KL-5101

Technical Documentation
Incremental Encoder Interface

Please keep for further use !

Edition date/Rev. date:	05.03.1998
Document no./Rev. no.:	TRS - V - BA - GB - 0101 - 00
Software version:	1.0
File name:	TRS-V-BA-GB-0101.DOC
Author:	KOH

TRSystemtechnik GmbH
Eglishalde 6
D-78647 Trossingen
Germany
Tel. +49 - (0) 7425 / 228-0
Fax +49 - (0) 7425 / 228-34

Imprint

TRSystemtechnik GmbH
D-78647 Trossingen
Eglishalde 6
Tel.: (++49) 07425/228-0
Fax: (++49) 07425/228-34

© Copyright 1998 TRSystemtechnik

Guarantee

In our ongoing efforts to improve our products, TRSystemtechnik reserve the right to alter the information contained in this document without prior notice.

Printing

This manual was edited using text formatting software on a DOS personal computer. The text was printed in *Arial*.

Fonts

Italics and **bold** type are used for the title of a document or to emphasize text passages.

Passages written in `Courier` show text which is visible on the display as well as software menu selections.

"< >" refers to keys on your computer keyboard (e.g. <RETURN>).

Note

Text following the "NOTE" symbol describes important features of the respective product.

Copyright Information ©

MS-DOS is a registered trademark of Microsoft Corporation.

Revision History

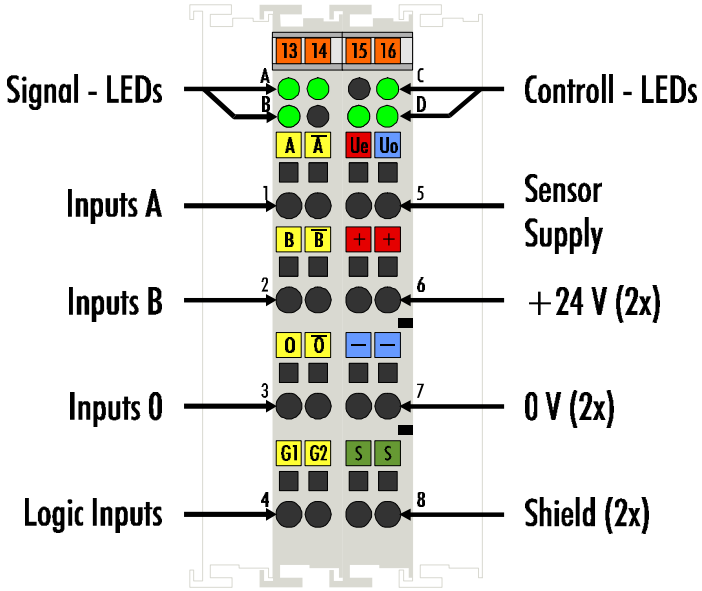
Note:
The cover of this document shows the current revision status and the corresponding date. Since each individual page has its own revision status and date in the footer, there may be different revision statuses within the document.

