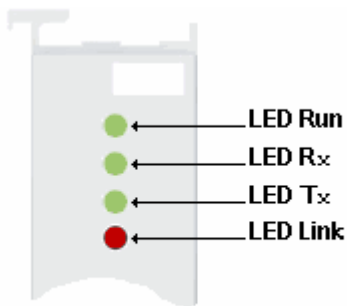


Fig. 6: KM6551 - LED displays

LED	Display	
Run (green)	off	Data transmission on the K-bus is not active
	on	Data transmission on the K-bus is active
Rx (green)	on	Data being received via radio
Tx (green)	on	Data being sent via radio
Link (green, orange, red)	on	Green - good signal quality Orange - moderate signal quality Red - poor signal quality or watchdog has triggered

## 2.5 DIP switch

You can activate the different modes of the KM6551-0000 using the DIP switch. This enables the simple exchange of the modules without additional configuration software.

- DIP switch in right position: ON
- DIP switch in left position: OFF

The picture illustrates the setting for *Slave 5*.

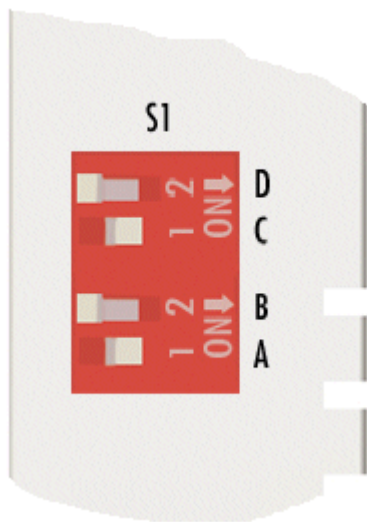


Fig. 7: DIP switch

DIP switch	A	B	C	D
Master mode	OFF	OFF	OFF	OFF
Slave 1	<b>ON</b>	OFF	OFF	OFF
Slave 2	OFF	<b>ON</b>	OFF	OFF
Slave 3	<b>ON</b>	<b>ON</b>	OFF	OFF
Slave 4	OFF	OFF	<b>ON</b>	OFF
Slave 5	<b>ON</b>	OFF	<b>ON</b>	OFF
Slave 6	OFF	<b>ON</b>	<b>ON</b>	OFF
Slave 7	<b>ON</b>	<b>ON</b>	<b>ON</b>	OFF
Broadcast slave	OFF	OFF	OFF	<b>ON</b>

## 3 IEEE802.15.4

### 3.1 Introduction

The terms IEEE 802.15.4 and ZigBee are used in many places as synonyms, although there is a clear demarcation between them, which will be briefly explained at this point.

The 802.15.4 standard, which was elaborated by the Institute of Electrical and Electronics Engineers (IEEE), specifies the Physical Layer (PHY) and the Medium Access Control (MAC), which correspond to the two lowest levels of the OSI layer model. The IEEE 802.15.4 standard was ratified at the beginning of May 2004 [1]. Therefore, apart from a few expected amendments and clarifications, work on it is deemed to be complete.

The ZigBee Alliance [2] was founded by several large firms from the semiconductor industry with the aim of developing a complete protocol suite on the basis of IEEE 802.15.4 for wireless communication extending up to the application interface. However, it is worth mentioning in this respect that the IEEE 802.15.4 standard is in no way linked to the ZigBee Alliance.

#### ● No ZigBee!



The KM6551-0000 data transmission module is based on IEEE 802.15.4, but it is not a ZigBee product and is also not ZigBee-compatible!

Technical data	KM6551
Data transmission band	2.4 GHz
Channels	16
Channel separation	5 MHz
Channel width	2 MHz
Available	Worldwide
Data transfer rate	250 kbit
Protocol	IEEE 802.15.4

16 channels, each with a gross data rate of 250 kB/s, are available in the worldwide available 2.4 GHz band.

### 2.4 GHz PHY

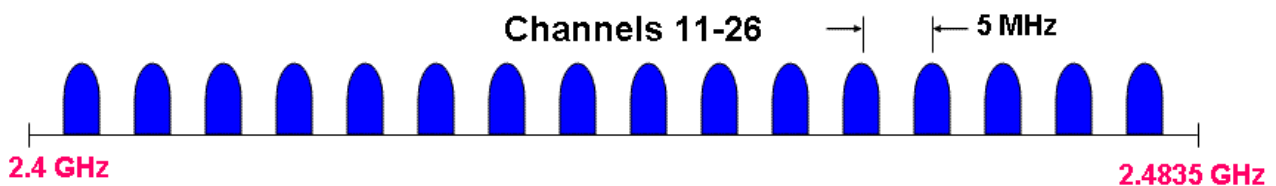


Fig. 8: Channels 11 to 26

[1] Institute of Electrical and Electronics Engineers (Ed.): IEEE Standard for Information technology -- Telecommunication and information exchange between systems -- Local and metropolitan area networks -- Specific requirements. Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Computer Society, New York, NY, USA, October 2003

[2] ZigBee Alliance, <http://www.zigbee.org>

### 3.2 Interference caused by other radio systems



**Check frequency ranges**

- WLAN networks on adjacent or the same channels
- Microwave ovens

**WLAN**

If the IEEE 802.15.4 channel used by the KM6551-0000 and the frequency range of a neighboring WLAN network overlap, this can lead to disruptions in the KM6551-0000 communication.

Select an IEEE 802.15.4 channel for the KM6551-0000 that uses the gaps between neighboring WLAN networks as shown in the figure below.

Even if the maximum possible three non-overlapping WLAN channels are used adjacently, four IEEE 802.15.4 channels remain that the KM6551-0000 can use without interference.

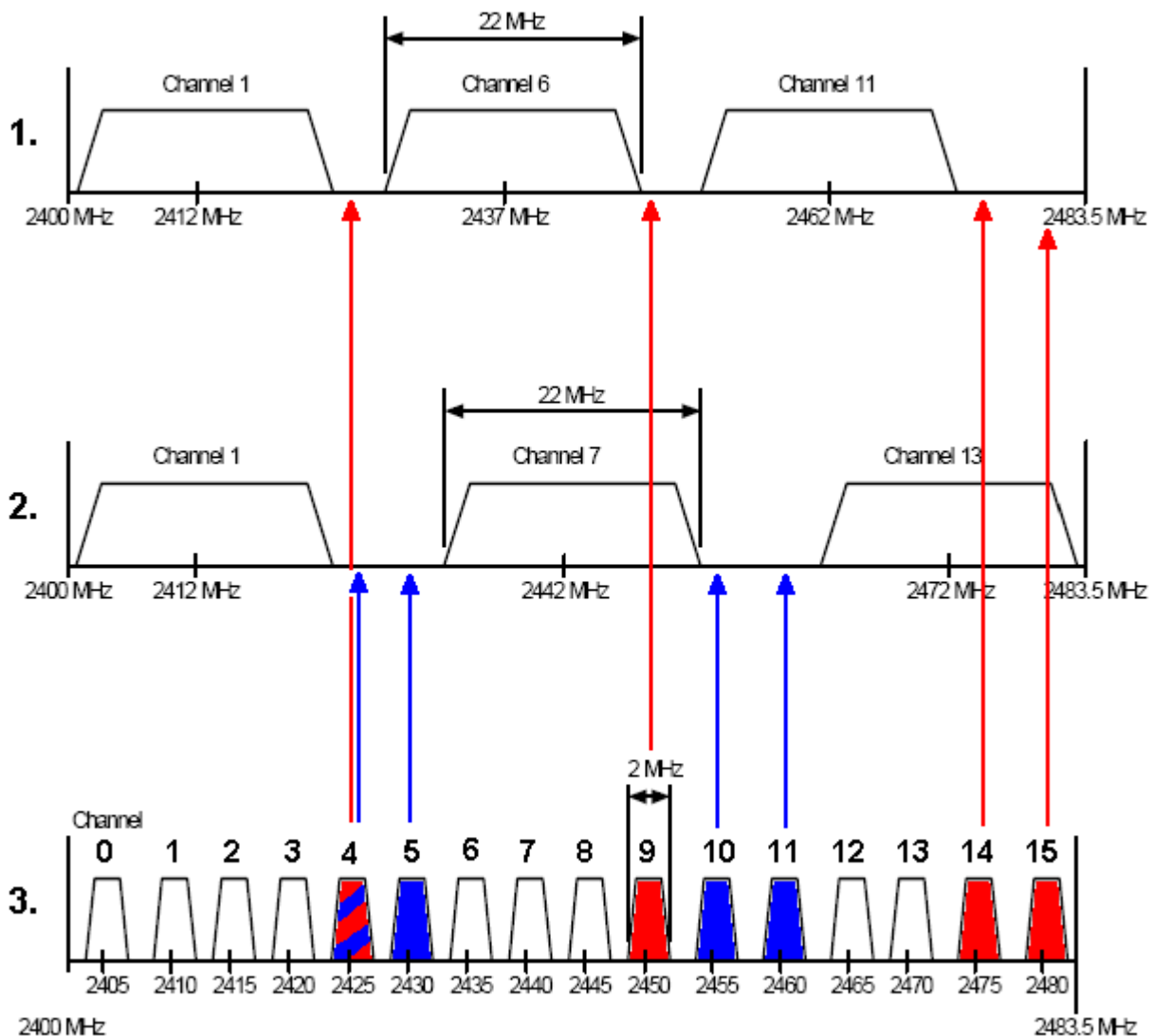


Fig. 9: Utilizing gaps between adjacent WLAN networks

1. Three non-overlapping WLAN networks in the WLAN channels permitted in North America (IEEE 802.11b)
2. Three non-overlapping WLAN networks in the WLAN channels permitted in Europe (IEEE 802.11b)
3. Placement of IEEE802.15.4 channels (2400 MHz PHY) in the gaps between WLAN networks

**Microwave ovens**

Since microwave ovens typically operate at a frequency of 2.455 GHz, neighboring poorly screened ovens can interfere with the transmission between the KM6551-0000.

In this case, remove the interfering devices or use only well-screened microwave ovens in the direct vicinity of the KM6551-0000.



## 4 Mounting and wiring

### 4.1 Recommended mounting rails

Terminal Modules und EtherCAT Modules of KMxxxx and EMxxxx series, same as the terminals of the EL66xx and EL67xx series can be snapped onto the following recommended mounting rails:

- DIN Rail TH 35-7.5 with 1 mm material thickness (according to EN 60715)
- DIN Rail TH 35-15 with 1,5 mm material thickness

#### ● Pay attention to the material thickness of the DIN Rail

**i** Terminal Modules und EtherCAT Modules of KMxxxx and EMxxxx series, same as the terminals of the EL66xx and EL67xx series does not fit to the DIN Rail TH 35-15 with 2,2 to 2,5 mm material thickness (according to EN 60715)!

### 4.2 Mounting and demounting - terminals with traction lever unlocking

The terminal modules are fastened to the assembly surface with the aid of a 35 mm mounting rail (e.g. mounting rail TH 35-15).

#### ● Fixing of mounting rails

**i** The locking mechanism of the terminals and couplers extends to the profile of the mounting rail. At the installation, the locking mechanism of the components must not come into conflict with the fixing bolts of the mounting rail. To mount the recommended mounting rails under the terminals and couplers, you should use flat mounting connections (e.g. countersunk screws or blind rivets).

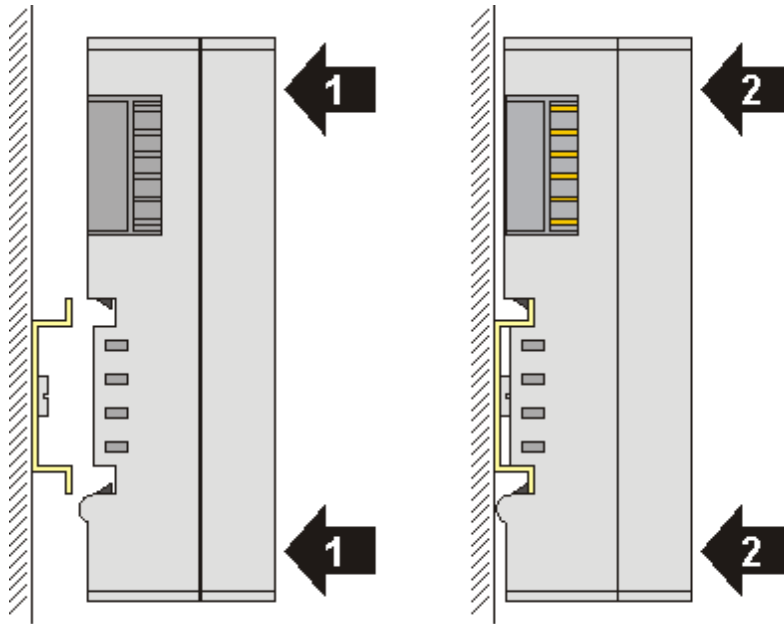
#### ⚠ WARNING

##### **Risk of electric shock and damage of device!**

Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the Bus Terminals!

#### **Mounting**

- Fit the mounting rail to the planned assembly location.

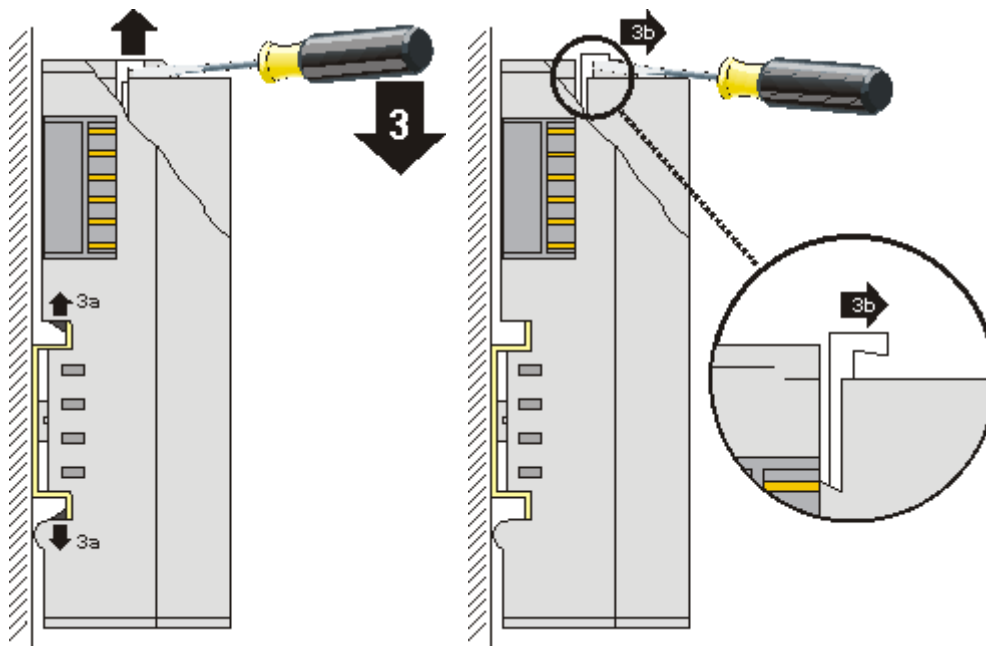


and press (1) the terminal module against the mounting rail until it latches in place on the mounting rail (2).

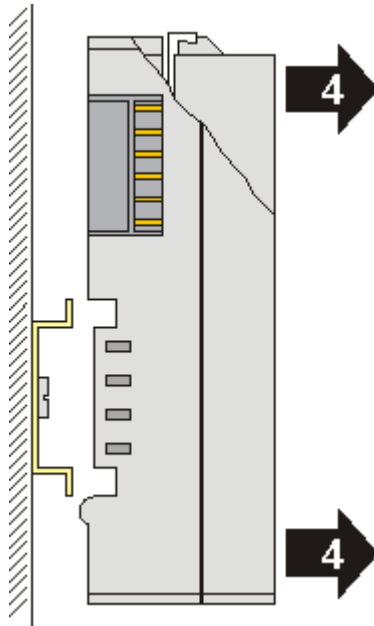
- Attach the cables.

