DATA SHEET

HT RC130 00

HITAG^{TM} Co - Processor

Product Specification
Revision 0.1
Preliminary

September 2003

Frosch Electronics OEG

Table of Contents

Author: Reinhold Frosch

1 Introduction

1.1 Purpose of the HITAG^{TM} Co-Processor

The HITAG co-processor is designed to perform all computations in a HITAG 1 / HITAG 2 / HITAG S system concerning security, except for those executed on the transponder. The following items represent the fundamental framework of the security concept:

- data encryption
- mutual authentication
- password verification

1.2 Additional Features

Additional useful features are:

- on-chip EEPROM to store secret data
- uncomplicated host interface
- sleep mode for reduced current consumption

2 Specifications

2.1 Limiting Values

SYMBOL	PARAMETER	MIN.	MAX.
T_{stg}	storage temperature range	–55°C	+125°C
T_j	operating temperature	–40°C	+85°C
V_{DD}	supply voltage	–0.2 V	+3.6 V
V_{max}	Voltage at any I/O pin and V_{DD}	–0.5 V	+3.6 V
V_I	Voltage at any I/O pin to VSS	–0.5 V	V_{DD} + 0.3 V
I_{pk}	Peak output current P1x, P2x		15 mA
I_{LU}	Latch-up current	100 mA	

2.2 DC Characteristics

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	Unit
V_{DD}	operating voltages		2.1	3,0V	3,6 V	V
I_{DD}	operating supply current			190 µA	500 µA	µA
V_{IL}	input low voltage		-0,1		0.2 V_{DD}	V
V_{IH}	input high voltage		0.8 V_{DD}		V_{DD} + 0,1	V
I_{IL}	input low current	$V_{IL} = 0$			0.5	µA
I_{IH}	input high current	$V_{IH} = V_{DD}$			0.5	µA
V_{OL}	Output low voltage	I_O = 4 mA			0.4	V
V_{OH}	Output high voltage	I_O = -4 mA	V_{DD} - 0,4			V
I_{PU}	Pull-Up current	V_I = 0 V	30	75	150	µA

2.3 Mechanical Specifications

NAME	DESCRIPTION	OUTLINE VERSION
SSOP20	plastic shrink small outline package, 20 pin	SOT339-1

For further information see also:
http://www.semiconductors.philips.com/acrobat/packages/SOT339-1.pdf

2.4 Data Retention, Data Endurance

Data retention is guaranteed for 20 years, data endurance for 200k erase/write cycles.

2.5 Timing

Timings are specified in the sections **Hardware Interface** and **Interface Protocol**.

2.6 Complete Hardware Product Specification

For complete hardware product specification see also:
http://www.semiconductors.philips.com/acrobat/other/identification/pcf7941-pp.pdf

3 Description of the Co-processor

3.1 General Description

An automotive RISC Controller with 8 Bit Harvard Architecture in a SSOP20 plastic package from Philips is used as co-processor.

3.2 Pin Assignment and Function

PIN	FUNCTION	DESCRIPTION	NOTE
1	n.c.		1
2	n.c.		1
3	P14	General purpose I/O, LED, active low	3
4	V_{DD}	Supply Voltage	
5	P13	General purpose I/O with internal pull-up	3
6	P12	General purpose I/O with internal pull-up	3
7	P11	General purpose I/O with internal pull-up	3
8	P22	General purpose I/O	3
9	n.c.		2
10	MSCL	Not used	4
11	MSDA	Not used	4
12	n.c.		2
13	Reset	Reset	
14	P10	Ready\	3
15	Sleep	Sleep	
16	D_{out}	Data Out	
17	D_{in}	Data In	
18	S_{clk}	SClk	
19	V_{SS}	Common Ground	
20	V_{fld}	Not used	1

Note 1: Should be tied to Gnd
Note 2: Can be left open
Note 3: If not used, tie to Gnd
Note 4: Has to be left open

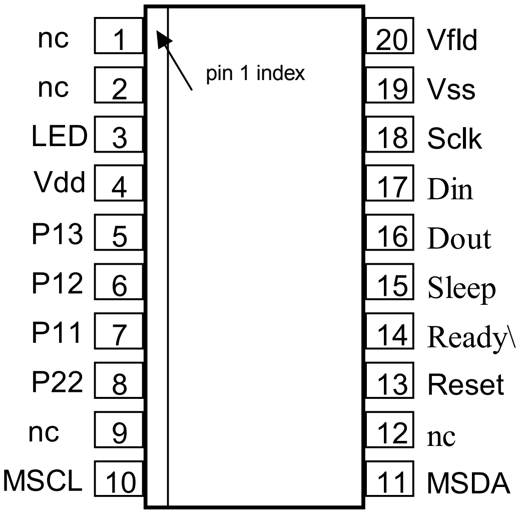

Figure 1: Pin Assignment

3.3 Hardware Interface

3.3.1 Host Interface Lines

The interface between host processor and co-processor uses a minimal number of control- and data lines in order to demand only a few port lines of the host processor.

The communication is done via a three/four line interface: D_{in} and D_{out} that can be driven by a bi-directional line on host side, a uni-directional clock line (S_{clk}) and a uni-directional acknowledge line (Ready\). The reset line (Reset) is also mandatory. It is used to reset the co-processor before startup and configuration (i. e. before each command). The data lines D_{in} and D_{out} are used to exchange data between the host processor and the co-processor. Data sent to the co-processor is called input, data sent from the co-processor is called output. The host processor must drive the clock line S_{CLK}. Since the processing time of the co-processor depends on the clock-frequency of the internal RC Oszillator and the internal state of the command, the co-processor pulls down the /READY-line to acknowledge that it is ready to accept new data.
For D_{out} a standard quasi-bi-directional I/O port is utilized:

Output LOW:	Push-pull driver forces the port to low
Output HIGH:	Push-pull state of the port is forced into tri-state

3.3.2 Reset

Prior to each command you must reset the co-processor.