Document created: 05.03.1998

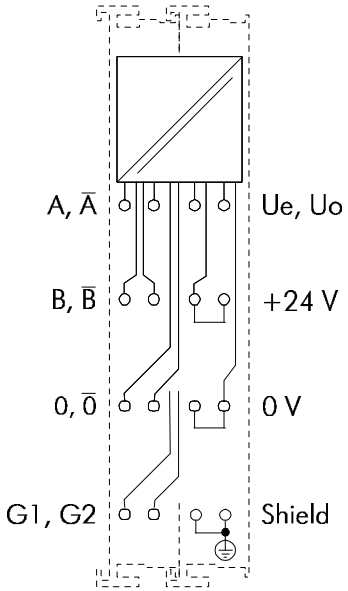
Revision	Date

Table of contents

Incremental Encoder Interface KL5101



Top View



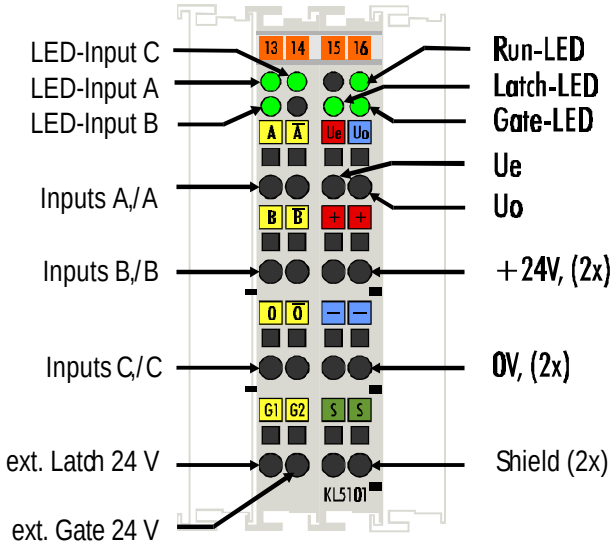
Contact Assembly

Technical data	KL5101
Sensor connectionn	A, A(inv), B, B(inv), C, C(inv)
Sensor operating voltage	5 V DC (please enquire about 15 V DC)
Counter	16-bit binary
Limit frequency	1 MHz (4-fold evaluation)
Quadrature decoder	1-2-4-fold evaluation
Zero pulse latch	16-bit
Commands	read, set, activate
Supply voltage	24 V DC (20 V ... 29 V)
Power contact current consumption	0.1 A (without sensor load current)
Bit width in the process image	48 I/Os: 2 x 16 bit data, 1 x 8 bits control/status, 1 x 8 bits not used
K-bus current consumption	25 mA
Weight approximately	85 g
Operating temperature	0°C ... +55°C
Storage temperature	-25°C ... +85°C
relative Feuchte	95% no condensation
Vibration/shock resistance	In accordance with IEC 68-2-6 / IEC 68-2-27
EMC resistance Burst / ESD	In accordance with EN 61000-4-4 / EN 61000-4-2 limits in accordance with EN 50082-2
Installation position	any
Degree of protection	IP20

Description of functions

The incremental encoder interface terminal KL5101 enables the connection of any incremental encoders to the bus coupler or the PLC. A 16-bit counter with a quadrature decoder and a 16-bit latch can be read, set or activated. Besides the decoder imputs A, B, C, an additional latch imput G1 (24 V) and a gate imput G2 (24 V) for disabling the counter are available. The 16-bit up / down counter mode can also be selected. In this mode of operation, input B is the counting input.
1-fold, 2-fold or 4-fold evaluation of the encoder signals A, B, C in simple or complementary form can be parametrized via the field bus. The settings specific to the terminal are stored permanently in a serial EEPROM and are thus protected against power failures.
The terminal is supplied as a 4-fold quadrature decoder with complementary evaluation of the encoder signals A, B, C. For operation of the encoder interface, the operating voltage of 24 V DC must be connected to the terminal contacts in addition to the encoder inputs.

Assignments of terminal contacts



Inputs A,/A:
Pulse input in the terminal's encoder and counter mode.

Inputs B,/B:
Phase-shifted pulse input in the terminal's encoder mode.

Counting direction input in the terminal's counter mode.
Counting direction:
+ 5V or open contact: up
0V: down

Inputs C,/C:
Zero point pulse input for the terminal's latch register.
This input is activated via the EN_LATC bit in the terminal's control byte.

External Latch 24V
Additional latch input of the terminal.
This input is activated via the EN_LAT_EXT bit in the terminal's control byte.
The counter value is latched when this input is alerted and an edge change takes place from 0 V to 24 V.

External Gate 24 V
A high level at this contact suppresses counting by the terminal.

Ue
Voltage supply for the encoder (+5 V).
Uo
Voltage supply for the encoder (0 V).

0V, 24 V
A supply of 0 V and 24 V voltage must be applied to these contacts for operation of the terminal.

LED display

The signal LED's indicate the status of the encoder inputs A, B, C and of the logic inputs of the gate and of the additonal external latch. The RUN LED indicates cyclic data transfer with the higher-level controller. The RUN LED goes off if no process data is exchanged for 100 ms.