**Demounting**

- Remove all the cables. Thanks to the KM/EM connector, it is not necessary to remove all the cables separately for this, but for each KM/EM connector simply undo 2 screws so that you can pull them off (fixed wiring)!
- Lever the unlatching hook on the left-hand side of the terminal module upwards with a screwdriver (3). As you do this
  - an internal mechanism pulls the two latching lugs (3a) from the top hat rail back into the terminal module,
  - the unlatching hook moves forwards (3b) and engages



- In the case 32 and 64 channel terminal modules (KMxxx4 and KMxxx8 or EMxxx4 and EMxxx8) you now lever the second unlatching hook on the right-hand side of the terminal module upwards in the same way.
- Pull (4) the terminal module away from the mounting surface.



### 4.3 Dimensions

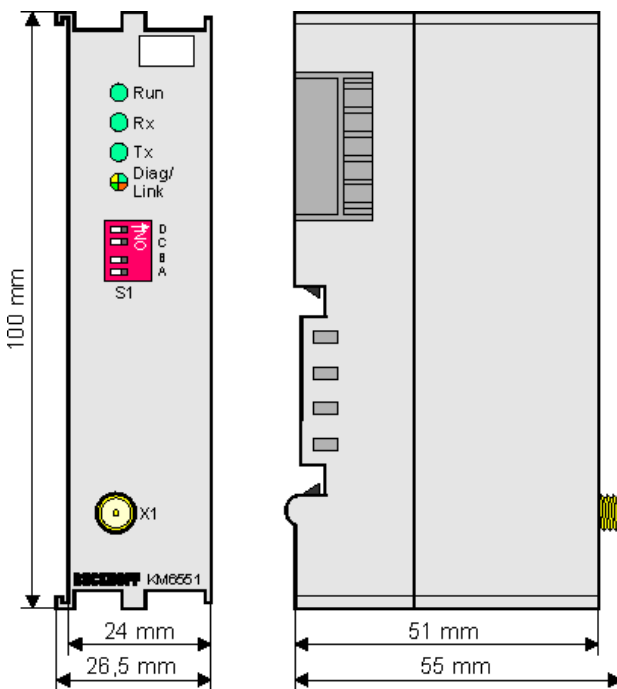


Fig. 10: KM6551 dimensions

### 4.4 Connection

The antennas are connected via a reverse SMA screw plug. Please screw the cable, the coaxial cable or the antenna hand tight to this screw plug.

**Coaxial cable**

<b>Name</b>	<b>Description</b>
ZK6000-0102-0020	Coaxial cable, characteristic impedance 50 $\Omega$ , preassembled plug connectors (SMA plug and reverse SMA socket), black, 2 m
ZK6000-0102-0040	Coaxial cable, characteristic impedance 50 $\Omega$ , preassembled plug connectors (SMA plug and reverse SMA socket), black, 4 m

## 4.5 Antenna alignment

### 4.5.1 Directional characteristic

Please pay attention to the directional characteristics and polarization of the antennas in order to mount and align them to each another optimally!

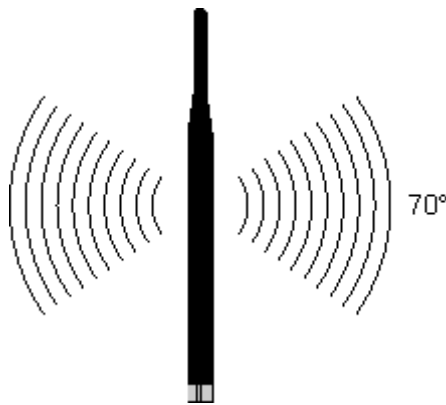
#### Omnidirectional antennas

##### ZS6201-0410, ZS6201-0500

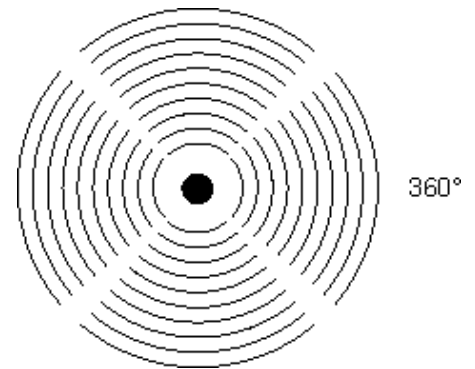
Design



Side view  
(vertical directional characteristic)



Top view  
(horizontal directional characteristic)



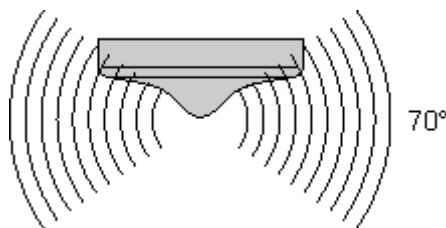
##### ZS6200-0400

Predestined for mounting below the ceiling.

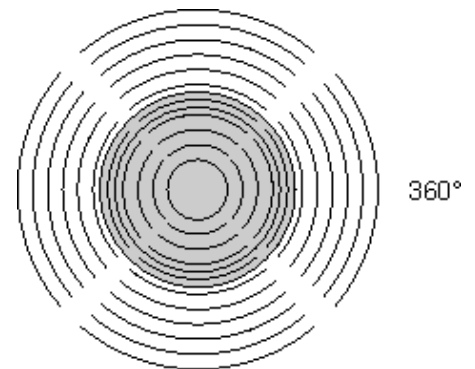
Design



Side view  
(vertical directional characteristic)



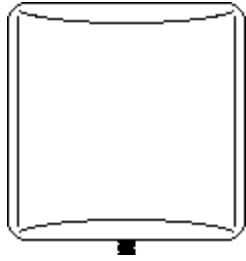
Top view  
(horizontal directional characteristic)



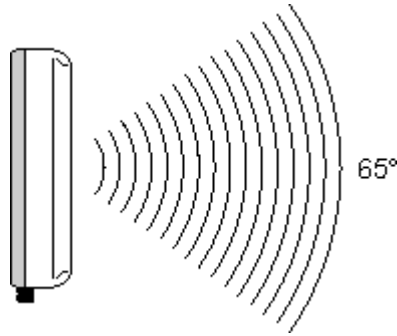
**Directional antennas**

**ZS6100-0900**

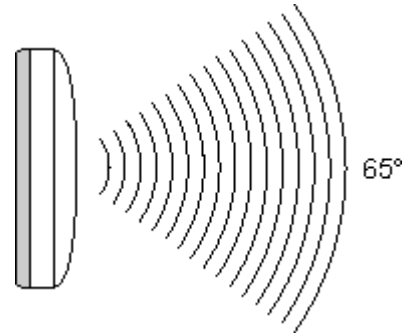
Design



Side view  
(vertical directional characteristic)

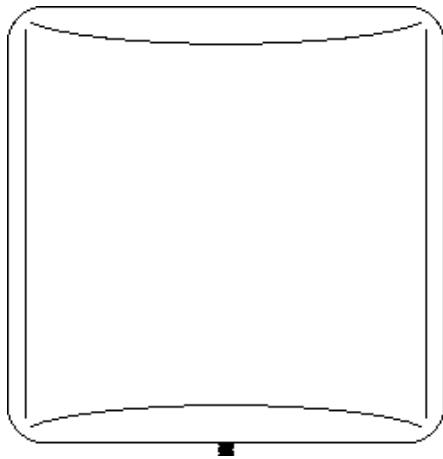


Top view  
(horizontal directional characteristic)

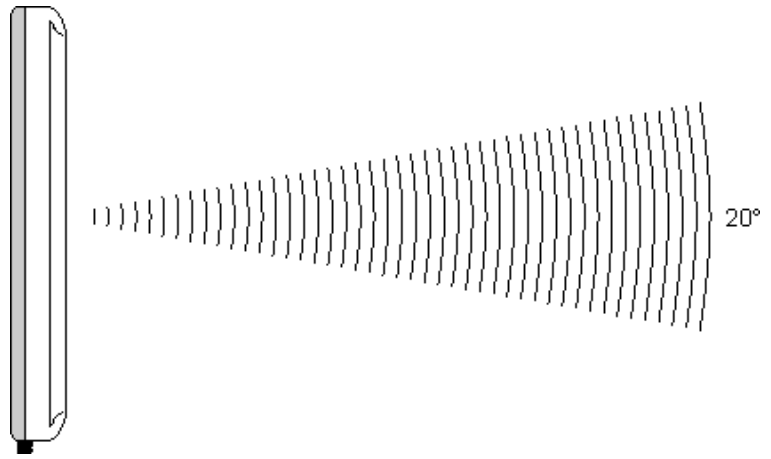


**ZS6100-1800**

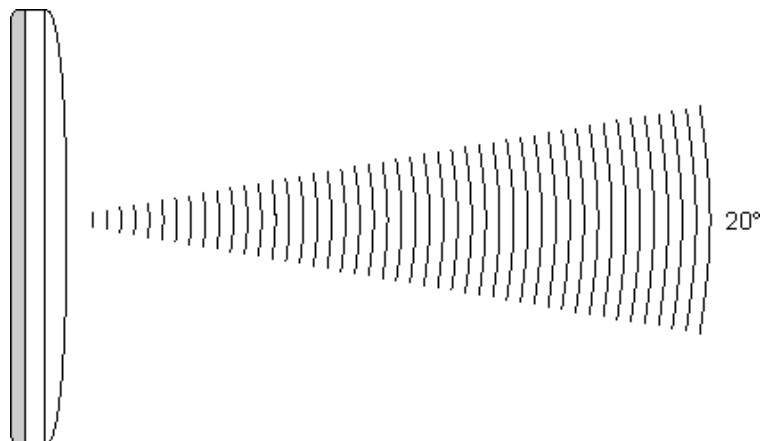
Design



Side view (vertical directional characteristic)



Top view (horizontal directional characteristic)



## 4.5.2 Alignment examples

Align the antennas so that each lies within the radiation cone of the opposite antenna.

### Omnidirectional antennas

Two ZS6201-0410 or ZS6201-0500



Fig. 11: Omnidirectional antennas

### Directional antennas

Two ZS6100-0900 or ZS6100-1800



Fig. 12: Directional antennas

### Mixed operation

e.g. one ZS6201-0410 and two ZS6100-1800

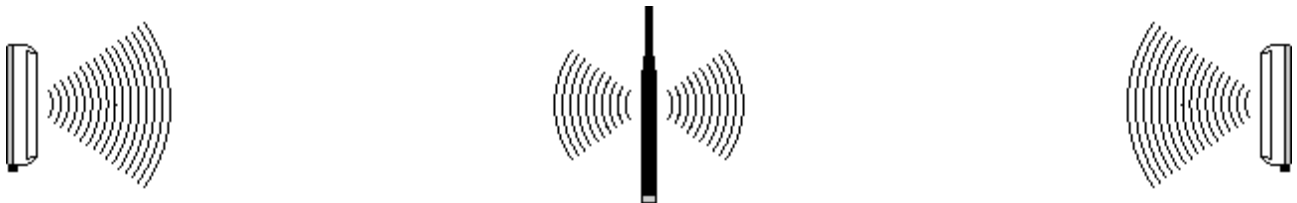


Fig. 13: Mixed operation

## 4.5.3 Polarization

For optimum transmission, the antennas of all of the KM6551-0000 used must have the same polarization.

### Omnidirectional antennas

Care must also be taken when using omnidirectional antennas that the antennas of all of the KM6551-0000 used have the same polarization.

Omnidirectional antennas such as the ZS6201-0410, ZS6201-0500 or ZS6200-0400 are mostly mounted for vertical polarization.

### Directional antennas

Arrows marked with the letters H and V are located on the rear side of the housing of the ZS6100-0900 directional antenna in order to identify the polarization (ZS6100-1800 in preparation).

Mount the directional antennas such that the marked arrows of all the antennas used correspond to one another.

#### 4.5.4 Placement of the antennas

Mount the antennas such that they can radiate freely!

There must be no obstructions in the direct vicinity of the antenna that could hinder the development of the Fresnel zone [► 25]. Metal obstacles such as control cabinets, machine parts, pipelines, iron beams etc. particularly hinder the development of the Fresnel zone!

The connection of the antennas [► 27] to the KM6551-0000 via the RSMA plug and coaxial cable [► 19] enables the antenna to be mounted remotely, so that you can position the antenna optimally.



## 4.6 Attenuation and range

### 4.6.1 Fresnel zone

In radio transmission, the space between the transmitting and receiving antennas is known as the Fresnel zone. The Fresnel zone is a notional spheroid between the antennas.

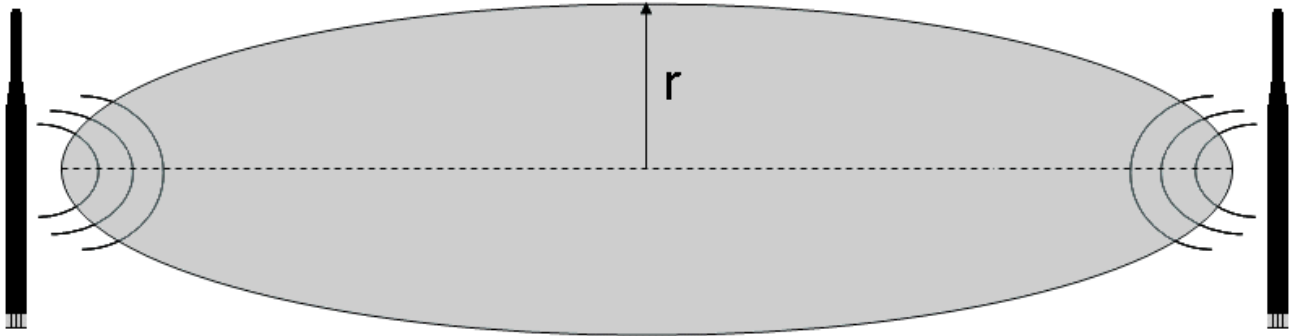


Fig. 14: Fresnel zone

The main portion of the energy is transmitted in the area of the Fresnel zone.

This zone should be free of obstructions (e.g. objects, houses, trees etc.). Metal obstacles such as control cabinets, machine parts, pipelines, iron beams etc. particularly hinder the development of the Fresnel zone!

Each hindrance of the Fresnel zone attenuates the transmission. If the Fresnel zone is half obscured, for example, the additional attenuation is 6 dB, i.e. the field strength is reduced to half of the free field value. Reception may then be disturbed or completely interrupted under certain circumstances.

If the Fresnel zone is free from obstructions, the propagating wave is only attenuated by the free space attenuation.

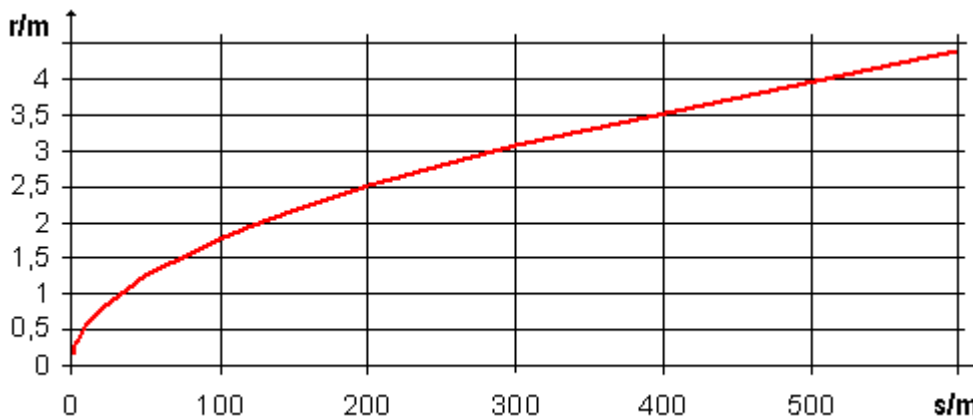


Fig. 15: Radius  $r$  of the Fresnel zone in relationship to the distance  $s$

### 4.6.2 Attenuation in practice

With an attenuation of 6 dB the range is shortened to half of the value for an unobstructed connection, with 12 dB it is shortened to a quarter.