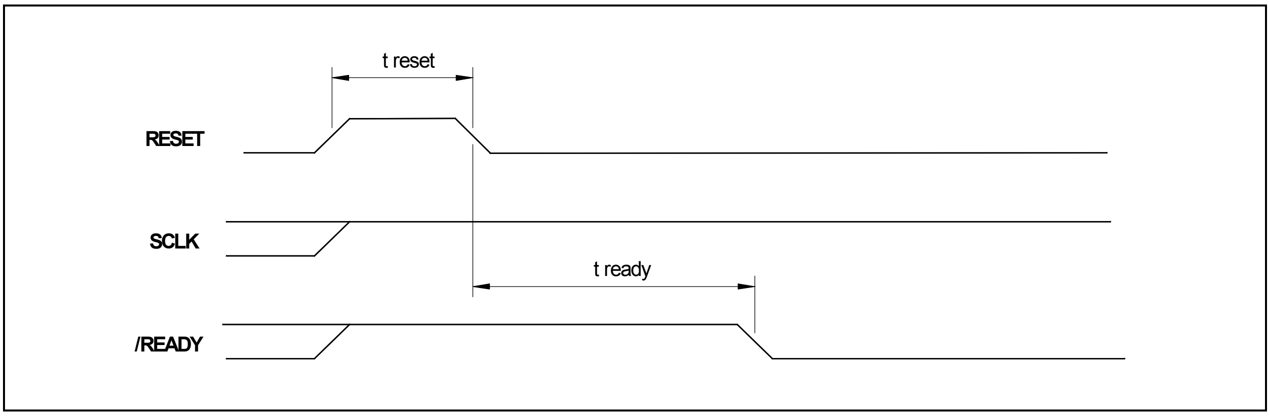

Figure 3: Resetting the co-processor

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t reset	minimum reset time	1.1			µs
t ready	reset to ready delay		310	600	µs

3.3.3 Data Transfer from Host to Co-Processor

The co-processor reads data at the rising edge of the clock SCLK. Immediately afterwards it pulls /READY to HIGH. If /READY is low again the co-processor is ready to accept new data.

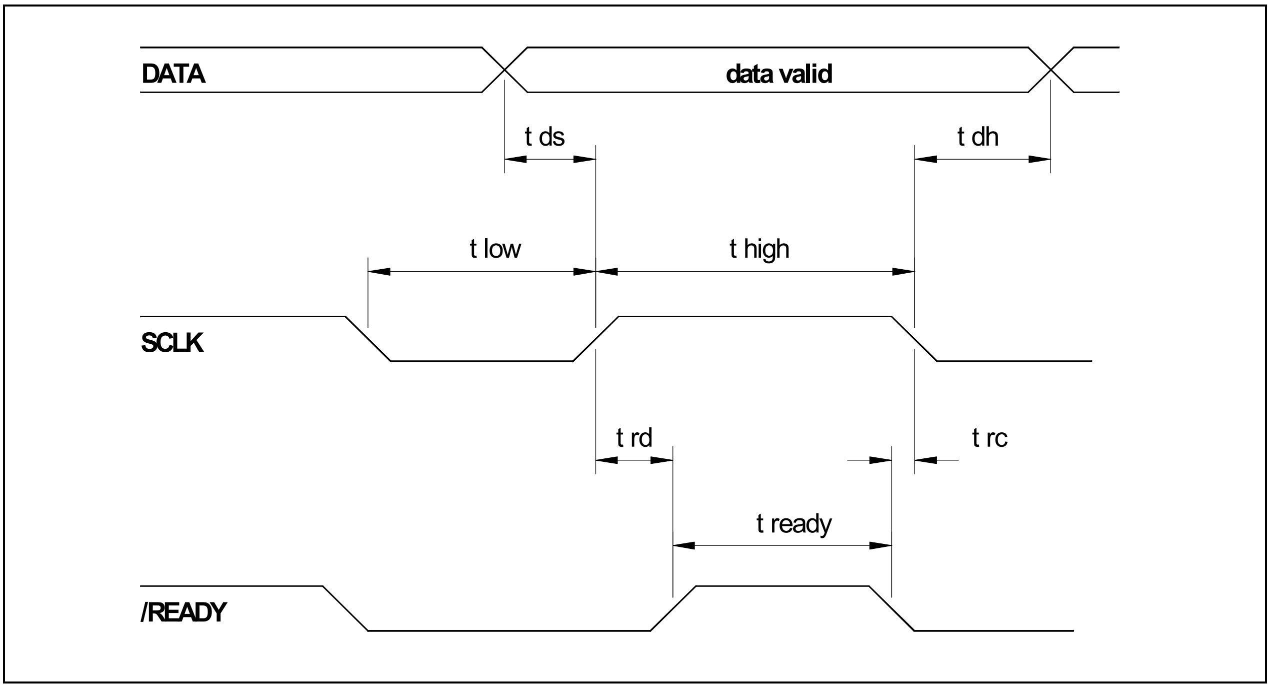

Figure 2: Data transfer to co-processor

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t low	SCLK low time	2,2			µs
t high	SCLK high time	2,2			µs
t ds	data setup time	0			µs
t dh	data hold time	0			µs
t rd	ready delay time		2,5		µs
t rc	SCLK to ready delay time	0			µs

3.3.4 Data Transfer from Co-Processor to Host

When the co-processor sends data to the host processor the host processor supplies the clock SLCK. Data changes after the rising edge of SLCK and /READY becomes HIGH. Data is valid when /READY becomes LOW.