Process data

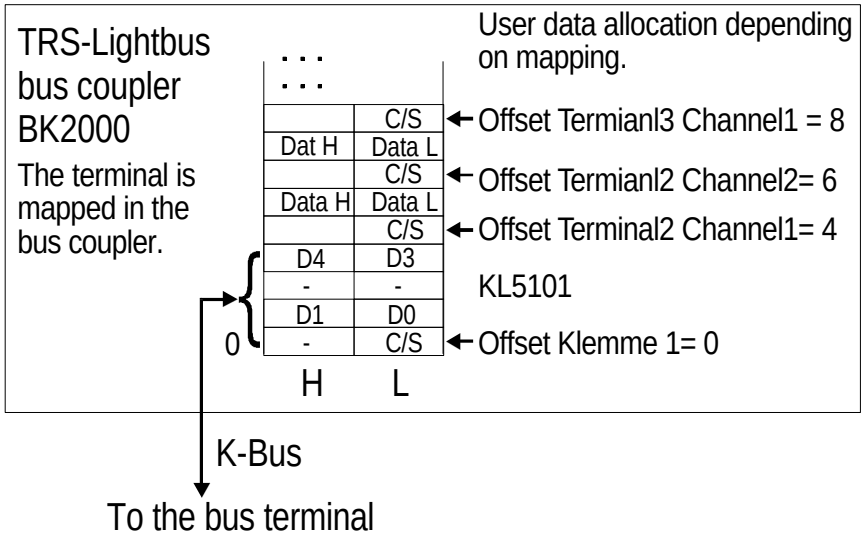
The KL5101 always occupies 6 bytes of input data and 6 bytes of output data. The control / status byte is at the least significant byte offset.
The data word D0/D1 contains the counter value (read/set) and
the data word D3/D4 contains the latch word (read).

Terminal configuration

The terminal can be configured and parametrized via the internal register structure.
Each terminal channel is mapped in the bus coupler. The data of the terminal is mapped differently in the memory of the bus coupler depending on the type of the bus coupler and on the set mapping configuration (eg Motorola/ intel format, word alignment,...). Contrary to the analog input and output terminals, in the case of the KL5101 the control and status byte is always mapped regardless of the higher-level field bus system.

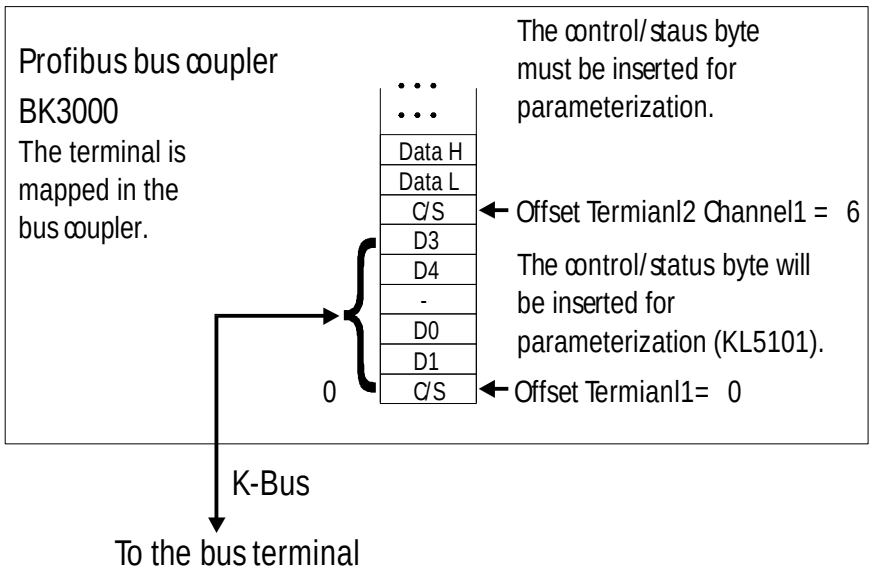
TRS Lightbus coupler BK2000

In the case of the TRS Lightbus coupler BK2000, the control /status byte is also always (ie in the case of all analog terminals) mapped in addition to the data bytes.It is always in the low byte at the offset address of the terminal channel.



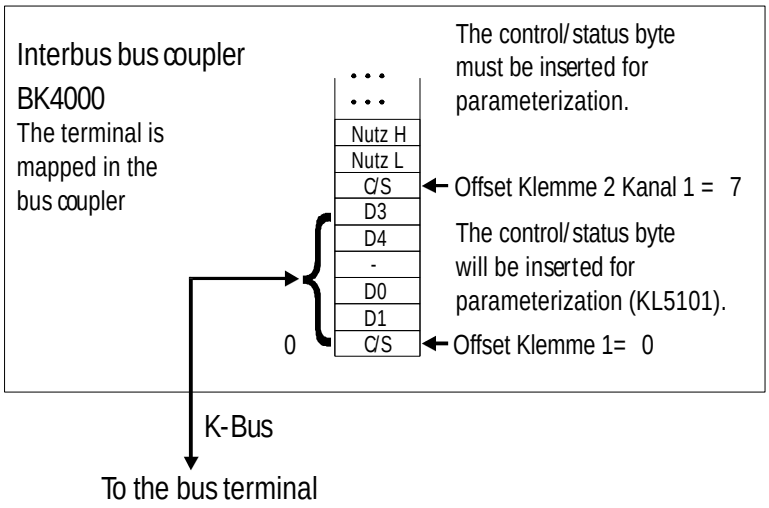
Profibus coupler BK3000

In the case of the Profibus coupler BK3000, the Kl5101 is always mapped with 6 bytes of input data and 6 bytes of output data.