Material	Attenuation	Range approx.	Example for an unobstructed range of 280 m
Thin wall	2 ... 5 dB	(free field range)/1.5 - (free field range)/2	180 m ... 140 m
Wooden wall	5 dB	(free field range)/2	140 m
Masonry wall	6 ... 12 dB	(free field range)/2 - (free field range)/4	140 m ... 70 m
Concrete wall	10 ... 20 dB	(free field range)/4 - (free field range)/8	70 m ... 5 m
Concrete ceiling	20 dB	(free field range)/8	< 35 m

### 4.6.3 Range of different antenna combinations

The given ranges are based on an unobstructed view and adherence to the Fresnel zone.

#### Two omnidirectional antennas

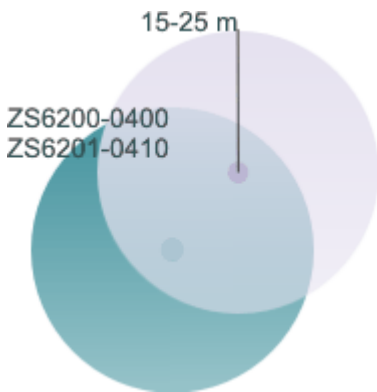


Fig. 16: Two omnidirectional antennas

#### Omnidirectional antenna combined with a directional antenna

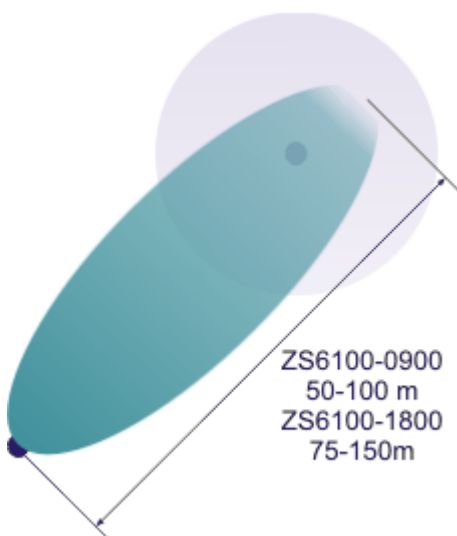


Fig. 17: Omnidirectional antenna combined with a directional antenna

**Two directional antennas**

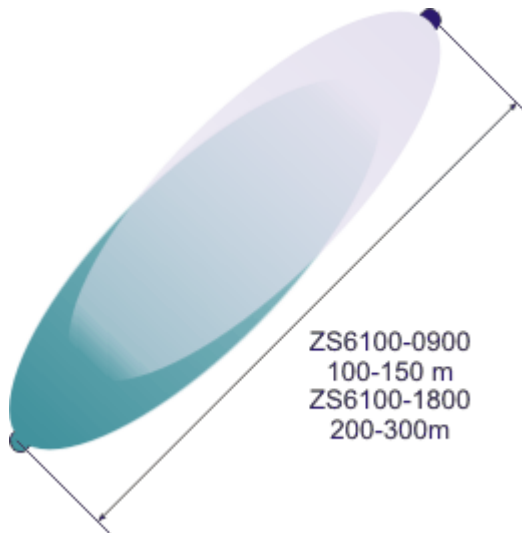


Fig. 18: Two directional antennas

## 4.7 Antennas

**Overview**

Name	Description
<a href="#">ZS6100-0900 [▶ 28]</a>	Directional antenna (gain 9 dBi), without cable
<a href="#">ZS6100-1800 [▶ 30]</a>	Directional antenna (gain 18 dBi), without cable
<a href="#">ZS6200-0400 [▶ 32]</a>	Omnidirectional antenna (gain 4 dBi), without cable
<a href="#">ZS6201-0410 [▶ 34]</a>	Rod antenna (gain 4 dBi), with cable (1 m)
<a href="#">ZS6201-0500 [▶ 36]</a>	Rod antenna (gain 5 dBi), without cable

**NOTE**

**CE conformity**

The CE conformity of the KM6551-0000 is only guaranteed if it is operated with original Beckhoff accessories (antennas, [coaxial cable \[▶ 19\]](#))!

### 4.7.1 ZS6100-0900



Fig. 19: ZS6100-0900 - Directional antenna

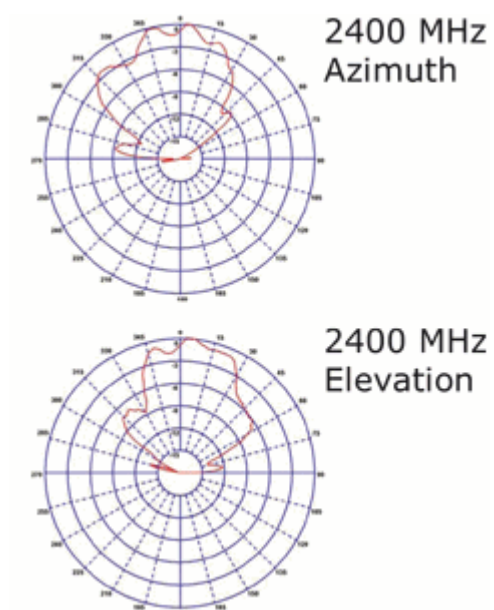


Fig. 20: ZS6100-0900 - Azimuth and Elevation for 2400 MHz

**Technical data**

<b>Technical data</b>	<b>ZS6100-0900</b>
Frequency range	2400...2485 MHz
Transmission factor	9 dBi
3 dB bandwidth, horizontal	65°
3 dB bandwidth, vertical	65°
Connection	SMA socket
Dimensions (W x H x D)	93 mm x 93 mm x 25 mm
Weight (incl. accessories and packaging)	approx. 190 g
Permissible ambient temperature range during operation	-40°C ... + 80°C
Permissible relative air humidity	95%, no condensation
Protection class	IP20
Installation position	variable
Approval	CE
Mounting	Bracket mounting, included in scope of supply
Suitable coaxial cable	ZS6000-0102-0020, ZS6000-0102-0040

**4.7.2 ZS6100-1800**



Fig. 21: ZS6100-1800 - Directional antenna with large gain

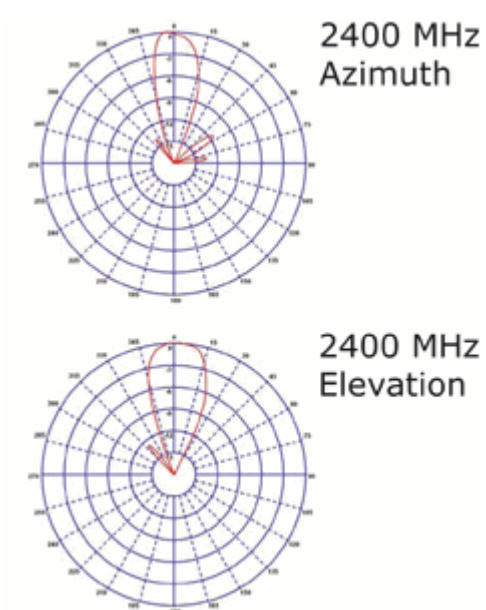


Fig. 22: ZS6100-1800 - Azimuth and Elevation for 2400 MHz

**Technical data**

<b>Technical data</b>	<b>ZS6100-1800</b>
Frequency range	2400...2485 MHz
Transmission factor	18 dBi
3 dB bandwidth, horizontal	20°
3 dB bandwidth, vertical	20°
Connection	SMA socket
Dimensions (W x H x D)	360 mm x 360 mm x 30 mm
Weight (incl. accessories and packaging)	approx. 3640 g
Permissible ambient temperature range during operation	-40°C ... + 80°C
Permissible relative air humidity	95%, no condensation
Protection class	IP20
Installation position	variable
Approval	CE
Mounting	Bracket mounting, included in scope of supply
Suitable coaxial cable	ZS6000-0102-0020, ZS6000-0102-0040

### 4.7.3 ZS6200-0400



Fig. 23: ZS6200-0400 - Omnidirectional antenna

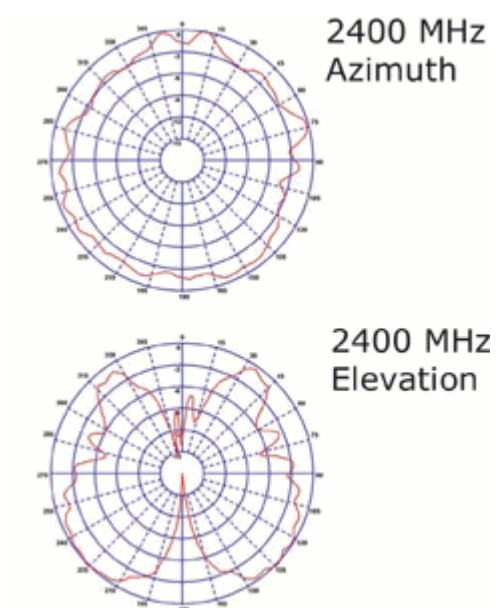


Fig. 24: ZS6200-0400 - Azimuth and Elevation for 2400 MHz



**Technical data**

<b>Technical data</b>	<b>ZS6200-0400</b>
Frequency range	2400...2485 MHz
Transmission factor	4 dBi
3 dB bandwidth, horizontal	360°
3 dB bandwidth, vertical	70°
Connection	SMA socket
Dimensions	Height: 110 mm, diameter: 45 mm
Weight (incl. accessories and packaging)	approx. 210 g
Permissible ambient temperature range during operation	-40°C ... + 80°C
Permissible relative air humidity	95%, no condensation
Protection class	IP20
Installation position	variable, predestined for mounting below the ceiling.
Approval	CE
Suitable coaxial cable	ZS6000-0102-0020, ZS6000-0102-0040

**4.7.4 ZS6201-0410**



Fig. 25: ZS6201-0410 - Rod antenna with cable

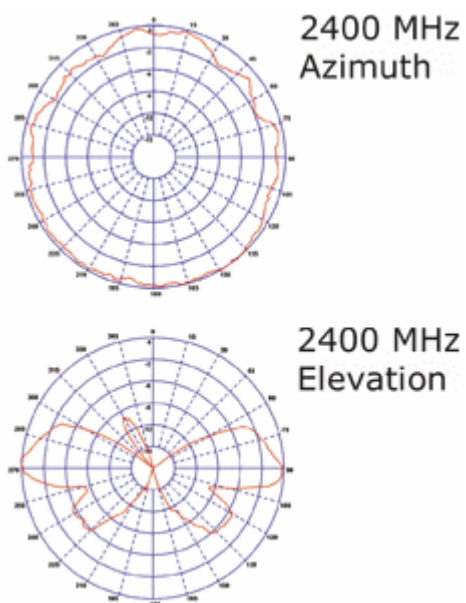


Fig. 26: ZS6201-0410 - Azimuth and Elevation for 2400 MHz

**Technical data**

<b>Technical data</b>	<b>ZS6201-0410</b>
Frequency range	2400...2485 MHz
Transmission factor	4 dBi
3 dB bandwidth, horizontal	360°
3 dB bandwidth, vertical	70°
Connection	Reverse SMA socket (with 1 m cable, permanently connected to antenna)
Dimensions	Height 202 mm, foot diameter 35 mm
Weight (incl. cable, accessories and packaging)	approx. 220 g
Permissible ambient temperature range during operation	-40°C ... + 80°C
Permissible relative air humidity	95%, no condensation
Mounting	Cap nut M14
Protection class	IP20
Installation position	variable
Approval	CE
Coaxial cable	1 m, included in scope of supply

**4.7.5 ZS6201-0500**

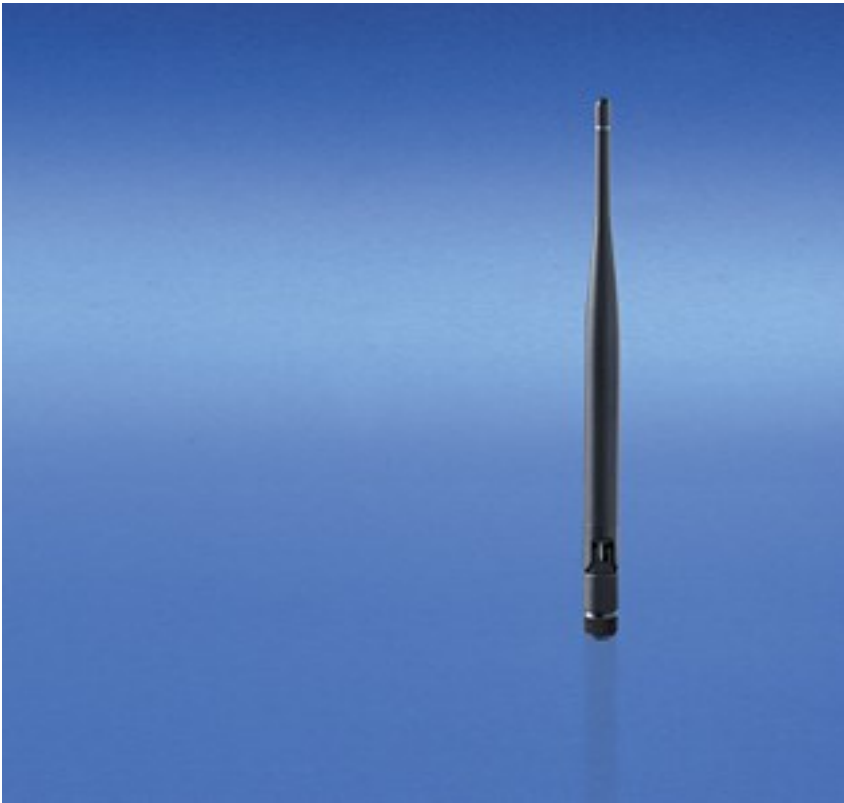


Fig. 27: ZS6201-0500 - Rod antenna

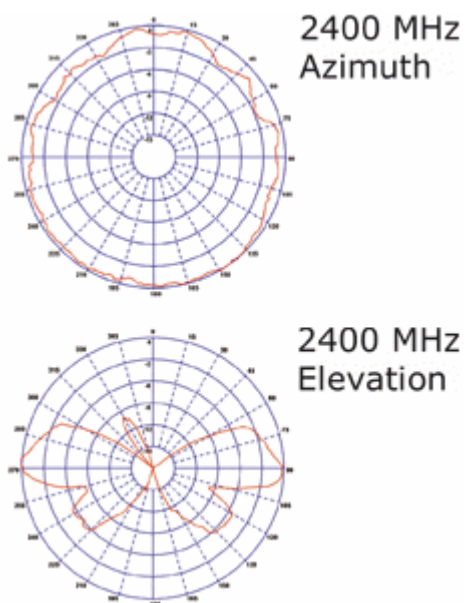


Fig. 28: ZS6201-0500 - Azimuth and Elevation for 2400 MHz

**Technical data**

<b>Technical data</b>	<b>ZS6201-0500</b>
Frequency range	2400...2485 MHz
Transmission factor	5 dBi
3 dB bandwidth, horizontal	360°
3 dB bandwidth, vertical	70°
Connection	Reverse SMA socket
Dimensions	Height 195 mm, foot diameter 12 mm
Weight (incl. packaging)	approx. 40 g
Permissible ambient temperature range during operation	-40°C ... + 80°C
Permissible relative air humidity	95%, no condensation
Mounting	Direct connection with hinged joint
Protection class	IP20
Installation position	variable
Approval	CE
Suitable coaxial cable	Not required, direct connection

## 5 Application examples - overview

- [Peer to peer mode \[► 38\]](#)
- [Master-Slave mode \[► 38\]](#)
- [Broadcast mode \[► 39\]](#)
- [Energy scan \[► 40\]](#)

### 5.1 Peer to peer mode

#### Application

The simplest type of communication between two KM6551-0000 is the peer to peer mode. To do this, set the DIP switches [\[► 13\]](#)

- of the master module to master mode and
- that of the second module to slave address 1.