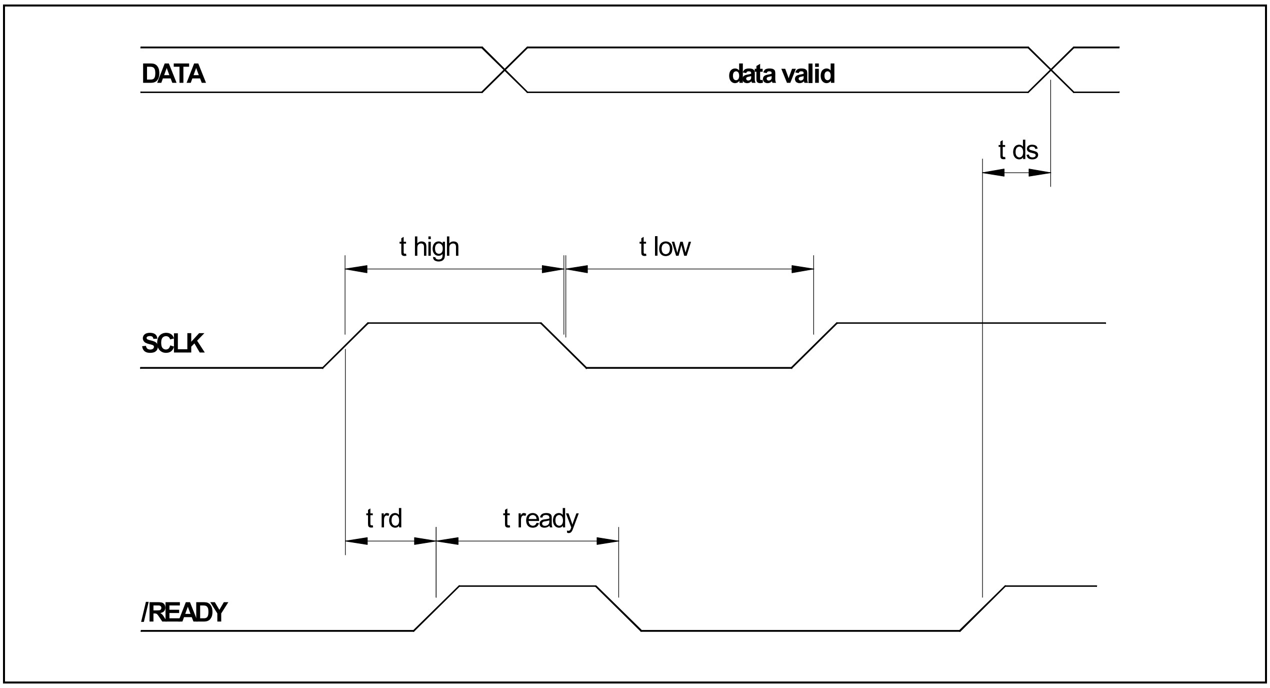

Figure 4: Data transfer from co-processor

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t low	SCLK low time	2,2			µs
t high	SCLK high time	2,2			µs
t ds	data setup time		33		µs
t rd	ready delay time		2,5		µs

3.3.5 Switching between Read Mode and Write Mode

During data exchange between host processor and co-processor the direction of data flow changes.
In read mode the co-processor reads data (input) and the host processor sends data.
In write mode the co-processor writes data (output) and the host processor receives data.
The co-processor has internal pull-up-resistors. In write mode the host processor is not allowed to pull the data line to high or low.
Thus the output driver of the host processor must be an open collector or open drain driver.
Internal or external pull-up-resistors are allowed.

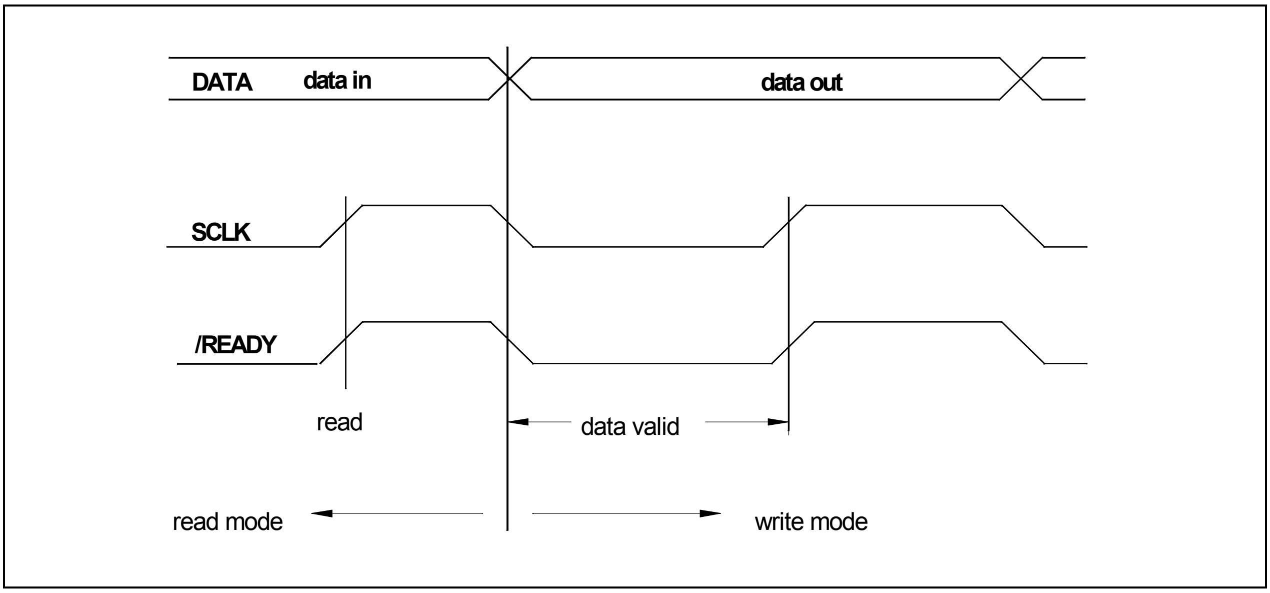

Figure 5: Switching between read mode and write mode

3.3.6 Sleep Mode

The co-processor provides a sleep mode. In sleep mode the current consumption is reduced to approximately 20 µA. To put the co-processor in sleep mode SLEEP must be set to HIGH. /READY becomes HIGH, when the co-processor has entered sleep mode.

When SLEEP is LOW, a reset (4.4) restarts the co-processor.
As soon as the co-processor is able to accept data it pulls /READY to LOW.

