KL-5051

Technical Documentation
BI – SSI Sensor Interface

Please keep for further use !

Edition date/Rev. date:	27.07.1998
Document no./Rev. no.:	TRS - V - BA - GB - 0125 - 00
Software version:	1.0
File name:	TRS-V-BA-GB-0125.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

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

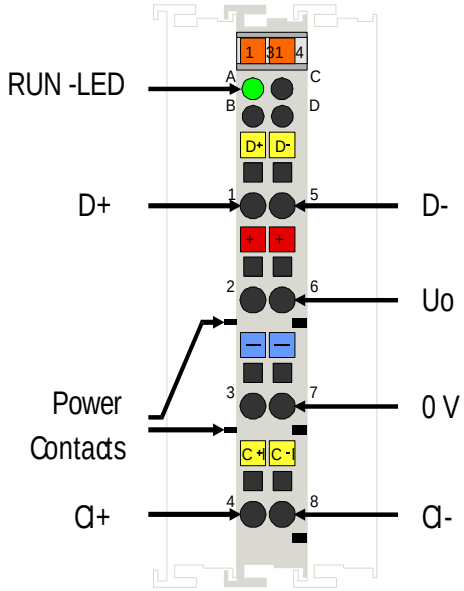
Note:
The cover of this document shows the current revision status and the corresponding date. Since each individual page has its own revision status and date in the footer, there may be different revision statuses within the document.

Document created: 27.07.1998

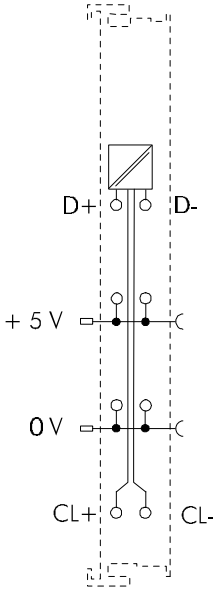
Revision	Date

Table of contents

SSI Sensor Interface KL-5051



Top View



KL 5051

Technical data	KL-5051
Sensor connection	binary input: D+, D-; binary output: Cl+, Cl-
Power supply	5 V DC via power contacts
Current consumption	typically 85 mA without sensor
Sensor supply	5 V DC
Data transfer rate	variable up to 1 MHz, 625 kHz default
Serial input	24 bit width (variable)
Data direction	read/write
Signal output	difference signal (RS422)
Signal input	difference signal (RS422)
Electrical isolation	Signal input via optocoupler, 500 V (T-Bus / field voltage)
Bit width in the process image	32 bits: 24 bits data, 8 bits control/status
Weight approx..	80 g
Operating temperature	0°C ... +55°C
Storage temperature	-25°C ... +85°C
Relative humidity	95%, no condensation
Vibration/shock resistance	conforms to IEC 68-2-6 / IEC 68-2-27
EMC resistance Burst / ESD	conforms to EN 61000-4-4 / EN 61000-4-2, limit EN 50082-2
Installation position	any
Type of protection	IP20

Description of functions

The BI-SSI interface terminal KL-5051 serves to link the digital digifas®7100/7200 servo amplifier from the Seidel company to the bus coupler or the controller. The interface consists of two logical channels. The drive is positioned via the first channel. With the second channel, release signals are set, parameter data is transferred and status information or parameter values are read
It is therefore possible for a subordinate access controller to accept the positioning information on Channel 1 while the higher-level controller sets release signals on Channel 2 and performs the monitoring tasks.

LED display

The Run LED indicates the operating state of the terminal.
On – normal operation

Off – watchdog timer overflow has occurred. The green LED goes off if no process data is transferred by the bus coupler for 100 ms.

Com error
On – communication fault, e.g. wire breakage on the data/clock line

Off – normal operation

Alarm
On – the connected device has sent a malfunction signal.

Off – normal operation

Process data

The KL-5051 is mapped with 6 bytes of input data and 6 bytes of output data. A0, A1, A2 and E0, E1, E2 constitute the channel for setting and detecting the operating data of the servo.
A3, A4, A5 and E3, E4, E5 constitute the channel for detecting the servo status and for setting the servo control. It is also used to parametrise the servo.