Now you have to set bit 0 (start bit) in the control byte [CB1 \[► 59\]](#) of both modules to TRUE. The modules acknowledge this bit by setting bit 0 of their status bytes [SB1 \[► 59\]](#). The establishment of communication and the communication itself then begin automatically.

Now 10 bytes are always exchanged between the modules. This continues until a start bit is reset to FALSE. It does not matter which side resets the start bit.

If the connection is disturbed or if the start bit is not set on the opposite side, the module reports an error with bit 6 in the status byte [SB1 \[► 59\]](#). The error code is output at the same time in status byte [SB2 \[► 59\]](#) (see ErrorID).

### 5.2 Master-Slave mode

#### Application

A further possibility is the exchange of data between a master and up to 7 slaves. The address of the slave is set accordingly with the [DIP switch \[► 13\]](#). The address may only be used once for each channel. Here too the control byte 1 [CB1 \[► 59\]](#) must be set to TRUE as bit 0 "Start BIT". Bits 3...5 are used to set the slave address of the target device. The terminal then sends telegrams to the corresponding slave. If the terminal receives a response, this is displayed in the status.

*Example: Task – you want to speak to slave 2.*

Set bit 0 to TRUE in order to start data exchange. Bit 3 must then be set in order to speak to slave 2. As soon as the reply from slave 2 is received (after approx. 20 ms), the master terminal confirms this in the status byte by setting bit 4. The address is always counted +1 in the status byte (see table).

#### CB1 [► 59]

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
Name	RegAccess	reserve	Add3	Add2	<b>Add1</b>	Scan	EnergyScan	<b>Start</b>
Value	0	0	0	0	1	0	0	1

#### SB1 [► 59]

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
Name	RegAccess	Error	Add3	<b>Add2</b>	Add1	Scan	EnergyScan	<b>Start</b>
Value	0	0	0	1	0	0	0	1

Address	Control byte bit 3...5		Status byte bit 3...5	
Slave 1	0 <sub>dec</sub>	000 <sub>bin</sub>	1 <sub>dec</sub>	001 <sub>bin</sub>
Slave 2	1 <sub>dec</sub>	001 <sub>bin</sub>	2 <sub>dec</sub>	010 <sub>bin</sub>
Slave 3	2 <sub>dec</sub>	010 <sub>bin</sub>	3 <sub>dec</sub>	011 <sub>bin</sub>
Slave 4	3 <sub>dec</sub>	011 <sub>bin</sub>	4 <sub>dec</sub>	100 <sub>bin</sub>
Slave 5	4 <sub>dec</sub>	100 <sub>bin</sub>	5 <sub>dec</sub>	101 <sub>bin</sub>
Slave 6	5 <sub>dec</sub>	101 <sub>bin</sub>	6 <sub>dec</sub>	110 <sub>bin</sub>
Slave 7	6 <sub>dec</sub>	110 <sub>bin</sub>	7 <sub>dec</sub>	111 <sub>bin</sub>

Hence, you can now poll the terminals at any desired speed and in any desired rhythm. If you only send data very rarely to a slave, remember that the watchdog in the slave terminal can trigger. The watchdog time in the slave is approx. 400 ms and can be set in [register R38 \[▶ 64\]](#). The default value is 20 and must be multiplied by 20 ms.

If a slave does not answer, then either the radio connection is disturbed or bit 0 of the control byte on the slave side is not set; the master terminal sets the error bit 6. The error code is contained in the high nibble in SB2.

If you want to address a new slave, make sure that the data bytes 0...9 are updated on the new slave at the same time. When the slave replies, you should only accept the data from the new slave if the new slave address appears in the status and the error bit is not set.

## 5.3 Broadcast mode

### Application

Broadcast mode enables data to be sent from a master to any number of slaves. Data is thereby sent to the slaves. The slaves receive the data, but the master does not receive any reply from the slaves.

For broadcast mode, the master must be set to master operation with the aid of the [DIP switch \[▶ 13\]](#) and [register 39 \[▶ 64\]](#) must be set to broadcast mode before setting data communication. It is sufficient for the slave to be set to broadcast-slave mode via the DIP switch.

The sequence for setting the master module to broadcast-master mode is as follows:

1. Remove the write protection [R31 \[▶ 63\]](#)
2. Write broadcast master mode to [R39 \[▶ 64\]](#)
3. Read out [R39 \[▶ 64\]](#) to check that master mode has really been set
4. Set the start bit in [CB1 \[▶ 59\]](#)

### Example: Task – start the data exchange (same for master and slave)

Set bit 0 to *TRUE* in order to start data exchange.

#### CB1 [▶ 59]

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
Name	RegAccess	reserve	Add3	Add2	Add1	Scan	EnergyScan	<b>Start</b>
Value	0	0	0	0	0	0	0	<b>1</b>

#### SB1 [▶ 59]

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
Name	RegAccess	Error	Add3	Add2	Add1	Scan	EnergyScan	<b>Start</b>
Value	0	0	0	0	0	0	0	<b>1</b>

## 5.4 Energy scan

### Application

The energy scan enables the 16 IEEE 802.15.4 channels to be monitored in order to detect other radio systems. The energy in a frequency band is determined and displayed. Each channel is measured for approx. 5 seconds before moving to the next channel. Care must be taken that a channel does not communicate during these 5 seconds. It therefore makes sense to repeat the scanning of the channels a couple of times in order to obtain a more accurate statement as to whether or not a channel is occupied. The energy level is displayed through 0...0xF; "0" means no energy, "0xF" or 16dec stands for high energy level.

Set bit 1 in CB1 to TRUE. The KM6551-0000 confirms this in the status with bit 1, which is then also set to TRUE. The scan is finished when bit 1 of SB1 goes to FALSE. The result is then available in the input data byte 0 to 7. Each channel then corresponds to a nibble, i.e. half of a byte.

### Example

Byte number	Description	Value (hex)	Meaning
1	SB1	-	Status byte 1
2	SB2	-	Status byte 2
3	Data IN[0]	0x3F	3 - channel 1 low energy, F - channel 2 very high energy
4	Data IN[1]	0x01	0 - channel 3 no energy, 1 - channel 4 very low energy
5	Data IN[2]	0x7F	7 - channel 5 moderate energy, F - channel 5 very high energy
6...9	...	...	... (not considered in this example for reasons of simplicity)
10	Data IN[8]	0x10	1 - channel 15 low energy, 0 - channel 16 no energy
11...12	Data IN[8...9]	-	not required

The result is to be interpreted as follows. Channels 2 and 5 are to be avoided at all costs, but channels 3 and 16 look very good. No energy was measured here. Please note that these are instantaneous values. You can exclude channels with a high energy from further searches for free channels, but channels with no or only moderate energy may lead under certain circumstances to entirely different results.

### Example: activation of the energy scan

Set bit 1 to *TRUE* in order to start data exchange.

#### CB1 [► 59]

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
<b>Name</b>	RegAccess	reserve	Add3	Add2	Add1	Scan	<b>EnergyScan</b>	Start
<b>Value</b>	0	0	0	0	0	0	<b>1</b>	0

The terminal sets bit 1 of the status byte SB1 to TRUE as long as the scan is active.

#### SB1 [► 59]

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
<b>Name</b>	RegAccess	Error	Add3	Add2	Add1	Scan	<b>EnergyScan</b>	Start
<b>Value</b>	0	0	0	0	0	0	<b>1</b>	0

The terminal resets bit 1 of the status byte SB1 to FALSE once the scan is finished. You can now evaluate the input data bytes 0...7 (bytes 8-9 have no meaning and should not be evaluated).

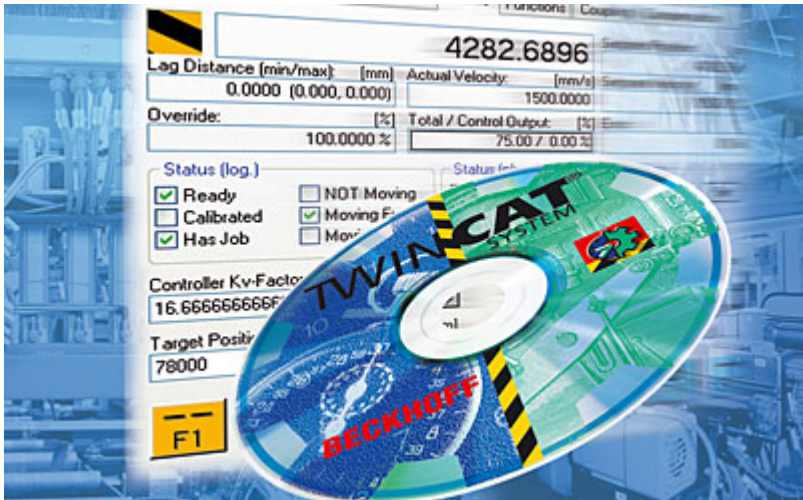
The scan takes approx. 80 seconds.



**SB1 [▶ 59]**

<b>Bit</b>	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
<b>Name</b>	RegAccess	Error	Add3	Add2	Add1	Scan	<b>EnergyScan</b>	Start
<b>Value</b>	0	0	0	0	0	0	<b>0</b>	0

## 6 TwinCAT



### PLC and Motion Control on the PC

TwinCAT - The **Windows Control and Automation Technology**

The TwinCAT automation software converts any compatible PC into a real-time controller with multi-PLC, NC axis control, programming environment and operating station. TwinCAT replaces conventional PLC and NC controllers as well as operating devices:

- open, compatible PC hardware
- Embedding of IEC 61131-3 software PLC, software NC and software CNC in Windows NT/2000/XP, NT/XP Embedded, CE
- Programming and runtime systems optionally together on one PC or separated
- Connection to all common fieldbus systems
- PC interfaces are supported
- Data communication with user interfaces and other programs by means of open Microsoft standards (OPC, OCX, DLL, etc.)

### TwinCAT architecture

TwinCAT consists of runtime systems for real-time execution of control programs and development environments for programming, diagnosis and configuration. Any Windows programs, for instance visualization programs or Office programs, can access TwinCAT data via Microsoft interfaces, or can execute commands.

### A practically oriented software solution

TwinCAT offers a precise time-base in which programs are executed with the highest deterministic features, independently of other processor tasks. The real-time load on a PC is set with TwinCAT: This achieves a defined operating behavior. TwinCAT displays the system load for running programs. A loading threshold can be set, in order to assure a defined computing capacity for the operating programs and for Windows NT/2000/XP. If this threshold is exceeded, a system message is generated.

### TwinCAT supports system diagnosis

The general use of hardware and software from the open PC world requires some checking: Unsuitable components can upset the PC system. Beckhoff integrates a handy display of the real-time jitter in order to provide administrators with a simple means of evaluating hardware and software. A system message during operation can draw attention to error states.

### Start/stop behavior

Depending on the setting, TwinCAT is started and stopped manually or automatically. Since TwinCAT is integrated into Windows NT/2000/XP as a service, an operator is not needed to start the system: switching on is enough.

### Restarting and data backup

When a program is started or restarted, TwinCAT loads programs and remanent data. To backup data, and to shut down Windows NT/2000/XP correctly, a UPS (uninterruptible power supply) is of great value.

### TwinCAT and "Blue Screen"

The TwinCAT system can be configured such that real-time capability is maintained in the event of a BSOD (Blue-Screen-of-Death) operating system crash. Real-time tasks such as PLC and NC can thus continue to run and place the controlled process in a safe state. Ultimately, it is the decision of the programmer whether or not to utilize this feature, bearing in mind that data or programs may already have been destroyed by the BSOD.

### World-wide connection through message routing - "remote" connection is inherent to the system

According to the requirement for operating resources, the TwinCAT software devices can be distributed: TwinCAT PLC programs can be executed on PCs and on Beckhoff Bus Terminal controllers. A "message router" manages and distributes all the messages, both in the system and via TCP/IP connections. PC systems can be connected to one another by TCP/IP; Bus Terminal controllers are connected via serial interfaces and fieldbus systems (EtherCAT, Lightbus, PROFIBUS DP, PROFINET, Interbus, CANopen, DeviceNet, RS232, RS485, Ethernet TCP/IP, Ethernet/IP).

### World-wide access

Since standard TCP/IP services from Windows NT/2000/XP are used, this data exchange can take place worldwide. The system offers scalable communication capacity and timeout periods for the monitoring of communications. OPC provides a standardized means for accessing many different SCADA packets. The SOAP (Simple Object Access Protocol) enables a connection between two computers to be established by means of an internet connection via standard HTTP. A TwinCAT component is available for this purpose.

### Beckhoff Information System


Further information on the TwinCAT automation software can be found in the Beckhoff Information System.

The setup for installing the Beckhoff Information System is available to you on the Beckhoff *Products & Solutions* DVD and on our website for [download](#).

In addition, the online version of the Beckhoff Information System can be found at <https://infosys.beckhoff.com>.

## 6.1 TwinCAT libraries

A TwinCAT library is available for all Beckhoff controller families (BC, BX, CX and IPC) for the operation of the KM6551-0000 under TwinCAT. This library takes care of communication with the terminal. It sets parameters in or reads parameters from the terminal. The use of the library simplifies communication with the terminal for the user.

ZIP file TC\_KM6551.zip:  <https://infosys.beckhoff.com/content/1033/km6551/Resources/zip/9305592971.zip>

Copy the libraries into the TwinCAT\PLC\Lib directory.

Other required libraries:

For Bus Terminal Controllers from the BCxx00 series

- Standard.lb6
- PlcHelper.lb6

For Bus Terminal Controllers from the BCxx20 and BCxx50 series

- Standard.lbx
- TcBaseBCxx50.lbx
- TcSystemBCxx50.lbx

For Bus Terminal Controllers from the BXxxxx series

- Standard.lbx
- TcBaseBX.lbx
- TcSystemBX.lbx

For 386-based systems such as Industrial PCs, Embedded PCs (CX)

- Standard.lib
- TcBase.lib
- TcSystem.lib

The user can write his own function blocks for the operation of the KM6551-0000 under other controllers.

## 6.2 TwinCAT examples

For the examples you need one CX9000 with K-bus, one BC9050, two KM6551-0000, two KL9010 end terminals and one KL2xx4 (optional).

**Example for master/slave communication** (<https://infosys.beckhoff.com/content/1033/km6551/Resources/zip/9305595147.zip>)

A KM6551-0000 is plugged into the CX9000 as a master module and optionally the KL2xx4 Bus Terminal. A KM6551-0000 is plugged into the BC9050 as a slave module.

**Example for broadcast communication** (<https://infosys.beckhoff.com/content/1033/km6551/Resources/zip/9305597323.zip>)

A KM6551-0000 is plugged into the CX9000 as a master module and optionally the KL2xx4 Bus Terminal. A KM6551-0000 is plugged into the BC9050 as a broadcast slave module.

**Example for energy scan** (<https://infosys.beckhoff.com/content/1033/km6551/Resources/prx/9305599499.prx>)

One BX9000, one KL6551-0000 and one KL9010 are required.

## 6.3 Function blocks

### 6.3.1 Function block FB\_KM6551\_MAIN

This function block takes care of communication to the KM6651-0000 and takes care of and makes settings in the module by means of register communication. Only one FB\_KM6551\_MAIN function block is permitted per KM6551-0000.