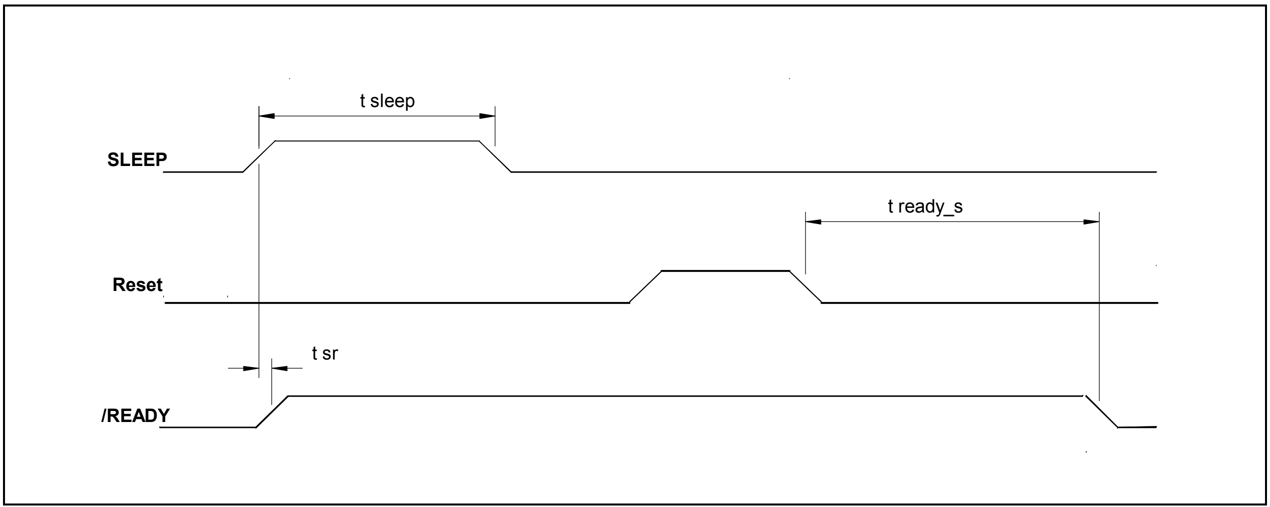

Figure 6: Entering and leaving sleep mode

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t sleep	sleep high time	6			µs
t sr	sleep to ready delay time		6		µs
t ready_s	sclk to ready delay			9	ms

4 Interface Protocol

4.1 Command Set

Each command is preceded by a reset. If the RESET pin is set to LOW the co-processor starts its program. Data must not be sent before /READY becomes LOW (figure 6).
The protocol between host processor and co-processor starts with a reset followed by 8 data-bits. These 8 bits are command bits which determine what the co-processor does.

Valid commands are:

	first out ↓							
start personalization	1	0	1	1	1	1	1	1
start personalization HitagS	1	0	1	1	1	1	1	0
get Version	1	1	1	1	1	1	1	0
start HITAG 1 crypto mode using Key A	1	1	0	1	0	1	1	1
start HITAG 1 crypto mode using Key B	1	1	1	1	0	1	1	1
start HITAG 2 crypto mode with password check	0	1	1	1	0	1	1	1
start HITAG 2 crypto mode without password check	0	1	1	0	0	1	1	1
start HitagS crypto mode with password check	0	1	0	0	0	0	0	0
start HitagS crypto mode without password check	0	1	1	0	0	0	0	0
Read EEPROM Byte	0	0	0	0	1	1	1	1
Write EEPROM Byte	0	0	0	0	1	1	0	1
Set Direction of User IO	0	0	0	0	1	0	0	0
Set User IO Output	1	0	0	0	1	0	0	1
Read User IO Input	1	0	0	0	1	0	1	0
LED On (active low)	1	0	0	0	1	1	0	0
LED Off (active low)	1	0	0	0	1	1	0	1
User Port On	1	0	0	0	1	1	1	0
User Port Off	1	0	0	0	1	1	1	1

All other combinations are not valid.

4.2 Personalization

In order to compute the data for encryption the co-processor requires secret data (i.e. keys, passwords ...). During personalization these secret data are stored in an on-chip EEPROM on the co-processor itself. These data can not be read. After personalization you can always use the internal data, which are stored on chip, as long as key, logdata or passwords do not change. Data retention is guaranteed for 10 years.
If the personalization is started it is not allowed to stop the communication before the last programming time ($t_{progend}$). Otherwise the co-processor might be damaged. In case of wrong data sent to the coprocessor no data are written to the EEPROM

Content of the EEPROM memory as delivered: FF hex

first out
↓

command to start personalization:

1	0	1	1	1	1	1	1

Data sent by the host processor are stored in blocks of 32 bits in the co-processors EEPROM. After the co-processor has received the 32nd bit, /READY goes to. Then /READY becomes LOW again and you can send the next 32 bits.

Key 16 is a 16 bit quantity, PW 24 is a 24 bit quantity, but the co-processor always expects 32 bits. Key 16 must be preceded by 16 dummy bits and PW 24 must be preceded by 8 dummy bits to form a total of 32 bits.