Interbus coupler BK4000

By default, the Interbus coupler BK4000 maps the Kl5101 with 6 bytes of input data and 6 bytes of output data.



Other bus couplers and further information

You will find further information on the mapping configuration of bus couplers in the annex of the respective bus coupler manual and under the heading of "Configuration of Masters".

Reference

The annex contains an overview of possible mapping configurations depending on the parameters that can be set.

Parametrization with the KS2000 software

Independently of the field bus system, parameters can be set via the serial configuration interface in the bus coupler using the TRS KS2000 configuration software.

Register Communication KL5101

General register description

Complex terminals that possess a processor are capable of bidirectionally exchanging data with the higher-level control system. Below, these terminals are referred to as intelligent bus terminals. They include the analog inputs (0-10V, -10-10V, 0-20mA, 4-20mA), the analog outputs (0-10V, -10-10V, 0-20mA, 4-20mA), serial interface terminals (RS485, RS232, TTY, data transfer terminals), counter terminals, the encoder interface, the SSI interface, the PWM terminal and all other parametrizable terminals.

Internally, all intelligent terminals possess a data structure that is identical in terms of its essential characteristics. This data area is organized in words and embraces 64 memory locations. The essential data and parameters of the terminal can be read and adjusted by way of this structure. Function calls with corresponding parameters are also possible. Each logical channel of an intelligent terminal has such a structure (therefore, 4-channel analog terminals have 4 register sets).

This structure is broken down into the following areas:
(You will find a list of all registers at the end of this documentation).

Area	Address
Process variables	0-7
Type-register	8-15
Manufacturer parameters	16-31
User parameters	32-47
Extended user area	48-63

Process variables

R0 - R7 Registers in the terminal's internal RAM

The process variables can be used in addition to the actual process image and their functions are specific to the terminal.

R0 - R5: These registers have a function that depends on the terminal type.

R6: Diagnostic register

The diagnostic register may contain additional diagnostic information. In the case of serial interface terminals, for example, parity errors that have occurred during data transfer are indicated.

R7: Command register

High-Byte_Write = function parameter
Low-Byte _Write = function number
High-Byte _Read = function result
Low-Byte_ Read = function number

Register type

R8-R15 Registers in the terminal`s internal ROM

From the producer the type- and system parameters are programmed fixed. That means that this parameters are not changeable. The user can only read the parameter.

R8: Terminal type:

The terminal type in register R8 is used for the identification.

R9: Software version X.y

The software version can be read as an ASCII string.

R10: Data length

R10 contains the number of the multiplexed shifting registers and the length in bit.
The bus coupler recognized this structure.

R11: Signal channels

In comparison to R10 register R11 contains the number of the logical available channels. So for example it is possible that a physical available shifting register can be consisted of sevaral signal channels.

R12: Minimum data length

The respective byte contains the minimum transmitting data length of a channel. Is the MSB is set the status byte is cancelled.

R13: Data type registers

Data type register	
0x00	Terminal without valid data type
0x01	Byte array
0x02	1 byte n bytes structure
0x03	Word array
0x04	1 byte n word structure
0x05	Double word array
0x06	1 byte n double words structure
0x07	1 byte 1 double word structure
0x08	1 byte 1 double word structure
0x11	Byte array with a variable logical channel length

Data type register	
0x12	1 byte n bytes structure with a variable logical channel length (eg 60xx)
0x13	Word-array with a variable logical channel length
0x14	1 byte n words structure with a variable logical channel length
0x15	Double word array with a variable logical channel length
0x16	1 byte n double words structure with a variable logical channel length

R14: not used

R15: Alignment bits (RAM)

The analog terminal is set to a byte limit in the terminal bus with the alignment bits.