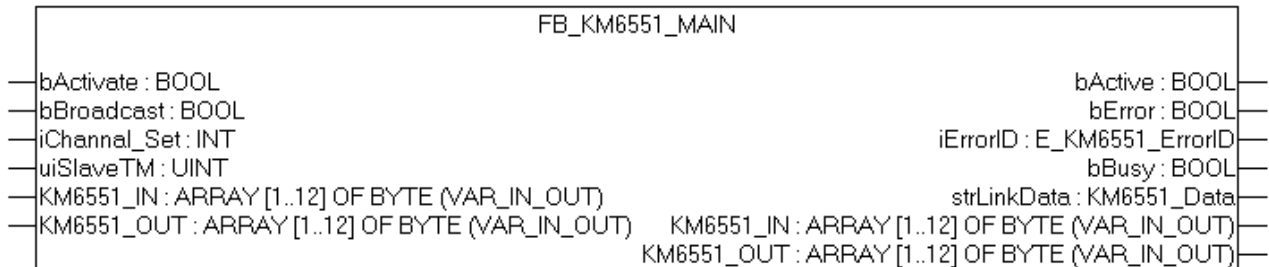


Fig. 29: Function block FB\_KM6551\_MAIN

#### VAR\_INPUT

```
bActivate
:BOOL;
bBroadcast
:BOOL;
iChannel_Set
:INT;
uiSlaveTM
:UINT;
KM6551_IN
:ARRAY[1..12] of BYTE;
KM6551_OUT
:ARRAY[1..12] of BYTE;
```

#### Key

**bActivate:** Positive edge activates the function block and writes parameters to the KM6551-0000.

**bBroadcast:** Can only be activated in master mode. TRUE – sets the KM6651-0000 to broadcast master mode (see register 39 [▶ 64]).

**iChannel\_Set:** The IEEE 802.15.4 channel is set here. Permitted values 0..15 (see register 32/33 [▶ 63]).

**uiSlaveTM:** Only usable in slave mode; setting of the watchdog for the slave mode (see register 38 [▶ 64]).

**KM6551\_IN:** Is connected to the INPUT data of the KM6551-0000.

**KM6551\_OUT:** Is connected to the OUTPUT data of the KM6551-0000.

#### VAR\_OUTPUT

```
bActive
:BOOL;
bError
:BOOL;
iErrorID
:E_KM6551_ERRORID;
bBusy
:BOOL;
strLinkData
:KM6551_Data;
```

#### Key

**bActive:** The function block has successfully transmitted the parameters to KM6551-0000 and can now commence data communication with the other KM6551-0000 function blocks.

**bError:** The function block has an error.

**iErrorID:** Contains the error code.

**bBusy:** The function block is still working as long as *bBusy* is set, i.e. is TRUE; wait until *bBusy* changes to

FALSE.

**strLinkData:** Data required by the higher-level function blocks. Connect this data to the additional function blocks that they call.

## 6.3.2 Energy scan

### Application

The energy scan enables the 16 IEEE 802.15.4 channels to be monitored in order to detect other radio systems. The energy in a frequency band is determined and displayed. Each channel is measured for approx. 5 seconds before moving to the next channel. Care must be taken that a channel does not communicate during these 5 seconds. It therefore makes sense to repeat the scanning of the channels a couple of times in order to obtain a more accurate statement as to whether or not a channel is occupied. The energy level is displayed through 0...0xF; "0" means no energy, "0xF" or 16dec stands for high energy level.

Set bit 1 in CB1 to TRUE. The KM6551-0000 confirms this in the status with bit 1, which is then also set to TRUE. The scan is finished when bit 1 of SB1 goes to FALSE. The result is then available in the input data byte 0 to 7. Each channel then corresponds to a nibble, i.e. half of a byte.

### Example

Byte number	Description	Value (hex)	Meaning
1	SB1	-	Status byte 1
2	SB2	-	Status byte 2
3	Data IN[0]	0x3F	3 - channel 1 low energy, F - channel 2 very high energy
4	Data IN[1]	0x01	0 - channel 3 no energy, 1 - channel 4 very low energy
5	Data IN[2]	0x7F	7 - channel 5 moderate energy, F - channel 5 very high energy
6...9	...	...	... (not considered in this example for reasons of simplicity)
10	Data IN[8]	0x10	1 - channel 15 low energy, 0 - channel 16 no energy
11...12	Data IN[8...9]	-	not required

The result is to be interpreted as follows. Channels 2 and 5 are to be avoided at all costs, but channels 3 and 16 look very good. No energy was measured here. Please note that these are instantaneous values. You can exclude channels with a high energy from further searches for free channels, but channels with no or only moderate energy may lead under certain circumstances to entirely different results.

### Example: activation of the energy scan

Set bit 1 to *TRUE* in order to start data exchange.

#### CB1 [► 59]

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
<b>Name</b>	RegAccess	reserve	Add3	Add2	Add1	Scan	<b>EnergyScan</b>	Start
<b>Value</b>	0	0	0	0	0	0	<b>1</b>	0

The terminal sets bit 1 of the status byte SB1 to TRUE as long as the scan is active.

#### SB1 [► 59]

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
<b>Name</b>	RegAccess	Error	Add3	Add2	Add1	Scan	<b>EnergyScan</b>	Start
<b>Value</b>	0	0	0	0	0	0	<b>1</b>	0

The terminal resets bit 1 of the status byte SB1 to FALSE once the scan is finished. You can now evaluate the input data bytes 0...7 (bytes 8-9 have no meaning and should not be evaluated).

The scan takes approx. 80 seconds.

**SB1 [▶ 59]**

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
Name	RegAccess	Error	Add3	Add2	Add1	Scan	EnergyScan	Start
Value	0	0	0	0	0	0	0	0

### 6.3.3 Master/Slave mode

#### 6.3.3.1 Function block FB\_KM6551\_MASTER\_10BYTE

This function block takes care of communication to the individual slaves. Only one function block can be called per slave. A maximum of 7 slaves are allowed, for which reason a maximum of 7 function blocks of this type may be called per master terminal. A positive edge on *bStart* activates communication to the slave that is stored in the variable *iSlaveAddr*. **bBusy** goes to TRUE as long as the function block is active. If **bBusy** changes to FALSE, the function block has finished. If *bError* is FALSE, then communication was successful and the input data is valid. If the **bError** bit is TRUE, an error has occurred. A precise error cause can be read out in **iErrorID**. **ptData\_IN** is a pointer address for the input data (ADR command to determine the pointer address) and **iLenData\_IN** is the length of the data. The length can be determined with SIZEOF and must not be larger than 10 bytes. The same applies to the output data. The **strLinkData** variable is linked to the FB\_KM6551\_MAIN function block (the variable has exactly the same name). The FB\_KM6551\_MAIN function block should be finished before the FB\_KM6551\_MASTER\_10BYTE function block is called, (see *bActive* in this function block).

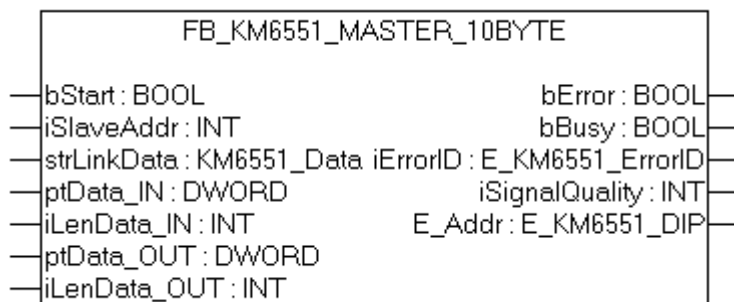


Fig. 30: Function block FB\_KM6551\_Master\_10Byte

**VAR\_INPUT**

```

bStart
:BOOL;
iSlaveAddr
:INT;
strLinkData
:KM6551_Data;
ptData_IN
:DWORD;
iLenData_IN      :INT;
ptData_OUT
:DWORD;
iLenData_OUT     :INT;
    
```

**Key**

- bStart:** A positive edge activates the function block.
- iSlaveAddr:** The address of the slave is entered here (see DIP switch on your slave module)
- strLinkData:** Is linked with *strLinkData* from FB\_KM6551\_MAIN.
- ptData\_IN:** Pointer to the variable into which the device data should be copied (pointer address is

determined with ADR(Variable\_name)).

*iLenData\_IN*: Length of the variable (the length can be determined with SIZEOF (variable\_name))

*ptData\_OUT*: Pointer to the variable into which the master terminal should transmit to the slave (pointer address is determined with ADR(Variable\_name)).

*iLenData\_OUT*: Length of the variable (the length can be determined with SIZEOF (variable\_name))

## VAR\_OUTPUT

```
bError
:BOOL;
bBusy
:BOOL;
iErrorID
:E_KM6551_ERRORID;
iSignalQuality:INT;
E_Addr
:E_KM6551_DIP;
```

## Key

**b\_Error**: The function block has an error.

**bBusy**: The function block is still working as long as *bBusy* is set, i.e. is TRUE; wait until *bBusy* changes to FALSE.

**iErrorID**: Contains the error code.

**iSignalQuality**: LQI value, quality of the signal received; 100 very good transmission, 0 very poor transmission - the LQI value should be as high as possible and should have a minimum value of 10 - 20. You can improve the LQI value by the use of better antennas or shorter cables or better alignment of the antennas.

*E\_Addr*: Reads out the DIP switch setting of the KM6551-0000 module and displays it.

### 6.3.3.2 Function block FB\_KM6551\_SLAVE\_10BYTE

This function block takes care of communication to the slave module. Only one function block can be called per slave. A positive edge on *bStart* activates communication to the slave. If *bError* is FALSE, then communication was successful and the input data is valid. If the **bError** bit is TRUE, an error has occurred. A precise error cause can be read out in **iErrorID**. *ptData\_IN* is a pointer address for the input data (ADR command to determine the pointer address) and **iLenData\_IN** is the length of the data. The length can be determined with SIZEOF and must not be larger than 10 bytes. The same applies to the output data. The **strLinkData** variable is linked to the FB\_KM6551\_MAIN function block (the variable has exactly the same name). The FB\_KM6551\_MAIN function block should be finished before the FB\_KM6551\_SLAVE\_10BYTE function block is called, (see *bActive* in this function block).

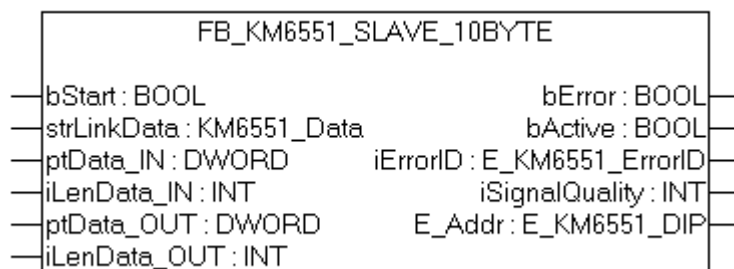


Fig. 31: Function block FB\_KM6551\_Slave\_10Byte

## VAR\_INPUT

```
bStart
:BOOL;
strLinkData
:KM6551_Data;
ptData_IN
:DWORD;
iLenData_IN :INT;
```



```
ptData_OT
:DWORD;
iLenData_OUT :INT;
```

### Key

**bStart:** A positive edge activates the function block.

**strLinkData:** Is linked with strLinkData from FB\_KM6551\_MAIN.

**ptData\_IN:** Pointer to the variable into which the device data should be copied (pointer address is determined with ADR(Variable\_name)).

**iLenData\_IN:** Length of the variable (the length can be determined with SIZEOF (variable\_name))

**ptData\_OUT:** Pointer to the variable into which the master terminal should transmit to the slave (pointer address is determined with ADR(Variable\_name)).

**iLenData\_OUT:** Length of the variable (the length can be determined with SIZEOF (variable\_name))

### VAR\_OUTPUT

```
bError
:BOOL;
bBusy
:BOOL;
iErrorID
:E_KM6551_ERRORID;
iSignalQuality :INT;
E_Addr
:E_KM6551_DIP;
```

### Key

**b Error:** The function block has an error.

**bBusy:** The function block is still working as long as *bBusy* is set, i.e. is TRUE; wait until *bBusy* changes to FALSE.

**iErrorID:** Contains the error code.

**iSignalQuality:** LQI value, quality of the signal received; 100 very good transmission, 0 very poor transmission - the LQI value should be as high as possible and should have a minimum value of 10 - 20. You can improve the LQI value by the use of better antennas or shorter cables or better alignment of the antennas.

**E\_Addr:** Reads out the DIP switch setting of the KM6651-0000 module and displays it.

## 6.3.4 Broadcast mode

### 6.3.4.1 Function block FB\_KM6551\_MASTERBROADCAST\_10BYTE

This function block takes care of communication to the individual slaves in broadcast mode. The number of data telegrams can be reduced with the time *tPolling*. Data will then only be sent within the time **tPolling**. If the *tPolling* time is high for the broadcast slaves, make sure that you also enter a higher watchdog time, as otherwise the slave will display a watchdog error. A positive edge on *bStart* activates communication. If the **bError** bit is TRUE, an error has occurred. A precise error cause can be read out in **iErrorID**. **ptData\_OUT** is a pointer address for the input data (ADR command to determine the pointer address) and **iLenData\_OUT** is the length of the data. The length can be determined with SIZEOF and must not be larger than 10 bytes. The **strLinkData** variable is linked to the FB\_KM6551\_MAIN function block (the variable has exactly the same name). The FB\_KM6551\_MAIN function block should be finished before the FB\_KM6551\_MASTER\_10BYTE function block is called, (see *bActive* in this function block).

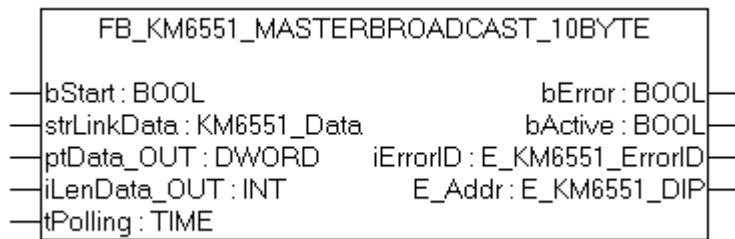


Fig. 32: Function block FB\_KM6551\_Masterbroadcast\_10Byte

**VAR\_INPUT**

```

bStart
:BOOL;
iSlaveAddr
:INT;
strLinkData
:KM6551_Data;
ptData_OUT
:DWORD;
iLenData_OUT :INT;
tPolling
:TIME;

```

**Key**

**bStart:** A positive edge activates the function block.

**iSlaveAddr:** The address of the slave is entered here (see DIP switch on your slave module)

**strLinkData:** Is linked with strLinkData from FB\_KM6551\_MAIN.

**ptData\_OUT:** Pointer to the variable into which the master terminal should transmit to the slave (pointer address is determined with ADR(Variable\_name)).

**iLenData\_OUT:** Length of the variable (the length can be determined with SIZEOF (variable\_name))

**tPolling:** Time cycle in which the data should be sent to the slaves.

**VAR\_OUTPUT**

```

bError
:BOOL;
bActive
:BOOL;
iErrorID
:E_KM6551_ERRORID;
E_Addr
:E_KM6551_DIP;

```

**Key**

**b Error:** The function block has an error.

**bActive:** Indicates whether or not the function block is working.

**iErrorID:** Contains the error code.

**E\_Addr:** Reads out the DIP switch setting of the KM6551-0000 and displays it.

**6.3.4.2 Function block FB\_KM6551\_SLAVEBROADCAST\_10BYTE**

This function block takes care of communication to the broadcast slave module. Only one function block can be called per slave. A positive edge on *bStart* activates communication to the slave. If *bError* is FALSE, then communication was successful and the input data is valid. If the **bError** bit is TRUE, an error has occurred. A precise error cause can be read out in **iErrorID**. **ptData\_IN** is a pointer address for the input data (ADR command to determine the pointer address) and **iLenData\_IN** is the length of the data. The length can be determined with SIZEOF and must not be larger than 10 bytes. The same applies to the output data. The **strLinkData** variable is linked to the FB\_KM6551\_MAIN function block (the variable has exactly the same name). The FB\_KM6551\_MAIN function block should be finished before the FB\_KM6551\_MASTER\_10BYTE function block is called, (see *bActive* in this function block).