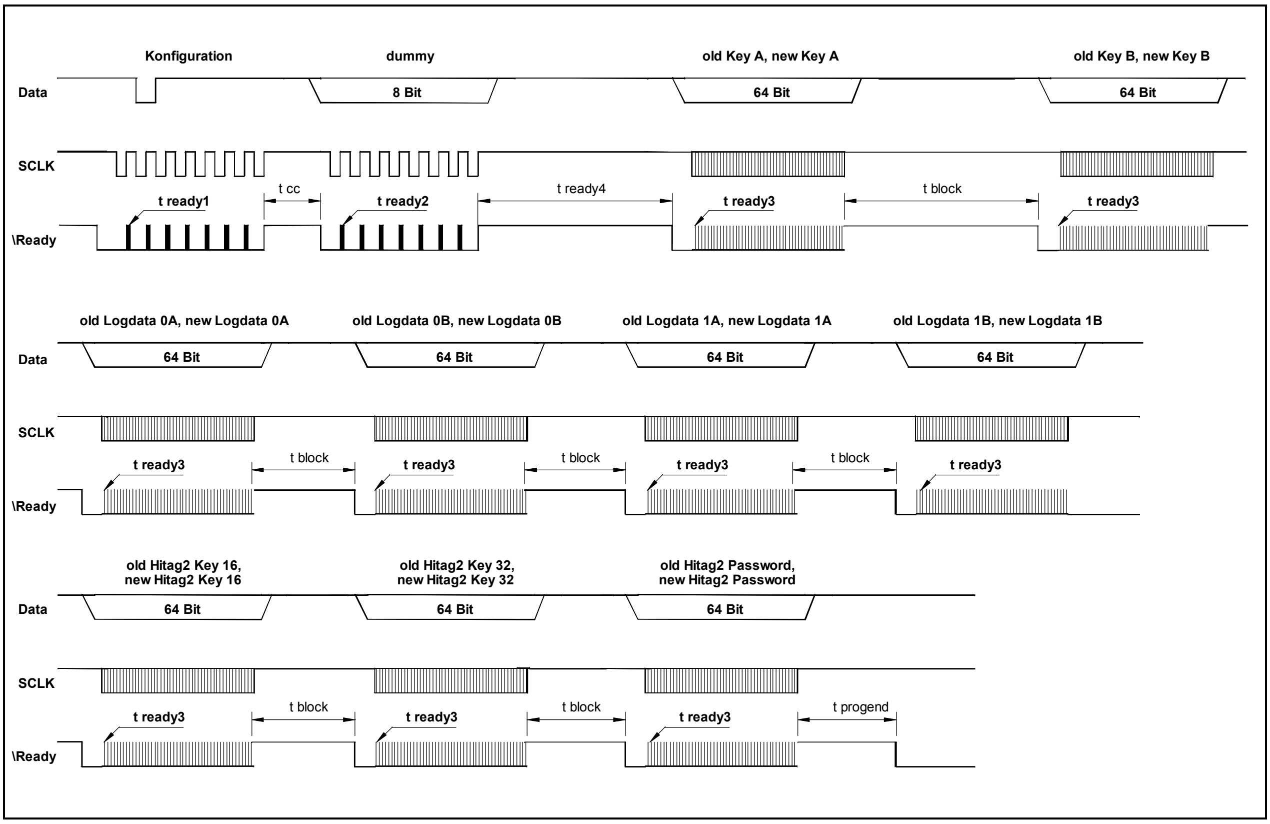

Figure 7: Personalization of the co-processor

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t cc	config ready delay		12		µs
t block	Calculation time between tw blocks		150		µs
t ready1	bit storage time for config byte		54		µs
t ready2	bit storage time for CntVal - Byte		54		µs
t ready3	bit storage time for data bit	54		63	µs
t ready4	storage time for CntVal		5		µs
T progend	last programming time		30		ms

	explanation	bits
Command	The first 8 bits determine the co-processor command. If bit 2 is zero the co-processor is set to personalization mode.	8
CntVal	see below	8
Key A	Key A for HITAG 1 transponders	32
Key B	Key B for HITAG 1 transponders	32
Logd. 0A	Logdata 0A for HITAG 1 transponders	32
Logd. 0B	Logdata 0B for HITAG 1 transponders	32
Logd. 1A	Logdata 1A for HITAG 1 transponders	32
Logd. 1B	Logdata 1B for HITAG 1 transponders	32
Key 16	16 Bit Key for HITAG 2 transponders (KEY HIGH)	32
Key 32	32 Bit Key for HITAG 2 transponders (KEY LOW)	32
PW 24	24 Bit Password/Tag for HITAG 2 transponders	32

Table 1: Abbreviations used for figure 5

4.3 Get Version

The host processor sends 8 bits (the command) to the co-processor followed by a 32 bit serial number. The actual value of this 32 bit number is of no significance. Then the host processor reads two information bytes and a 31 byte program version string.

first out ↓

Command to read Softwareversion of Coprocessor:

1	1	1	1	1	1	1	0

The first byte of information is the value CntVal which determines the length of the EEPROM programming interval. The second information byte contains the EEPROM address where the pseudo random number is presently stored. The program version string contains the version number and the date and time when the program version was released.

An example of a valid version string is:

„V 1.50 Thu Nov 21 12:41:25 1996“

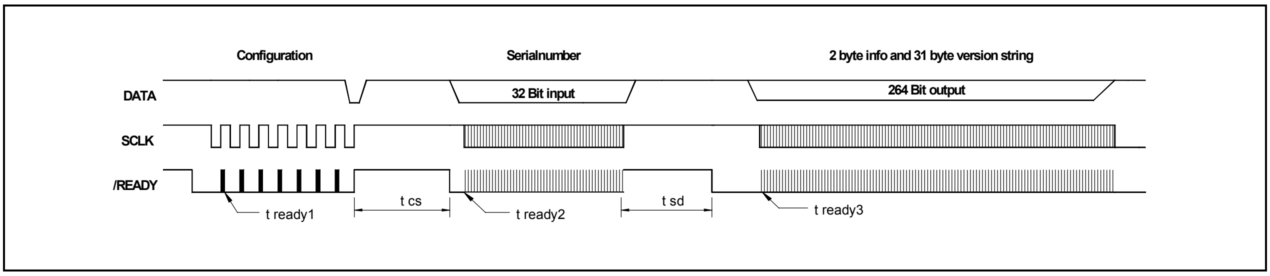

Figure 8: Protocol for Get Version

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t cs	config ready delay		95		µs
t sd	data output ready delay		206		µs
t ready1	bit storage time for config byte		54		µs
t ready2	bit storage time for CntVal - Byte		54		µs
t ready3	bit storage time for data bit	63		118	µs