Manufacturer parameters

R16 - R30 is the area of the " Manufacturer Parameters" (SEEROM)

The manufacturer parameters are specific to each terminal type. They are programmed by the manufacturer, but can also be modified from the control system. The manufacturer parameters are stored permanently in a serial EEPROM in the terminal and are therefore not destroyed by power failures.
These registers can only be modified after setting a code word in R31.

User parameters

R31 - R47 "Application Parameters" area (SEEROM)

The application parameters are specific to each terminal type. They can be modified by the programmer. The application parameters are stored permanently in a serial EEPROM in the terminal and cannot be destroyed by power failures. From software version 2.A, the user area is write-protected by way of a code word.

R31: Code word register in the RAM

The code word 0x1235 must be entered here to enable modification of parameters in the user area. Write protection is set if a different value is entered in this register. When write protection is inactive, the code word is returned during reading of the register. The register contains the value zero when write protection is active.

R32: Feature register

This register defines the operating modes of the terminal. For example, a user-specific scaling can be activated for the analog I/Os.

R33 - R47

Registers that depend on the terminal type.

Extended application area

R47-R63

These registers have not yet been implemented.

Register access via proces data transfer

Bit 7=1: Register mode

When bit 7 of the control byte is set, the first two bytes of the user data are not used for process data transfer, but are written into or read out of the terminal's register set.

Bit 6=0: read
Bit 6=1: write

In bit 6 of the control byte, you define whether a register is to be read or written. When bit 6 is not set, a register is read without modification. The value can be taken from the input process image.
When bit 6 is set, the user data is written into a register. The operation is

concluded as soon as the status byte in the input process image has assumed the same value as the control byte in the output process image.

Bits 0 to 5: address

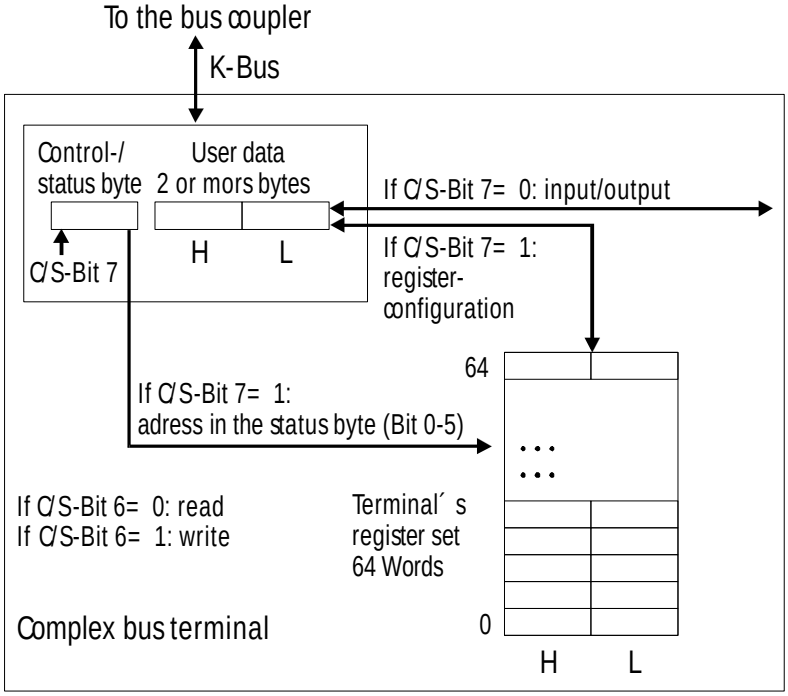
The address of the register to be addressed is entered in bits 0 to 5 of control byte.

Control byte in the register mode

MSB

REG=1	W/NR	A5	A4	A3	A2	A1	A0

REG = 0 : Process data transfer
REG = 1 : Access to register structure
W/NR = 0 : Read register
W/NR = 1 : Write register
A5..A0 = Register address
A total of 64 registers can be addressed with the addresses A5...A0.



The control or status byte occupies the lowest address of a logical channel. The corresponding register values are located in the following 2 data bytes (the BK2000 is an exception to this rule: here, an unused data byte is inserted after the control or status byte, thus setting the register value to a word limit).

Example

Reading register 8 in the BK2000 with a KL3022 and the end terminal.

If the following bytes are transferred from the controller to the terminal,

Byte0	Byte1	Byte2	Byte3
0x88	0xXX	0xXX	0xXX

the terminal returns the following type designation (0xBCE corresponds to the unsigned interger 3022).