Byte	Function	Byte	Function
A0	Operating data-control	E0	Operating data status
A1	Speed setpoint	E1	Actual position
A2	Speed setpoint	E2	Actual position
A3	Parameter-control	E3	Parameter status
A4	Parameter/control-servo	E4	Parameter/status-servo
Á5	Parameter/control-servo	E5	Parameter/status-servo

In A1, A2, the speed setpoint is specified as a 16-bit signed integer.
Refer to the manual of the servo for details of the maximum setpoint inputs.

The absolute actual position is located in E1, E2 as a 16-bit unsigned integer. The actual position has a resolution of 65536 increments per revolution.

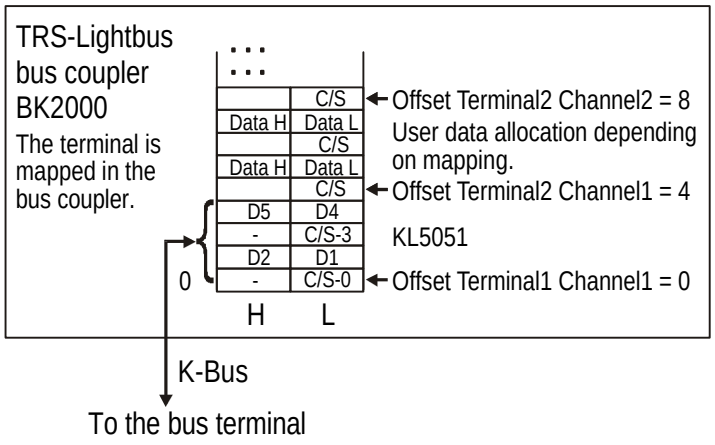
Actual position in E1, E2	Rotation angle
0x0000	0
0x3FFF	90 °
0xBFFF	270 °

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,...). For parametrization of a terminal, the control/status byte must als be mapped.

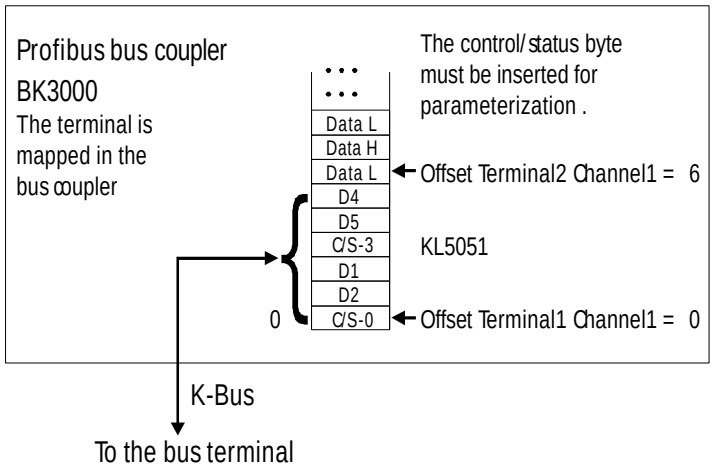
TRS Lightbus coupler BK2000

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.



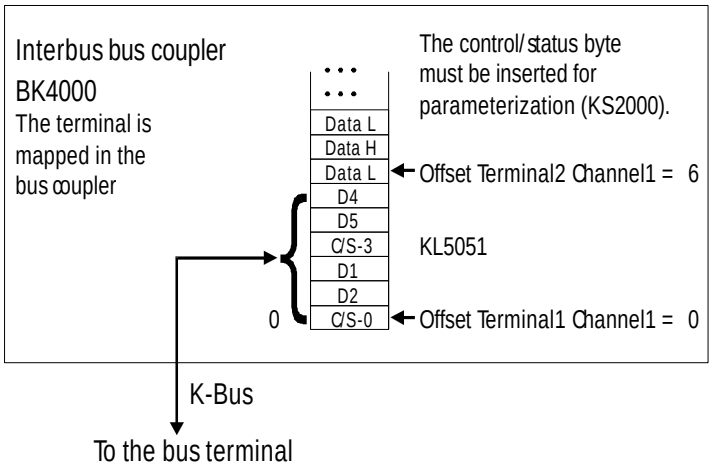
Profibus coupler BK3000

In the case of the Profibus coupler BK3000, the KL-5051 is always mapped with 6 bytes of input and 6 bytes of output data.