4.4 Cipher Phase

In cipher mode the co-processor generates encryption stream data. The host processor reads the encryption stream and encodes or decodes the data received from or sent to the transponder by exoring data bits and encryption stream data. The co-processor switches to cipher phase after having successfully executed one of the following commands:

- start HITAG 1 crypto mode using Key A
- start HITAG 1 crypto mode using Key B
- start HITAG 2 crypto mode with password check
- start HITAG 2 crypto mode without password check

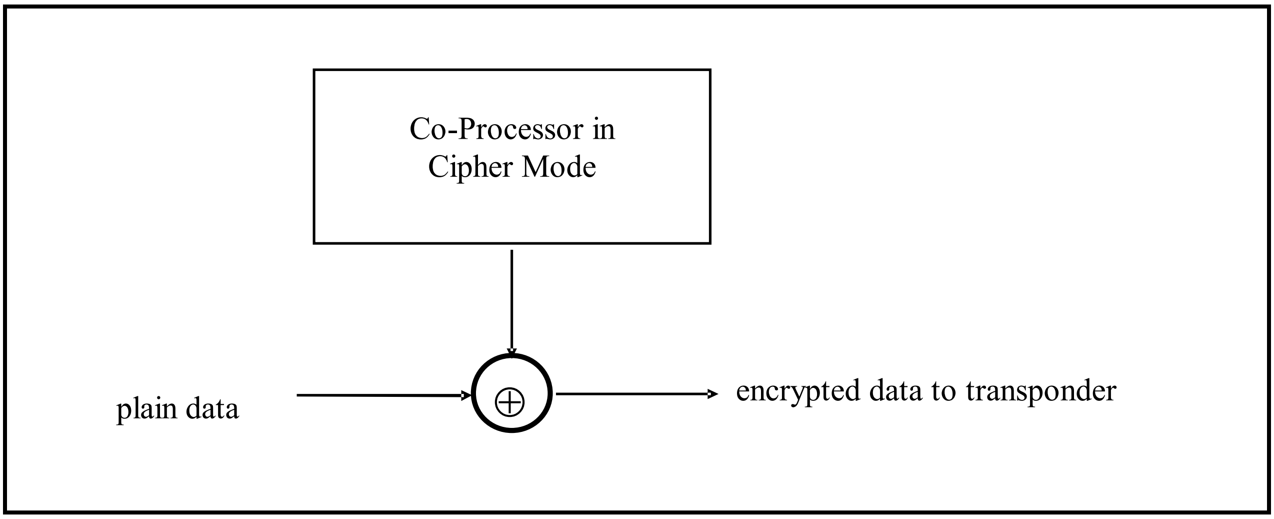

Figure 9: Cipher Mode: Data-flow for encryption

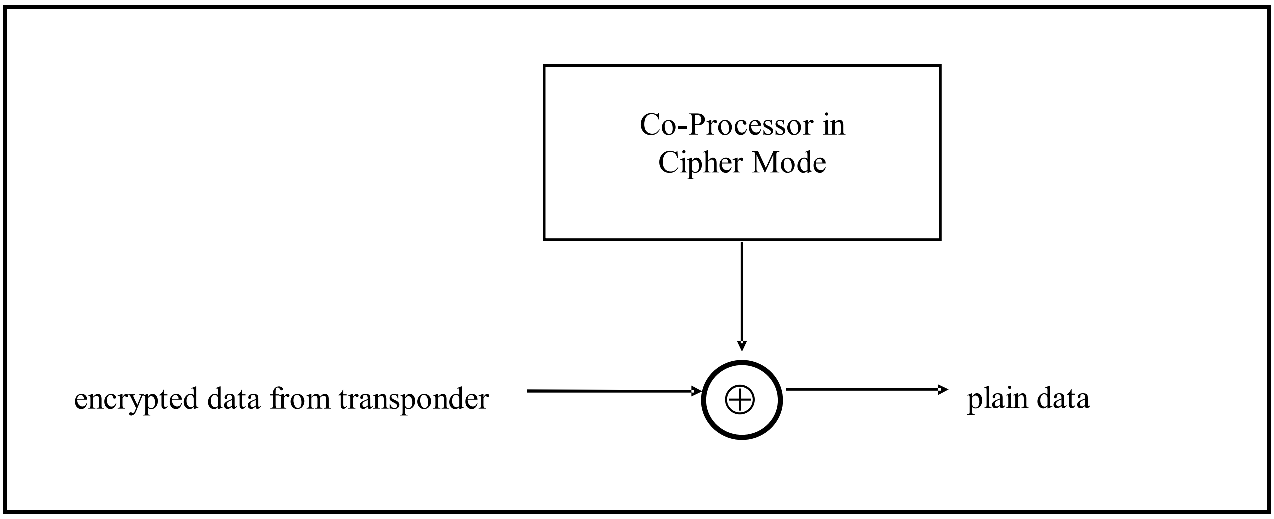

Figure 10: Cipher Mode: Data-flow for decryption

4.5 HITAG 1 Crypto Mode

The host processor sends 8 bits (the command) to the co-processor in order to start the HITAG 1 crypto protocol. The 3rd bit determines whether Key A, Logdata 0A, Logdata 1A or Key B, Logdata 0B, Logdata 1B shall be used.

first out
↓

Command to start HITAG 1 crypto mode using Key A:	1	1	0	1	0	1	1	1
Command to start HITAG 1 crypto mode using Key B:	1	1	1	1	0	1	1	1

All computations required to perform the authentication on the read/write device are performed. First, the host processor reads the serial number of the transponder and sends it to the co-processor. Then the host processor reads a random number generated by the co-processor and sends it to the transponder.
The host processor reads the encrypted logdata 0A (0B) from the transponder and sends it to the co-processor. The co-processor decipheres and compares logdata 0A (0B) with the data stored in its internal EEPROM. If the comparison fails the co-processor does not respond and /READY remains HIGH. In this case the co-processor needs a reset.