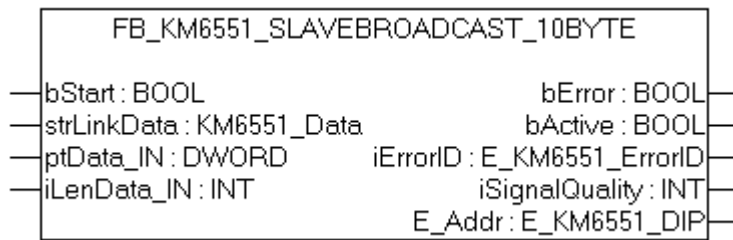


Fig. 33: Function block FB\_KM6551\_Slavebroadcast\_10Byte

**VAR\_INPUT**

```

bStart
:BOOL;
strLinkData
:KM6551_Data;
ptData_IN
:DWORD;
iLenData_IN      :INT;

```

**Key**

**bStart:** A positive edge activates the function block.

**strLinkData:** Is linked with strLinkData from FB\_KM6551\_MAIN.

**ptData\_IN:** Pointer to the variable into which the device data should be copied (pointer address is determined with ADR(Variable\_name)).

**iLenData\_IN:** Length of the variable (the length can be determined with SIZEOF (variable\_name))

**VAR\_OUTPUT**

```

bError
:BOOL;
bActive
:BOOL;
iErrorID
:E_KM6551_ERRORID;
iSignalQuality      :INT;
E_Addr
:E_KM6551_DIP;

```

**Key**

**b Error:** The function block has an error.

**bActive:** Indicates whether or not the function block is working.

**iErrorID:** Contains the error code.

**iSignalQuality:** LQI value, quality of the signal received; 100 very good transmission, 0 very poor transmission - the LQI value should be as high as possible and should have a minimum value of 10 - 20. You can improve the LQI value by the use of better antennas or shorter cables or better alignment of the antennas.

**E\_Addr:** Reads out the DIP switch setting of the KM6651-0000 module and displays it.

## 7 KS2000 Configuration Software

### 7.1 KS2000 - Introduction

The KS2000 configuration software permits configuration, commissioning and parameterization of bus couplers, of the affiliated bus terminals and of Fieldbus Box Modules. The connection between bus coupler / Fieldbus Box Module and the PC is established by means of the serial configuration cable or the fieldbus.



Fig. 34: KS2000 configuration software

#### Configuration

You can configure the Fieldbus stations with the Configuration Software KS2000 offline. That means, setting up a terminal station with all settings on the couplers and terminals resp. the Fieldbus Box Modules can be prepared before the commissioning phase. Later on, this configuration can be transferred to the terminal station in the commissioning phase by means of a download. For documentation purposes, you are provided with the breakdown of the terminal station, a parts list of modules used and a list of the parameters you have modified. After an upload, existing fieldbus stations are at your disposal for further editing.

#### Parameterization

KS2000 offers simple access to the parameters of a fieldbus station: specific high-level dialogs are available for all bus couplers, all intelligent bus terminals and Fieldbus Box modules with the aid of which settings can be modified easily. Alternatively, you have full access to all internal registers of the bus couplers and intelligent terminals. Refer to the register description for the meanings of the registers.

## Commissioning

The KS2000 software facilitates commissioning of machine components or their fieldbus stations: Configured settings can be transferred to the fieldbus modules by means of a download. After a *login* to the terminal station, it is possible to define settings in couplers, terminals and Fieldbus Box modules directly *online*. The same high-level dialogs and register access are available for this purpose as in the configuration phase.

The KS2000 offers access to the process images of the bus couplers and Fieldbus Box modules.

- Thus, the coupler's input and output images can be observed by monitoring.
- Process values can be specified in the output image for commissioning of the output modules.

All possibilities in the *online mode* can be used in parallel with the actual fieldbus mode of the terminal station. The fieldbus protocol always has the higher priority in this case.

## 7.2 Parameterization with KS2000

Connect the configuration interface of your Fieldbus Coupler with the serial interface of your PC via the configuration cable and start the *KS2000* Configuration Software.



Click on the *Login* button. The configuration software will now load the information for the connected fieldbus station.

In the example shown, this is

- a BK9000 Bus Coupler for Ethernet
- a KL1002 input terminal
- a KM6551-0000 data exchange terminal
- a KL9010 Bus End Terminal

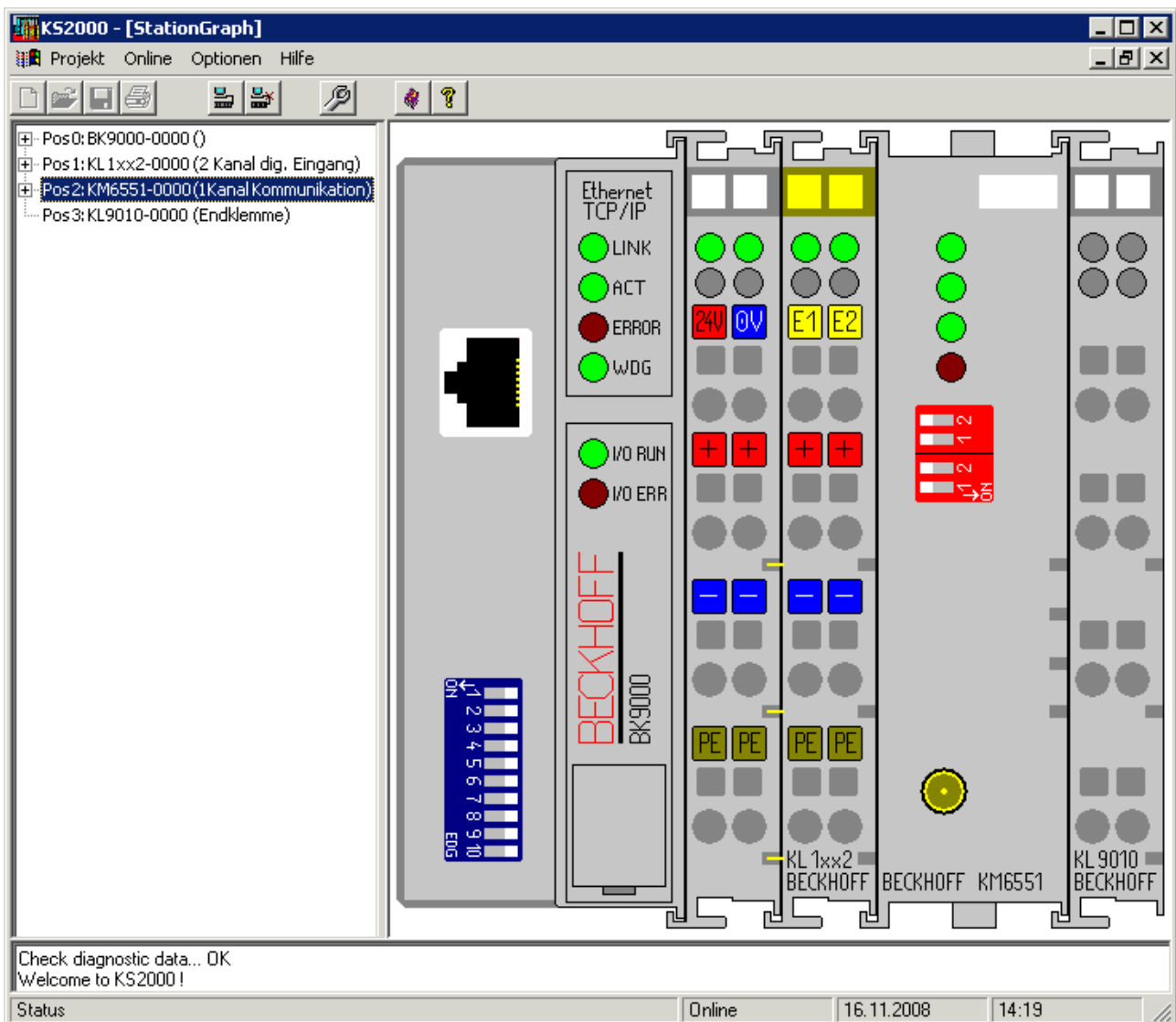


Fig. 35: Display of the fieldbus station in KS2000

The left-hand KS2000 window displays the terminals of the fieldbus station in a tree structure. The right-hand KS2000 window contains a graphic display of the fieldbus station terminals.

In the tree structure of the left-hand window, click on the plus-sign next to the module whose parameters you wish to change (pos. 2 in the example).

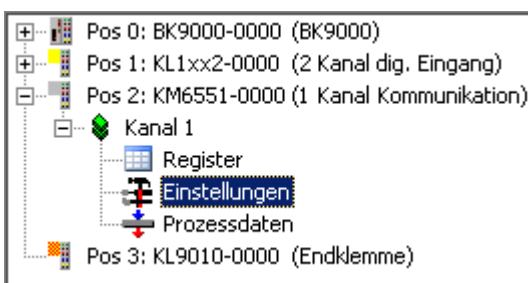


Fig. 36: KS2000 branch for channel 1 of the KM6551

The tree branches *Register*, *Settings* and *ProcData* are displayed for the KM6551-0000:

- [Register](#) [▶ 56] allows direct access to the registers of the KM6551-0000.
- Under [settings](#) [▶ 55] you will find dialog masks for parameterizing the KM6551-0000.
- [Process data](#) [▶ 57] shows the process data of the KM6551-0000.

## 7.3 Settings

Dialog screen for parameterizing the KM6551.

Pos.: 2                      Klemmenkanal: 1                      Firmware: 1D  
Typ: KM6551-0000

**Einstellungen**  
Modus (Dip Switch):                      Master  
Funkkanal für KM6551:                      5

**Scan Einstellungen**  
Gefundene Slaves: 0                      Scannen

**Slaves**  
7 6 5 4 3 2 1

Übernehmen  
Abbrechen

Fig. 37: Dialog screen for parameterizing the KM6551

### Settings

#### Radio channel for KM6551 (R33)

[R33](#) [▶ 64](#)

Here you can set the radio channel. (Default: 5, permissible values: 0 to 15).

#### Scan settings

##### Found slaves

Displays the number of slaves found ([R40](#) [▶ 64](#)).

##### Slaves (R40)

[R40](#) [▶ 64](#)

Displays the slave numbers of the found slaves.

## 7.4 Register

Under *Register* you can directly access the registers of the KM6551-0000 terminal module. The meaning of the register is explained in the [Register overview](#) [▶ 62].

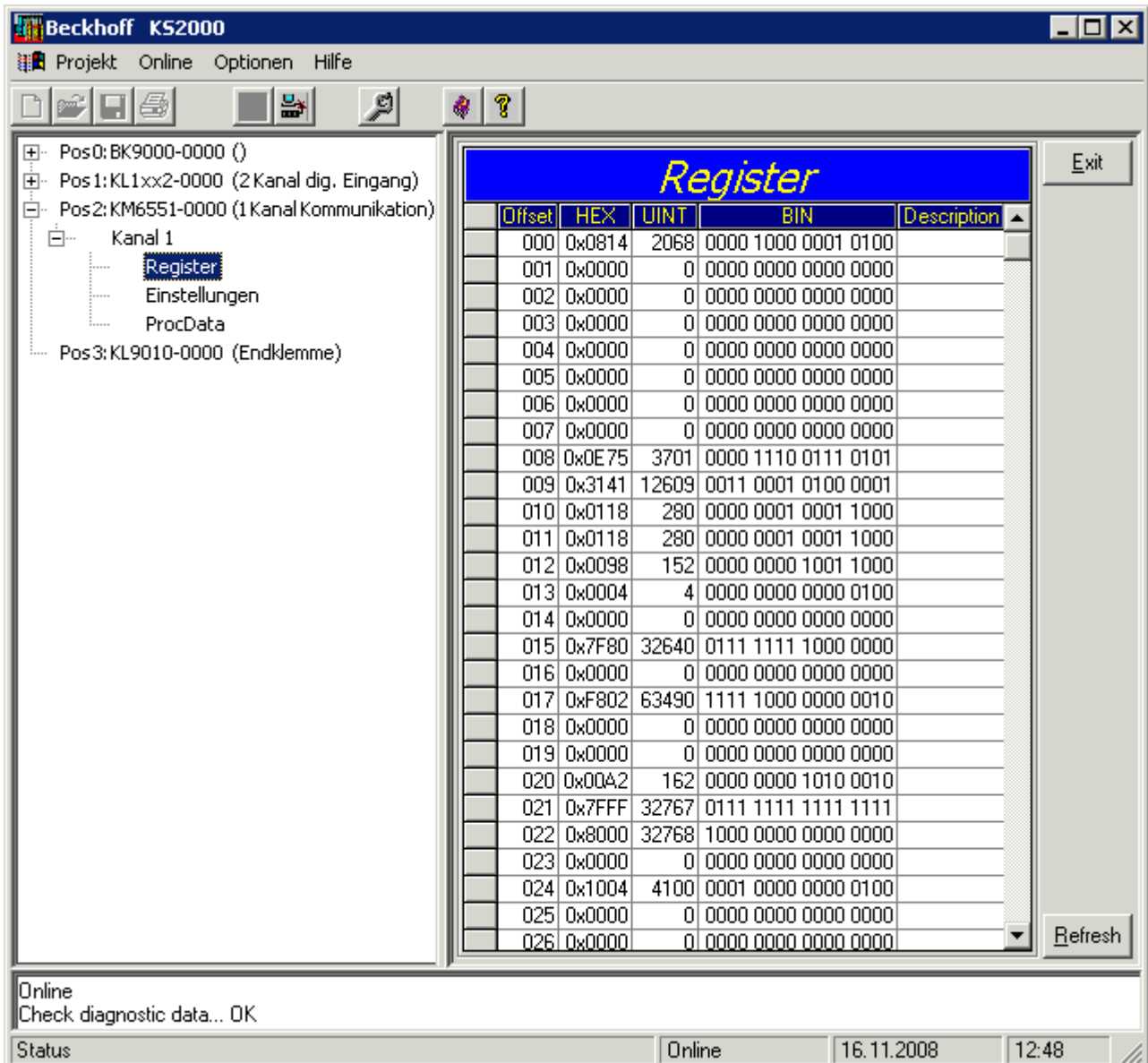


Fig. 38: Register view in KS2000



## 7.5 Process data

The Status byte (Status), the Control byte (Ctrl) and the process data (Data) are displayed in a tree structure under *ProcData*.

Prozessdaten (Hex.)

Pos	Typ	E-Adresse	Wert	Bitlänge	A-Adresse	Wert	Bitlänge
2	KL6551-0000						
	Kanal 1						
	Parameter Status	0.0	0x0000	16			
	Input Data 1	2.0	0x00	8			
	Input Data 2	3.0	0x00	8			
	Input Data 3	4.0	0x00	8			
	Input Data 4	5.0	0x00	8			
	Input Data 5	6.0	0x00	8			
	Input Data 6	7.0	0x00	8			
	Input Data 7	8.0	0x00	8			
	Input Data 8	9.0	0x00	8			
	Input Data 9	10.0	0x00	8			
	Input Data 10	11.0	0x00	8			
	Parameter Control				0.0	0x0000	16
	Output Data 1				2.0	0x00	8
	Output Data 2				3.0	0x00	8
	Output Data 3				4.0	0x00	8
	Output Data 4				5.0	0x00	8
	Output Data 5				6.0	0x00	8
	Output Data 6				7.0	0x00	8
	Output Data 7				8.0	0x00	8
	Output Data 8				9.0	0x00	8
	Output Data 9				10.0	0x00	8
	Output Data 10				11.0	0x00	8

Fig. 39: Process Data field

The spectacles mark the data that are currently graphically displayed in the *History* field.