Interbus coupler BK4000

By default, the Interbus coupler BK4000 maps KL-5051 with 6 bytes of input 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 KL-5051

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

Type registers

R8 - R15 Registers in the terminal's internal ROM

The type and system parameters are programmed permanently by the manufacturer and can only be read by the user, but cannot be modified.

R8: Terminal type

The terminal type in register R8 is needed to identify the terminal.

R9: Software version X.y

The software version can be read as an ASCII character string.

R10: Data length

R10 contains the number of multiplexed shift registers and their length in bits.
The bus coupler sees this structure.

R11: Signal channels

In comparison with R10, the number of logically existing channels is located here. For example, one physically existing shift register may consist of several signal channels.

R12: Minimum data length

The respective byte contains the minimum data length of a channel to be transferred. The status byte is omitted if the MSB is set.

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
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

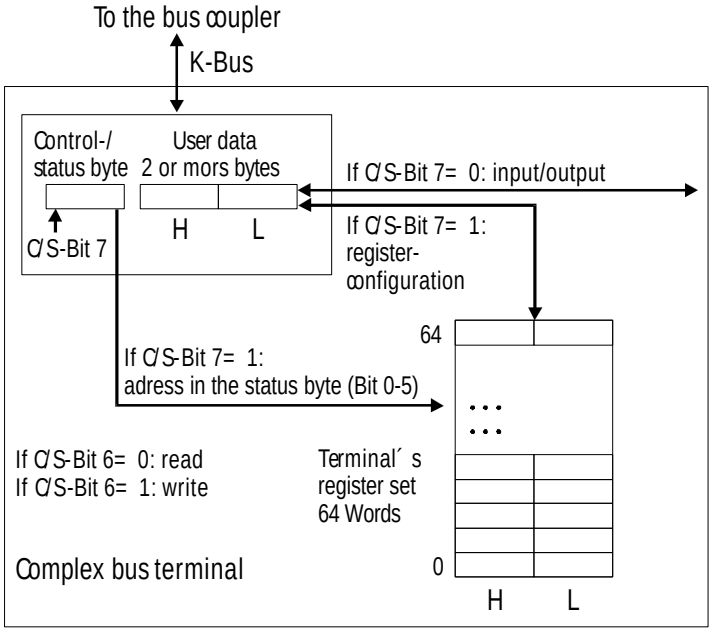
The address of the register to be addressed is entered in bits 0 to 5 of control byte.

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

Data exchange with the BI-SSI sensor

Communication with the servo amplifier takes place via the process data (A0-A5, E0-E5). A0/E0 is the control/status byte for operating data communication and A3/E3 is the control/status byte for parameter and servo status communication with the device.

Operating data status byte E0

The operating data status byte outputs possible error messages of the servo amplifier during process data exchange.

MSB

REG=0	ERROR	ALARM	KOM_ERR	CRC_ERR			

Bit	
ERROR	Is set when ALARM or KOM_ERR is set.
ALARM	The alarm bit of the servo amplifier is inserted here.
KOM_ERR	A communication disturbance has occurred. No valid data is being exchanged. Possible causes: the servo interface is not ready or is deenergised, a wire breakage has occurred or the connecting leads of the terminal have been swapped.
CRC_ERR	Invalid telegrams are occurring during data transfer (possibly EMC problem).

Parameter control byte A3 when setting the servo control (Bit 7 = 0)

Various actions are executed in the servo amplifier with this control byte.

MSB

REG=0		RD_PARH	RD_PARL	RS_ANS	RF	/NSTOP	/PSTOP

Bit	
REG	This bit switches bewtween the servo parameters and servo control/status communication.
RD_PARH	Read Parameter High Word (parameter address in A4)
RD_PARL	Read Parameter Low Word (parameter address in A4)
RS_ANS	Reset of response monitoring or of the following error. If the servo signals an error, response monitoring has occurred, for example, and the error can be reset by setting this bit. If the error message is not cancelled in the status byte E3 (SERV_ERR) (e.g. interior temperature too high), the servo amplifier must be deenergised (the other error messages can only be reset in this way).
RF	Controller enabling. The output stage is enabled and, at the same time, the brake (if available) is enabled.
/NSTOP (active low)	Negative setpoints are set to the setpoint zero.
/PSTOP (activw low)	Positve setpoints are set to the setpoint zero.