If the comparison succeeds the co-processor encrypts logdata 1A (1B) and pulls /READY to LOW. The host processor reads the encrypted logdata 1A (1B) and sends it to the transponder. With this step the co-processor has finished initialization and authentication and switches to cipher phase.

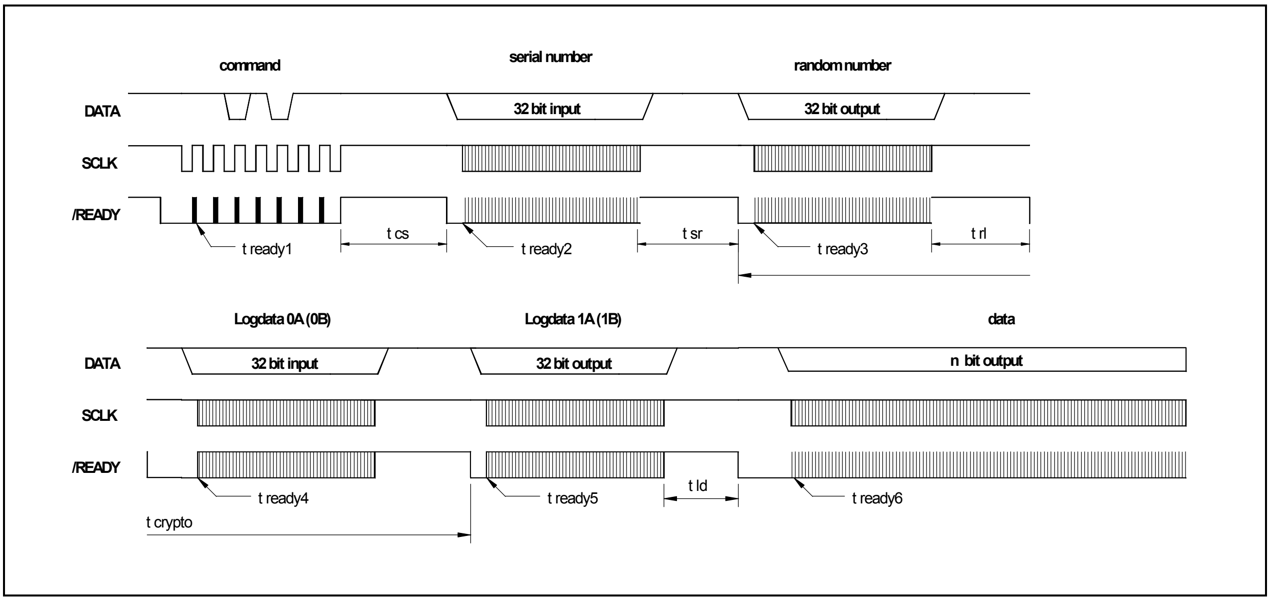

Figure 11: Protocol for crypto mode for HITAG 1

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t cs	config ready delay		12		µs
t sr	serial number ready delay		860		µs
t rl	random ready delay		5		µs
t crypto	logdata 0 compare ready		10		ms
t ld	logdata 1 ready delay		600		µs
t ready1	bit storage time for config byte		6		µs
t ready2	bit storage time for serial number		6		µs
t ready3	bit output time for random number		6		µs
t ready4	bit storage time for logdata 0		6		µs
t ready5	bit output time for logdata 1		6		µs
t ready6	data bit output time		133		µs

4.6 HITAG 2 Crypto Mode with Password Check

first out
↓

Command to start HITAG 2 crypto mode with password check:

0	1	1	1	0	1	1	1

All steps required to perform the authentication on the read/write device are performed. First, the host processor reads the serial number of the transponder and sends it to the co-processor. Then the host processor reads a 32 bit random number and a 32 bit secret data stream generated by the coprocessor and sends them to the transponder. Afterwards the host processor reads the encrypted configuration page (including configbyte and password) from the transponder and sends it to the co-processor. The co-processor decipheres and compares the received password with the data stored in its internal EEPROM. If the comparison fails the co-processor does not respond and /READY remains HIGH. In this case the co-processor needs a reset.
If the comparison succeeds the co-processor pulls /READY to low and the host processor reads the decrypted 8 configuration-bits. With this step the co-processor has finished initialization and authentication and switches to cipher phase.