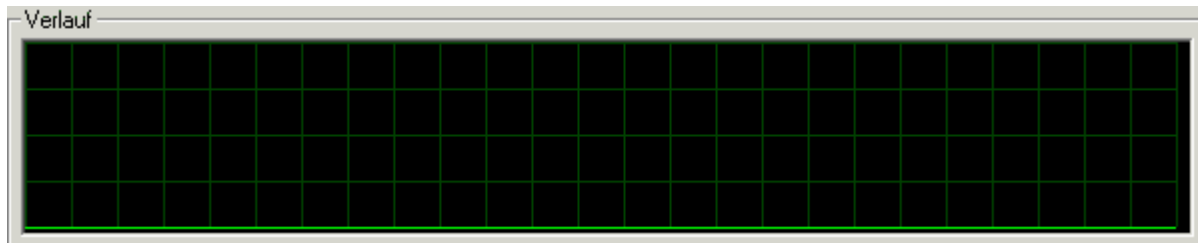


Fig. 40: History field

The current input value is displayed numerically in the *Value* field.

Wert

Dezimal

Hexadezimal

Binär

Fig. 41: Value field

Output values can be modified through direct input or by means of the fader control.

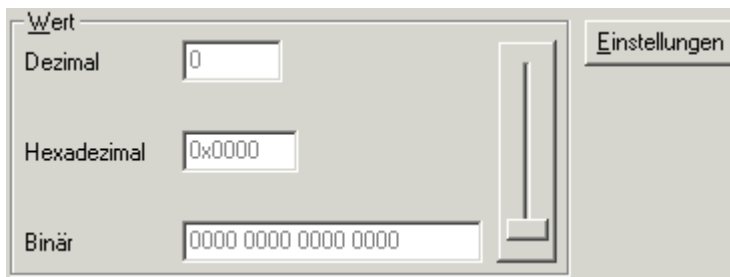


Fig. 42: Value field

**⚠ CAUTION****Danger for persons, the environment or devices!**

Note that changing output values (forcing) can have a direct effect on your automation application. Only modify these output values if you are certain that the state of your equipment permits it, and that there will be no risk to people or to the machine!

After pressing the *Settings* button you can set the format of the numerical display to hexadecimal, decimal or binary.

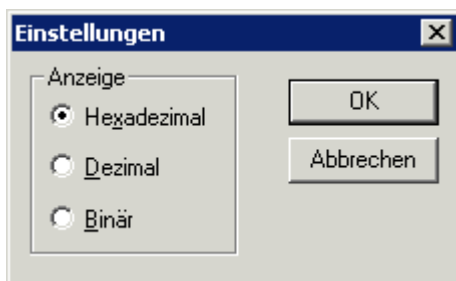


Fig. 43: Setting the display

# 8 Access from the user program

## 8.1 Process image

The KM6551-0000 terminal module represents itself in the process image with a maximum of 12 bytes of input data and 12 bytes of output data. These are organized as follows:

Format	Input data	Output data
Byte	SB1 [ <a href="#">▶ 60</a> ]	CB1 [ <a href="#">▶ 59</a> ]
Byte	SB2	CB2
Array of bytes (0...9)	DataIN	DataOUT

### Key

SBn: status byte n  
 CBn: control byte n

DataIN: Array of 10 input bytes (0...9)  
 DataOUT: Array of 10 input bytes (0...9)

The meaning of the control and status bytes is explained in [Control and status bytes \[\[▶ 59\]\(#\)\]](#).

In process data operation, the input data is transmitted in the DataIN array and the output data in the DataOUT array.



### Use of the data arrays

- use them, for example, to transmit the process data from analog input or output channels.
- also use the 10 bytes in order to transmit larger amounts of data in several cycles using a self-defined protocol (e.g. 2 bytes for the header, 8 bytes for the user data).

## 8.2 Control and Status Bytes

### 8.2.1 Process data mode

#### Control bytes (for process data mode)

The control bytes are located in the output image and are transmitted from the controller to the terminal module.

#### CB1: Low byte

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
Name	RegAccess	-	Address			Scan	EnergyScan	Start

#### CB2: High byte

Bit	CB2.7	CB2.6	CB2.5	CB2.4	CB2.3	CB2.2	CB2.1	CB2.0
Name	-	-	-	-	-	-	-	-

**Key**

Bit	Name	Description
CB2.0 to CB2.7	reserve	0 <sub>bin</sub> reserved
CB1.7	RegAccess	0 <sub>bin</sub> Register communication off (process data mode)
CB1.6	reserve	0 <sub>bin</sub> reserved
CB1.3 to CB1.5	Address	0..7 <sub>dec</sub> In master mode, with whichever device the connection is running 0 - Slave 1 1 - Slave 2 ... 6 - Slave 7
CB1.2	Scan	1 <sub>bin</sub> Activates scanning of the connected slaves (only possible in master mode). The scanned slaves can be found in register <a href="#">R38</a> [ <a href="#">▶ 64</a> ].
CB1.1	EnergyScan	1 <sub>bin</sub> Activates the energy measurement for the 16 channels of the 2.4 GHz band. EnergyScan may not be activated in data exchange or scan mode.
CB1.0	Start	1 <sub>bin</sub> Activates communication

**Status byte (for process data mode)**

The status bytes are located in the input image and are transmitted from terminal module to the controller.

**SB1: Low byte**

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
Name	RegAccess	Error	Address			Scan	EnergyScan	Start

**SB2: High byte**

Bit	SB2.7	SB2.6	SB2.5	SB2.4	SB2.3	SB2.2	SB2.1	SB2.0
Name	ErrorID				LQI			

**Key**

Bit	Name	Description
SB2.7 to SB2.4	ErrorID	0..F <sub>hex</sub> Error message 1 - invalid or forbidden DIP switch position 2 - watchdog error 3... F not used
SB2.3 to SB2.0	LQI	0..F <sub>hex</sub> Quality of the radio signal
SB1.7	RegAccess	0 <sub>bin</sub> Acknowledgment for process data mode
SB1.6	Error	1 <sub>bin</sub> an internal error has occurred (current process data is no longer valid)
SB1.5 to SB1.3	Address	1..7 <sub>dec</sub> Slave address 1: Slave 1 2: Slave 2 ... 7: Slave 7
SB1.2	Scan	1 <sub>bin</sub>
SB1.1	EnergyScan	1 <sub>bin</sub> TRUE if energy scanning is active; will be reset to FALSE when the terminal has finished.
SB1.0	Start	1 <sub>bin</sub>

## 8.2.2 Register communication

Control byte 2 and status byte 2 have no function in the case of register communication.

### Control byte 1 (in register communication)

Control byte 1 (CB1) is located in the output image, and is transmitted from the controller to the terminal module.

Bit	CB1.7	CB1.6	CB1.5	CB1.4	CB1.3	CB1.2	CB1.1	CB1.0
Name	RegAccess	R/W	Reg. no.					

#### Key

Bit	Name	Description	
CB1.7	RegAccess	1 <sub>bin</sub>	Register communication switched on
CB1.6	R/W	0 <sub>bin</sub>	Read access
		1 <sub>bin</sub>	Write access
CB1.5 to CB1.0	Reg. no.	Register number: Enter here the number of the register that you wish - to read with input data word DataIn, or - to write with output data word DataOut.	

### Status byte 1 (in register communication)

Status byte 1 (SB1) is located in the input image and is transmitted from terminal module to the controller.

Bit	SB1.7	SB1.6	SB1.5	SB1.4	SB1.3	SB1.2	SB1.1	SB1.0
Name	RegAccess	R/W	Reg. no.					

#### Key

Bit	Name	Description	
SB1.7	RegAccess	1 <sub>bin</sub>	Acknowledgment for register access
SB1.6	R	0 <sub>bin</sub>	Read access
SB1.5 to SB1.0	Reg. no.	Number of the register that was read or written.	

## 8.3 Register overview

These registers are used to parameterize the terminal module. They can be read or written by means of the register communication [► 65].

Register no.	Comment	Default value		R/W	Memory
R0	reserved	0x0000	0 <sub>dec</sub>	-	-
...	...	...	...	...	...
R7	reserved	0x0000	0 <sub>dec</sub>	-	-
R8 [► 63]	Terminal type	0x1997	6551 <sub>dec</sub>	R	ROM
R9 [► 63]	Firmware version	e.g. 0x3144	e.g. 1D <sub>ASCII</sub>	R	ROM
R10	Multiplex shift register	0x0118	280 <sub>dec</sub>	R	ROM
R11	Signal channels	0x0218	280 <sub>dec</sub>	R	ROM
R12	minimum data length of a channel	0x0098	152 <sub>dec</sub>	R	ROM
R13	Data structure	0x0000	0 <sub>dec</sub>	R	ROM
R14	reserved	0x0000	0 <sub>dec</sub>	-	-
R15	Alignment register	typically 0x7F80	typically 32640 <sub>dec</sub>	R/W	RAM
R16 [► 63]	DIP switch setting	e.g. 0x0000	e.g. 0 <sub>dec</sub>	R	RAM
R17	internal use	typically 0x0000	typically 0 <sub>dec</sub>	R	EEPROM
R18	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM
R19	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM
R20	reserved for internal use	0x0001	1 <sub>dec</sub>	R	EEPROM
R21	reserved for internal use	0x0500	1280 <sub>dec</sub>	R	EEPROM
R22	reserved for internal use	0x0000	0 <sub>dec</sub>	R	EEPROM
R23	reserved for internal use	0x00FF	255 <sub>dec</sub>	R	EEPROM
R24	reserved	0x0000	0 <sub>dec</sub>	-	-
...	...	...	...	...	...
R30	reserved	0x0000	0 <sub>dec</sub>	-	-
R31 [► 63]	Code word register	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R32 [► 63]	IEEE 802.15.4 channel	0x0005	5 <sub>dec</sub>	R/W	EEPROM
R33	Control register for R32	0x0000	0 <sub>dec</sub>	R	RAM
R34	reserved	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R35	reserved	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R36	reserved	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R37	reserved	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R38 [► 64]	Watchdog for slave mode	0x0014	20 <sub>dec</sub>	R/W	EEPROM
R39 [► 64]	Broadcast mode	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R40	Scan register	-	-	R	RAM
R41 [► 64]	Network ID	0x0000	0 <sub>dec</sub>	R/W	RAM
R42	Wrong channel (internal)	-	-	R	EEPROM
R43	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM
R44	Reset counter (internal)	-	-	R	EEPROM
R45	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM
R46 [► 64]	Attenuation of the transmission power	0x0000	0 <sub>dec</sub>	R/W	EEPROM
R47	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM
...	...	...	...	...	...
R63	reserved	0x0000	0 <sub>dec</sub>	R	EEPROM

## 8.4 Register description

These registers are used to parameterize the terminal module. They can be read or written by means of the [register communication](#) [► 65].

### R8: Module ID

Register R8 contains the name for the terminal module.

KM6551-0000: 0x1997 (6551<sub>dec</sub>)

### R9: Firmware version

Register R9 contains the ASCII coding of the terminal's firmware version, e.g. **0x3141** = '1A'. '0x31' corresponds to the ASCII character '1', '0x41' corresponds to the ASCII character 'A'.

This value cannot be changed.

### R12: Minimum data length of a channel

Bits 0 to 6 of the high-order byte specify the minimum number of output data in bits:  $000.0000_{bin} = 0_{dec}$ , hence 0 byte.

Bits 0 to 6 of the low-order byte specify the minimum number of input data in bits:  $001.1000_{bin} = 24_{dec}$ , hence 3 bytes.

The fact that bit 7 is set indicates that the control and status byte are not mandatory for the terminal function and are not transferred in compact mode.

### R16: DIP switch setting

The DIP switch setting is stored in register R16.

Value (hex)	String (ASCII)	Operation Mode
0x414D	MA	Master
0x4253	SB	Slave for broadcast reception
0x3153	S1	Slave with address 1
0x3253	S2	Slave with address 2
0x3353	S3	Slave with address 3
0x3453	S4	Slave with address 4
0x3553	S5	Slave with address 5
0x3653	S6	Slave with address 6
0x3753	S7	Slave with address 7
0xFFFF	-	Unknown DIP switch

### R31: Code word register

- If you write values into the user registers without first entering the user code word (0x1235) into the code word register, the terminal will not accept the supplied data.
- If you write values into the user registers and have previously entered the user code word (0x1235) in the code word register, these values are stored in the RAM registers and in the SEEPROM registers and are therefore retained when the terminal is restarted.

The code word is reset when the terminal is restarted.

### R32: Channel register (read/write)

IEEE 802.15.4 allows the use of one of 16 available channels. These frequency ranges work without influencing one another. The bandwidth of the signal is 2 MHz and the channel separation is 5 MHz. The channel should be selected such that it does not collide with WLAN or other systems that transmit in the

2.4 GHz range. Permitted settings in register 0...15.

It is permitted to change the channel during operation. The terminal displays the channel change in R33. The frequency channel has been accepted if R32 and R33 are identical.

### R33: Channel register (read only)

The terminal acknowledges the acceptance of the channel in this channel. When accepting the new channel in R32, the terminal confirms this by entering the channel in R33.

### R38: Watchdog (only activated in slave mode)

The watchdog for the slave is set in register R38. If the value is 0, the watchdog is deactivated. Therefore, no error bit will be set in the event of interrupted communication. The default value is 20<sub>dec</sub>. The value from R38 must be multiplied by approx. 20 ms.

Example: R38 = 100 (100 x 20 ms = 2000 ms or 2 sec).

### R39: Broadcast mode

Broadcast mode for the broadcast master is activated in register R39. The register is not evaluated by the broadcast slaves. The slaves are "made into" broadcast slaves via the DIP switch.

0x0000: Master-slave- or peer to peer mode (default)

0x4342: Broadcast mode (master)

### R40: Scan (only possible in master mode)

The slaves found are entered here if the scan has been activated with the bit CB1.2. Each bit represents a found slave address.

### R41: Network ID

Using the network ID you can distinguish between up to 255 radio networks. The KM6551-0000 only accepts telegrams from modules with the same network ID.

Permissible value range: 0...255<sub>dec</sub>. The default value is 0<sub>dec</sub>.

## ● Operation of several radio networks

**i** If you wish to operate different radio networks within the range (e.g. inside a factory hall), distinguishing the networks by the use of different channel numbers guarantees more freedom from interference than if you operate networks with different network IDs on the same channel! Use the network ID to distinguish between different radio networks only if no more free channels are available!

### R46: Attenuation of the transmission power

You can attenuate the transmission power of the KM6551-0 using bits 0000 to 7 of register R46. The setting only takes effect when the module is restarted.

<b>Bit</b>	15	14	13	12	11	10	9	8
<b>Name</b>	-	-	-	-	-	-	-	-

<b>Bit</b>	7	6	5	4	3	2	1	0
<b>Name</b>	Large Scale		Small Scale			-	-	-



**Key**

Bit	Name	Description	default		
15...8	-	reserved	00 <sub>bin</sub>		
7...6	Large Scale	11 <sub>bin</sub>	minus 30 dB		
		10 <sub>bin</sub>	minus 20 dB		
		01 <sub>bin</sub>	minus 10 dB		
		00 <sub>bin</sub>	minus 0.0 dB		
5...3	Small Scale	111 <sub>bin</sub>	minus 6.3 dB		
		110 <sub>bin</sub>	minus 4.9 dB		
		101 <sub>bin</sub>	minus 3.7 dB		
		100 <sub>bin</sub>	minus 2.8 dB		
		011 <sub>bin</sub>	minus 1.9 dB		
		010 <sub>bin</sub>	minus 1.2 dB		
		001 <sub>bin</sub>	minus 0.5 dB		
		000 <sub>bin</sub>	minus 0.0 dB		
		2...0	-	reserved	000 <sub>bin</sub>

## 8.5 Examples of Register Communication

The numbering of the bytes in the examples corresponds to the display without word alignment.