Parameter status byte E3 when detecting the servo status

The activated servo status word is read continuously by the servo and updated.

MSB

REG=0	KOM_ERR	RD_PARH_Q	RD_PARL_Q	SERV_ERR	RF_Q	/NSTOP_Q	/PSTOP_Q

Bit	
REG	0: the channel consisting of A3,A4,A5,E3,E4,E5 is in the operating mode with which the servo control/status is operated.
KOM_ERR	A data transfer error has occurred.
RD_PARH_Q	The high word of the requested parameter value is in E4,E5.
RD_PARL_Q	The low word of the requested parameter value is in E4,E5.
SERV_ERR	The servo amplifier's power section is signalling an error.
RF_Q	The output stage and, if available, the brake are enabled.
/NSTOP_Q (aktiv low)	1: negative sepoints are possible. 0: negative sepoints are set to zero.
/PSTOP_Q (aktiv low)	1: positive sepoints are possbile. 0: positive setpoints are set to zero.

Parameter control byte A3 in the mode parameter (Bit 7 = 1)

The parameter data is written into a buffer with the parameter address and is transferred to the servo on request. This request can take place with the last buffer entry. The terminal generates the data frame and checks and evaluates the checksum. The parameter consists of up to one double work, but shorter parameter data can also be sent.

MSB

REG=1		RD_PARH	RD_PARL		PUT_HW	PUT_LW	TRS_BUFF

Bit	
REG	This bit switches between servo parameter and servo control/status communication.
RD_PARH	Read Parameter High Word (parameter address in A4)
RD_PARL	Read Parameter Low Word (parameter address in A4)
PUT_HW	Write high word of the parameter into buffer (A4, A5 parameter high word).
PUT_LW	Write low word of the parameter into buffer (A4, A5 parameter high word).
TRS_BUFF	Write data from buffer to parameter address defined by A4.

Parameter status byte E3 in the mode parameter

During parameter communication (REG=1) with the servo, various acknowledgemnets are output in the status byte

MSB

REG=1	KOM_ERR	RD_PARH_Q	RD_PARL_Q	SERV_ERR	PUT_HW_Q	PUT_LW_Q	TRS_BUF_Q

Bit	
REG	1: the channel consisting of A3,A4,A5,E3,E4,E5 is in the mode with which parameter communication is taking place.
KOM_ERR	A data transfer error has occurred.
RD_PARH_Q	The high word of the requested parameter value is in E4,E5.
RD_PARL_Q	The low word of the requested parameter value is in E4,E5.
SERV_ERR	The power section of the servo is signalling an error.
PUT_HW_Q	The high word has been written into the buffer.
PUT_LW_Q	The low word has been written into the buffer.
TRS_BUF_Q	Data has been transferred successfully.

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 KL-5051,maps itself in the bus coupler depending on the set parameters.

Standard Format

The KL-5051 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.

Default: CAN CAL, CANopen

DevicNet

	I/O Offset	High Byte	Low Byte
Complete evaluation	3		
MOTOROLA format = 0	2	D5	D4
Word alignment = 0	1	CT/ST-3	D2
	0	D1	CT/ST-0

Default: Interbus, Profibus

	I/O Offset	High Byte	Low Byte
Complete evaluation	3		
MOTOROLA format = 1	2	D4	D5
Word alignment = 0	1	CT/ST-3	D1
	0	D2	CT/ST-0

Default: Lightbus

	I/O Offset	High Byte	Low Byte
Complete evaluation	3	D5	D4
MOTOROLA format = 0	2		CT/ST-3
Word alignment = 1	1	D2	D1
	0		CT/ST-0

	I/O Offset	High Byte	Low Byte
Complete evaluation	3	D4	D5
MOTOROLA format = 1	2		CT/ST-3
Word alignment = 1	1	D1	D2
	0		CT/ST-0

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-0(A0): Control- Byte (appears in the PI of the outputs).
ST-0(E0): Status- Byte (appears in the PI of the inputs).
CT-3(A3): Control- Byte (appears in the PI of the outputs).
ST-3(E3): Status- Byte (appears in the PI of the inputs).
D1, D2, D4, D5 = A1, E1, A2, E2, A4, E4, A5, E5

Table of the register set
of the KL-5051

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	5051	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	0x0000	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