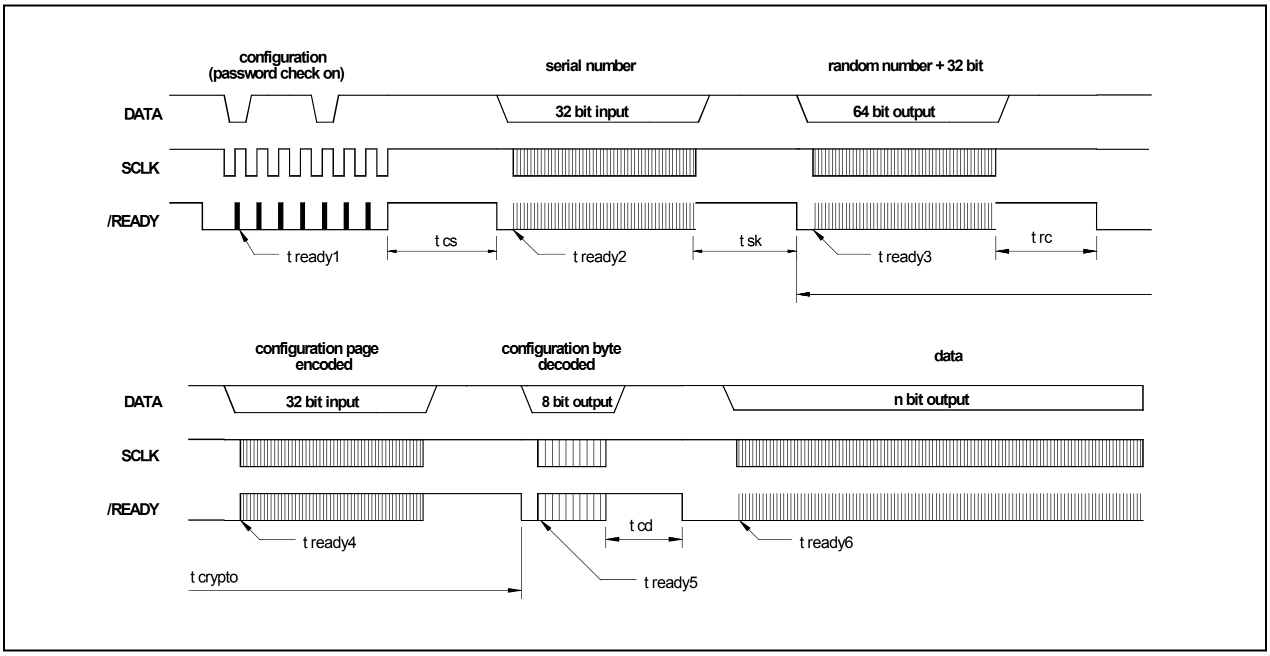

Figure 12: Protocol for crypto mode for HITAG 2 with password check

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t cs	config ready delay		95		µs
t sk	serial number ready delay	18			ms
t rc	random number output delay		78		µs
t crypto	decryption time for configuration page	17			ms
t cd	configuration page output delay		188		µs
t ready1	bit storage time for config byte		54		µs
t ready2	bit storage time for serial number	54		63	µs
t ready3	bit output time for random number and data	63		73	µs
t ready4	bit storage time for configuration page	54		63	µs
t ready5	bit output time for configuration byte	63		73	µs
t ready6	databit output time		285		µs

4.7 HITAG 2 Crypto Mode without Password Check

first out
↓

Command to start HITAG 2 crypto mode without password check:

0	1	1	0	0	1	1	1

First, the host processor reads the serial number of the transponder and sends it to the co-processor. Then the host processor reads a 32 bit random number and a 32 bit secret data stream generated by the co-processor and sends them to the transponder. Afterwards the host processor reads the encrypted configuration page (including configbyte and password) from the transponder and sends it to the co-processor. The co-processor decodes it and the host processor reads it. With this step the co-processor has finished initialization and switches to cipher phase.

Attention: This command does not check the password sent by the tag.

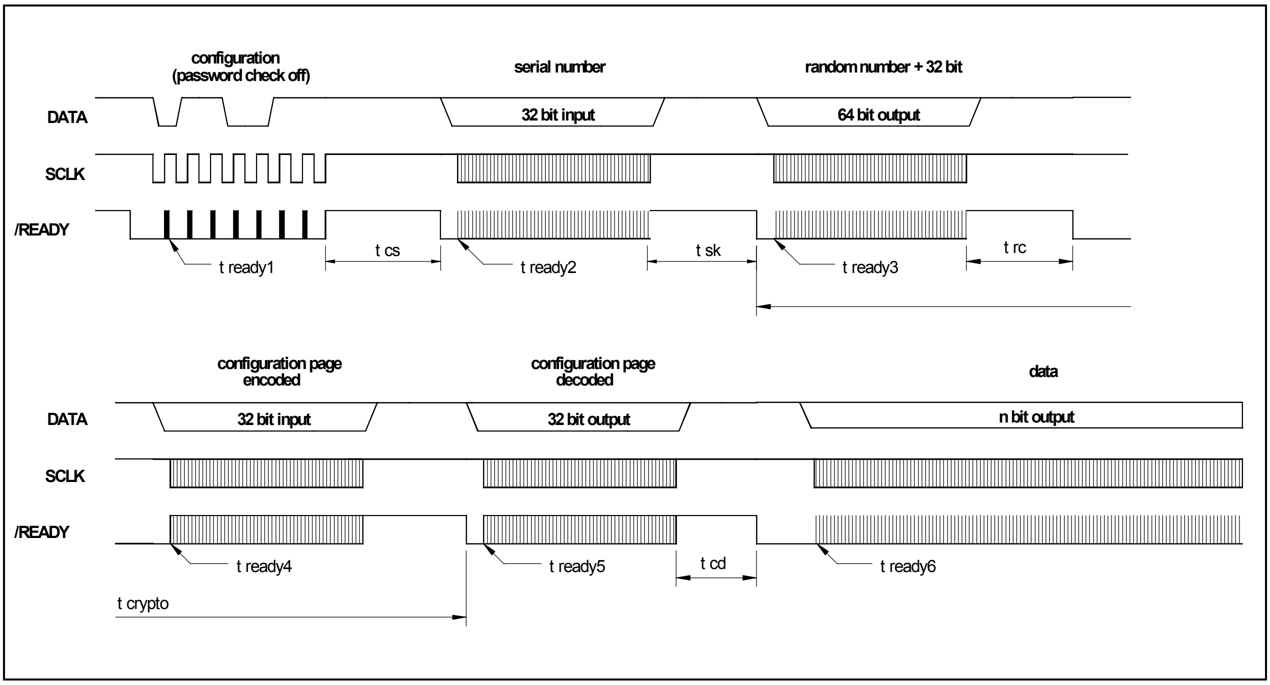

Figure 13: Protocol for crypto mode for HITAG 2 without password check

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
t cs	config ready delay		12		µs
t sk	serial number ready delay		400		µs
t rc	random number output ready delay		10		µs
t crypto	decryption time of configuration page	17			ms
t cd	configuration page output ready delay		52		µs
t ready1	bit storage time for config byte		6		µs
t ready2	bit storage time for serial number		6		µs
t ready3	bit output time for random number and data		6		µs
t ready4	bit storage time for configuration page		6		µs
t ready5	configuration page - bit output time		6		µs
t ready6	data bit output time		66		µs

5 Ordering information

Type Number	Description
Nyd	
Nyd	

Definitions

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics section of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.	
Application information	
Where application information is given, it is advisory and does not form part of the specification.	

Life support applications

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury.
Frosch Electronics customers using or selling these products for use in such applications do so on their own risk and agree to fully indemnify Frosch Electronics for any damages resulting from such improper use or sale.