### 8.5.1 Example 1: reading the firmware version from Register 9

**Output Data**

Byte 0: Control byte	Byte 1: DataOUT1, high byte	Byte 2: DataOUT1, low byte
0x89 (1000 1001 <sub>bin</sub> )	0xXX	0xXX

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 not set means: reading the register.
- Bits 0.5 to 0.0 specify the register number 9 with 00 1001<sub>bin</sub>.
- The output data word (byte 1 and byte 2) has no meaning during read access. To change a register, write the required value into the output word.

**Input Data (answer of the Bus Terminal)**

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0x89	0x33	0x41

Explanation:

- The terminal returns the value of the control byte as a receipt in the status byte.

- The terminal returns the firmware version 0x3341 in the input data word (byte 1 and byte 2). This is to be interpreted as an ASCII code:
  - ASCII code 0x33 represents the digit 3
  - ASCII code 0x41 represents the letter A
 The firmware version is thus 3A.

## 8.5.2 Example 2: Writing to an user register



### Code word

In normal mode all user registers are read-only with the exception of Register 31. In order to deactivate this write protection you must write the code word (0x1235) into Register 31. If a value other than 0x1235 is written into Register 31, write protection is reactivated. Please note that changes to a register only become effective after restarting the terminal (power-off/power-on).

### I. Write the code word (0x1235) into Register 31.

#### Output Data

Byte 0: Control byte	Byte 1: DataOUT1, high byte	Byte 2: DataOUT1, low byte
0xDF (1101 1111 <sub>bin</sub> )	0x12	0x35

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 set means: writing to the register.
- Bits 0.5 to 0.0 specify the register number 31 with 01 1111<sub>bin</sub>.
- The output data word (byte 1 and byte 2) contains the code word (0x1235) for deactivating write protection.

#### Input Data (answer of the Bus Terminal)

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0x9F (1001 1111 <sub>bin</sub> )	0xXX	0xXX

Explanation:

- The terminal returns a value as a receipt in the status byte that differs only in bit 0.6 from the value of the control byte.
- The input data word (byte 1 and byte 2) is of no importance after the write access. Any values still displayed are invalid!

### II. Read Register 31 (check the set code word)

#### Output Data

Byte 0: Control byte	Byte 1: DataOUT1, high byte	Byte 2: DataOUT1, low byte
0x9F (1001 1111 <sub>bin</sub> )	0xXX	0xXX

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 not set means: reading the register.
- Bits 0.5 to 0.0 specify the register number 31 with 01 1111<sub>bin</sub>.
- The output data word (byte 1 and byte 2) has no meaning during read access.

**Input Data (answer of the Bus Terminal)**

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0x9F (1001 1111 <sub>bin</sub> )	0x12	0x35

Explanation:

- The terminal returns the value of the control byte as a receipt in the status byte.
- The terminal returns the current value of the code word register in the input data word (byte 1 and byte 2).

**III. Write to Register 32 (change contents of the feature register)**

**Output data**

Byte 0: Control byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0xE0 (1110 0000 <sub>bin</sub> )	0x00	0x02

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 set means: writing to the register.
- Bits 0.5 to 0.0 indicate register number 32 with 10 0000<sub>bin</sub>.
- The output data word (byte 1 and byte 2) contains the new value for the feature register.

**⚠ CAUTION**

**Observe the register description!**

The value of 0x0002 given here is just an example!

The bits of the feature register change the properties of the terminal and have a different meaning, depending on the type of terminal. Refer to the description of the feature register of your terminal (chapter *Register description*) regarding the meaning of the individual bits before changing the values.

**Input data (response from the Bus Terminal)**

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0xA0 (1010 0000 <sub>bin</sub> )	0xFF	0xFF

Explanation:

- The terminal returns a value as a receipt in the status byte that differs only in bit 0.6 from the value of the control byte.
- The input data word (byte 1 and byte 2) is of no importance after the write access. Any values still displayed are invalid!

**IV. Read Register 32 (check changed feature register)**

**Output Data**

Byte 0: Control byte	Byte 1: DataOUT1, high byte	Byte 2: DataOUT1, low byte
0xA0 (1010 0000 <sub>bin</sub> )	0xFF	0xFF

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 not set means: reading the register.
- Bits 0.5 to 0.0 indicate register number 32 with 10 0000<sub>bin</sub>.
- The output data word (byte 1 and byte 2) has no meaning during read access.

**Input Data (answer of the Bus Terminal)**

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0xA0 (1010 0000 <sub>bin</sub> )	0x00	0x02

Explanation:

- The terminal returns the value of the control byte as a receipt in the status byte.
- The terminal returns the current value of the feature register in the input data word (byte 1 and byte 2).

**V. Write Register 31 (reset code word)****Output Data**

Byte 0: Control byte	Byte 1: DataOUT1, high byte	Byte 2: DataOUT1, low byte
0xDF (1101 1111 <sub>bin</sub> )	0x00	0x00

Explanation:

- Bit 0.7 set means: Register communication switched on.
- Bit 0.6 set means: writing to the register.
- Bits 0.5 to 0.0 specify the register number 31 with 01 1111<sub>bin</sub>.
- The output data word (byte 1 and byte 2) contains 0x0000 for reactivating write protection.

**Input Data (answer of the Bus Terminal)**

Byte 0: Status byte	Byte 1: DataIN1, high byte	Byte 2: DataIN1, low byte
0x9F (1001 1111 <sub>bin</sub> )	0xFF	0xFF

Explanation:

- The terminal returns a value as a receipt in the status byte that differs only in bit 0.6 from the value of the control byte.
- The input data word (byte 1 and byte 2) is of no importance after the write access. Any values still displayed are invalid!

# 9 Appendix

## 9.1 General operating conditions

### Approval and use

The KM6551-0000 module meets the requirements of EN 300-440-02 V1.1.2.

Operation of the KM6551-0000 module is permitted in all EU countries as well as in Switzerland and Liechtenstein. Other countries on enquiry.

The use of the KM6551-0000 is permitted with the following antennas:

Name	Description
<a href="#">ZS6100-0900 [► 28]</a>	Directional antenna (gain 9 dBi), without cable
<a href="#">ZS6100-1800 [► 30]</a>	Directional antenna (gain 18 dBi), without cable
<a href="#">ZS6200-0400 [► 32]</a>	Omnidirectional antenna (gain 4 dBi), without cable
<a href="#">ZS6201-0410 [► 34]</a>	Rod antenna (gain 4 dBi), with cable (1 m)
<a href="#">ZS6201-0500 [► 36]</a>	Rod antenna (gain 5 dBi), without cable

### NOTE

#### CE conformity

The CE conformity of the KM6551-0000 is only guaranteed if it is operated with original Beckhoff accessories (antennas, [coaxial cable \[► 19\]](#))!

### Environmental conditions

The following conditions must be met in order to ensure flawless operation of the fieldbus components.

### Operation

The components may not be used without additional protection in the following locations:

- in difficult environments, such as where there are corrosive vapors or gases, or high dust levels
- in the presence of high levels of ionizing radiation

Condition	Permissible range
Permissible ambient temperature during operation	0°C ... +55°C
Permissible ambient temperature during operation	-25°C ... +85°C
Installation position	variable
Vibration resistance	conforms to EN 60068-2-6
Shock resistance	conforms to EN 60068-2-27, EN 60068-2-29
EMC immunity	conforms to EN 61000-6-2
Emission	conforms to EN 61000-6-4, EN 300-440-02 V1.1.2
Safety of persons in electromagnetic fields	conforms to EN 50371:2002

**Transport and storage**

Condition	Permissible range
Permissible ambient temperature during storage	-25°C... +85°C
Relative humidity	95 %, no condensation
Free fall	up to 1 m in the original packaging

**Protection classes and types**

Condition	Permissible range
Protection class in accordance with IEC 536 (VDE 0106, Part 1)	A protective conductor connection to the profile rail is necessary!
Protection class conforms to IEC 529	IP20 (protection against contact with a standard test finger)
Protection against foreign objects	Less than 12 mm in diameter
Protection against water	no protection

## 9.2 EC declaration of conformity

**BECKHOFF** New Automation Technology

### EG-Konformitätserklärung, EC Declaration of Conformity

<b>Hersteller</b> <i>Manufacturer</i>	<b>Beckhoff Automation GmbH</b>
<b>Anschrift</b> <i>Address</i>	Eiserstr. 5 33415 Verl Bundesrepublik Deutschland
<b>Produktbezeichnung</b> <i>Product description</i>	<b>KM6551 Wireless-Datenaustauschklemme</b> KM6551 Wireless data exchange terminal

Die hier genannten Baugruppen sind entwickelt, konstruiert und gefertigt in Übereinstimmung mit den EG-Richtlinien 1999/5/EG R&TTE-Richtlinie, 2004/108/EG EMV-Richtlinie und 2006/95/EG Niederspannungsrichtlinie.

**Folgende Normen wurden angewandt:**

*The components mentioned herein have been developed, designed and manufactured in accordance with the EC Guideline 1999/5/EG, 2004/108/IEC and 2006/95/IEC. The following standards have been used:*

<b>Generic Standard: EN 61000-6-2:2006</b> <i>Generic Standard: EN 61000-6-2:2006</i>	<b>Störfestigkeit für Industriebereich</b> <i>immunity for industrial environments</i>
<b>Generic Standard: EN 61000-6-4:2007</b> <i>Generic Standard: EN 61000-6-4:2007</i>	<b>Störaussendung für Industriebereich</b> <i>emission standard for industrial environments</i>
<b>Standard: EN 300 440-2:V1.1.2</b> <i>Standard: EN 300 440-2:V1.1.2</i>	<b>EMV und Funk Spektrumangelegenheiten (ERM) – Funkanlagen mit geringer Reichweite</b> <i>EMC and radio spectrum matters (ERM) – short range devices</i>
<b>Generic Standard: EN 50371:2002</b> <i>Generic Standard: EN 50371:2002</i>	<b>Sicherheit von Personen in elektromagnetischen Feldern</b> <i>human exposure to radio frequency electromagnetic fields</i>

Verl, den / the 04.03.2009

**Unterschrift, signature**  
*Name, name*  
**Funktion, function**



\_\_\_\_\_  
**Hans Beckhoff**  
Geschäftsführer, Executive Director

Fig. 44: EC declaration of conformity

## 9.3 Calculating with decibels

In communication technology power is expressed in decibels (dB), a tenth of the unit Bel. It is the logarithmic ratio between two quantities with the same unit.

A reference variable (P1), e.g. a milliwatt (mW) is compared with the measured variable (P2). The logarithmic correlation was discovered by Alexander Graham Bell, in whose honor the unit Bel was named.

Since the number values would be too unwieldy if the Bel was used, it was agreed to use 1/10 of the value, i.e. the decibel.

Definition of the level difference: Level difference [dB] =  $10 \log ([P1] / [P2])$ .

Definition of a power ratio: power ratio =  $10^{\text{level difference}/10}$

The advantage of expressing the powers and losses (attenuations) in dB is that the calculation method for power ratios can be replaced by a lower calculation method for the dB calculation.

Power ratio	dB calculation
Multiplication or division	Addition or subtraction
Exponent	Factor

### Examples of power ratios

Factor	Amplification [dB]
x 1	+0 dB
x 1.25	+1 dB
x 2	+3 dB
x 4	+6 dB
x 10	+10 dB
x 16	+12 dB
x 100	+20 dB
x 1000	+30 dB

Factor	Attenuation [dB]
x 1	-0 dB
x 0.8	-1 dB
x 0.5	-3 dB
x 0.25	-6 dB
x 0.1	-10 dB
x 0.6	-12 dB
x 0.01	-20 dB
x 0.001	-30 dB

### Examples of calculations with decibels

Change	in dB
$10 / 2 = 5$	$10 - 3 = 7$
$2 \times 2 \times 2 = 8$	$3 + 3 + 3 = 9$
$2 \times 100 = 200$	$3 + 20 = 23$
$1000 / 2 = 500$	$30 - 3 = 27$

## 9.4 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.



**Beckhoff's branch offices and representatives**

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages: <https://www.beckhoff.com>

You will also find further documentation for Beckhoff components there.

**Beckhoff Support**

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline: +49 5246 963 157  
Fax: +49 5246 963 9157  
e-mail: [support@beckhoff.com](mailto:support@beckhoff.com)

**Beckhoff Service**

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline: +49 5246 963 460  
Fax: +49 5246 963 479  
e-mail: [service@beckhoff.com](mailto:service@beckhoff.com)

**Beckhoff Headquarters**

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20  
33415 Verl  
Germany

Phone: +49 5246 963 0  
Fax: +49 5246 963 198  
e-mail: [info@beckhoff.com](mailto:info@beckhoff.com)  
web: <https://www.beckhoff.com>

## List of figures

Fig. 1	KM6551-0000 - Terminal module for radio transmission .....	8
Fig. 2	Basic Function Principles.....	10
Fig. 3	Peer to Peer.....	11
Fig. 4	Master-Slave mode.....	11
Fig. 5	Broadcast mode.....	12
Fig. 6	KM6551 - LED displays .....	12
Fig. 7	DIP switch.....	13
Fig. 8	Channels 11 to 26.....	14
Fig. 9	Utilizing gaps between adjacent WLAN networks .....	15
Fig. 10	KM6551 dimensions .....	19
Fig. 11	Omnidirectional antennas .....	23
Fig. 12	Directional antennas .....	23
Fig. 13	Mixed operation .....	23
Fig. 14	Fresnel zone .....	25
Fig. 15	Radius $r$ of the Fresnel zone in relationship to the distance $s$ .....	25
Fig. 16	Two omnidirectional antennas .....	26
Fig. 17	Omnidirectional antenna combined with a directional antenna .....	26
Fig. 18	Two directional antennas.....	27
Fig. 19	ZS6100-0900 - Directional antenna .....	28
Fig. 20	ZS6100-0900 - Azimuth and Elevation for 2400 MHz .....	28
Fig. 21	ZS6100-1800 - Directional antenna with large gain.....	30
Fig. 22	ZS6100-1800 - Azimuth and Elevation for 2400 MHz .....	30
Fig. 23	ZS6200-0400 - Omnidirectional antenna.....	32
Fig. 24	ZS6200-0400 - Azimuth and Elevation for 2400 MHz .....	32
Fig. 25	ZS6201-0410 - Rod antenna with cable .....	34
Fig. 26	ZS6201-0410 - Azimuth and Elevation for 2400 MHz .....	34
Fig. 27	ZS6201-0500 - Rod antenna .....	36
Fig. 28	ZS6201-0500 - Azimuth and Elevation for 2400 MHz .....	36
Fig. 29	Function block FB_KM6551_MAIN.....	45
Fig. 30	Function block FB_KM6551_Master_10Byte .....	47
Fig. 31	Function block FB_KM6551_Slave_10Byte .....	48
Fig. 32	Function block FB_KM6551_Masterbroadcast_10Byte.....	50
Fig. 33	Function block FB_KM6551_Slavebroadcast_10Byte.....	51
Fig. 34	KS2000 configuration software.....	52
Fig. 35	Display of the fieldbus station in KS2000 .....	54
Fig. 36	KS2000 branch for channel 1 of the KM6551 .....	54
Fig. 37	Dialog screen for parameterizing the KM6551 .....	55
Fig. 38	Register view in KS2000.....	56
Fig. 39	Process Data field.....	57
Fig. 40	History field .....	57
Fig. 41	Value field .....	57
Fig. 42	Value field .....	58
Fig. 43	Setting the display .....	58
Fig. 44	EC declaration of conformity.....	71



Beckhoff Automation GmbH & Co. KG  
Hülshorstweg 20  
33415 Verl  
Germany  
Phone: +49 5246 9630  
[info@beckhoff.com](mailto:info@beckhoff.com)  
[www.beckhoff.com](http://www.beckhoff.com)