Byte0	Byte1	Byte2	Byte3
0x88	0x00	0xCE	0x0B

A further example

Writing register 31 in the BK2000 with an intelligent terminal and the end terminal.

If the following bytes (user code word) are transferred from the controller to the terminal,

Byte0	Byte1	Byte2	Byte3
0xDF	0xXX	0x12	0x35

the user code word is set and the terminal returns the register address with the bit 7 for register access as the acknowledgement.

Byte0	Byte1	Byte2	Byte3
0x9F	0x00	0x00	0x00

Terminal-specific register description
Application parameters

R32: Feature register: [0x2200]

Feature Bit No.		Mode description
Bit 0 - 8	-	not used
Bit 9	0/1	0: Simple evaluation of the signals A, B, C 1: complementary evaluation of the signals A, /A, B, /B, C, /C [1]
Bit 10 Bit 11	0 0	4-fold evaluation of the encoder signals A,B,C, i.e. both rising and falling edges of the encoder signals A, B are counted. [00]
	0 1	1-fold evaluation of the encoder signals A, B, C. i.e. every period of the encoder signal A is counted.
	1 0	2-fold evaluation of the encoder signals A, B, C, i.e. every edge of the encoder signal A is counted.
	1 1	4-fold evaluation of the encoder signals A, B, C
Bit 12	1	FAST_M The fast mode is permanently set (see control byte) [0]
Bit 13	0/1	0: The encoder data is loaded on a free-running basis 1: The data is loaded synchronously to the read cycle of the terminal bus (jitter < 15 µs) [1]
Bit 14	-	not used
Bit 15	0/1	0: Encoder interface. [0] 1: Counter mode is activated. 16-bit up/down counter Input A: Counter Input B: Counting direction (5 V or open = up, 0 V = down) Input C: Latch

CONTROL byte in process transfer

The control byte is transferred from the controller to the terminal. It can be used in the register mode (REG = 1) or in process data transfer (REG = 0). Various actions are triggered in the the KL5101 with the control byte:

MSB

REG=0		CFAST_M			CNT_SET	EN_ LAT_EXT	EN_LATC

Bit	Function
CFAST_M	In the fast mode, only the counting function of the encoder is executed, i.e. the following control bits CNT_SET, EN_LAT_EXT and EN_LATC are ignored by the terminal. This bit is combined logically (OR) with the FAST_M-bit in the feature register R32.
CNT_SET	The counter is set to the value that is specified via the process data with the rising edge of CNT_SET.
EN_LAT_EXT	The external latch input is activated. With the first external latch impulse after validity of the EN_LAT_EXT bit, the counter value in the latch register is stored. The pulses that follow have no influence on the latch register when the bit is set. Attention must be paid to ensuring that the corresponding latch valid bit (LAT_EXT_VAL) has been removed from the terminal before alerting of the zero pulse.
EN_LATC	The zero point latch (C input) is activated. The counter value is stored in the latch register with the first external latch pulse after validity of the EN_LATC bit (this has priority over EN_LAT_EXT). The pulses that follow have no influence on the latch register when the bit is set. Attention must be paid to ensuring that the corresponding latch valid bit (LATC_VAL) has been removed from the terminal before the zero pulse is alerted (the LATC_VAL bit cannot be removed from the terminal until the C pulse has a low level).

STATUS byte in process data transfer

The status byte is transferred from the terminal to the controller. The status byte contains various status bits of the KL5101.

MSB

REG=0			OVERFLOW	UNDERFLOW	CNTSET_ ACC	LAT_EXT_ VAL	LATC_VAL

Bit	
OVERFLOW	This bit is set if an overflow (65535 to 0) of the 16-bit counter occurs. It is reset when the counter exceeds a third of the measurement range (21845 to 21846) or as soon as an underflow occurs.
UNDERFLOW	This bit is set if an underflow (0 to 65535) of the 16-bit counter occurs. It is reset when the counter drops below two thirds of the measurement range (43690 to 43689) or as soon as an overflow occurs.
CNTSET_ACC	The data for setting the counter has been accepted by the terminal.
LAT_EXT_VAL	An external latch pulse has occurred. The data D2,D3 in the process image corresponds to the latched value when the bit is set. To activate the latch input again, EN_LAT_EXT must first be removed and then set again.
LATC_VAL	A zero point latch has occurred. The data D2,D3 in the process image corresponds to the latched value when the bit is set. To activate the latch input again, EN_LATC must first be removed and then set again.

Annex

As already described in the chapter on terminal configuration, each bus terminal is mapped in the bus coupler. In the standard case, this mapping is done with the default setting in the bus coupler / bus terminal. This default setting can be modified with the TRS configuration software KS2000 or using master configuration software (eg ComProfibus). The following tables provide information on how KL5101,maps itself in the bus coupler depending on the set parameters.

Mapping in the bus coupler

The KL5101 is mapped in the bus coupler depending on the set parameters. The terminal is always evaluated completely, the terminal occupies memory space in the process image of the input and outputs.

		I/O Offset	High Byte	Low Byte
Complete evaluation	= X	3		
MOTOROLA format	= 0	2	D4	D3
Word alignment	= 0	1	-	D1
		0	D0	CT/ST

		I/O Offset	High Byte	Low Byte
Complete evaluation	= X	3		
MOTOROLA format	= 1	2	D3	D4
Word alignment	= 0	1	-	D0
		0	D1	CT/ST

		I/O Offset	High Byte	Low Byte
Complete evaluation	= X	3	D4	D3
MOTOROLA format	= 0	2	-	-
Word alignment	= 1	1	D1	D0
		0	-	CT/ST

		I/O Offset	High Byte	Low Byte
Complete evaluation	= X	3	D3	D4
MOTOROLA format	= 1	2	-	-
Word alignment	= 1	1	D0	D1
		0	-	CT/ST

Legend

Complete evaluation: The terminal is mapped with control / status byte.
Motorola format: The Motorola or Intel formal can be set.
Word alignment: The terminal is at a word limit in the bus coupler.
CT: Control- Byte (appears in the PI of the outputs).
ST: Status- Byte (appears in the PI of the inputs).
D0/D1: Counter word (read/set)
D3/D4: Latch word (read)

Table of the register set
of the KL5101

Address	Description	Default value	R/ W	Storage medium
R0	not used	0x0000	R	
R1	not used	0x0000	R	
R2	not used	0x0000	R	
R3	not used	0x0000	R	
R4	not used	0x0000	R	
R5	not used	0x0000	R	
R6	Diagnostic register – not used	0x0000	R	
R7	Command register - not used	0x0000	R	
R8	Terminal type	5101	R	ROM
R9	Software version number	0x????	R	ROM
R10	Multiplex shift register	0x0218	R	ROM
R11	Signal channels	0x0130	R	ROM
R12	Minimum data length	0x3030	R	ROM
R13	Data structure	0x0000	R	ROM
R14	not used	0x0000	R	
R15	Alignment register	variable	R/W	RAM
R16	Hardware version number	0x????	R/W	SEEROM
R17	not used	0x0000	R/W	SEEROM
R18	not used	0x0000	R/W	SEEROM
R19	not used	0x0000	R/W	SEEROM
R20	not used	0x0000	R/W	SEEROM
R21	not used	0x0000	R/W	SEEROM
R22	not used	0x0000	R/W	SEEROM
R23	not used	0x0000	R/W	SEEROM
R24	not used	0x0000	R/W	SEEROM
R25	not used	0x0000	R/W	SEEROM
R26	not used	0x0000	R/W	SEEROM
R27	not used	0x0000	R/W	SEEROM
R28	not used	0x0000	R/W	SEEROM
R29	not used	0x0000	R/W	SEEROM
R30	not used	0x0000	R/W	SEEROM
R31	Code word register	variable	R/W	RAM
R32	Feature register	0x2200	R/W	SEEROM
R33	not used	0x0000	R/W	SEEROM
R34	not used	0x0000	R/W	SEEROM
R35	not used	0x0000	R/W	SEEROM
R36	not used	0x0000	R/W	SEEROM
R37	not used	0x0000	R/W	SEEROM
R38	not used	0x0000	R/W	SEEROM
R39	not used	0x0000	R/W	SEEROM
R40	not used	0x0000	R/W	SEEROM
R41	not used	0x0000	R/W	SEEROM
R42	not used	0x0000	R/W	SEEROM
R43	not used	0x0000	R/W	SEEROM
R44	not used	0x0000	R/W	SEEROM
R45	not used	0x0000	R/W	SEEROM
R46	not used	0x0000	R/W	SEEROM
R47	not used	0x0000	R/W	SEEROM