ib technology

Data Sheet

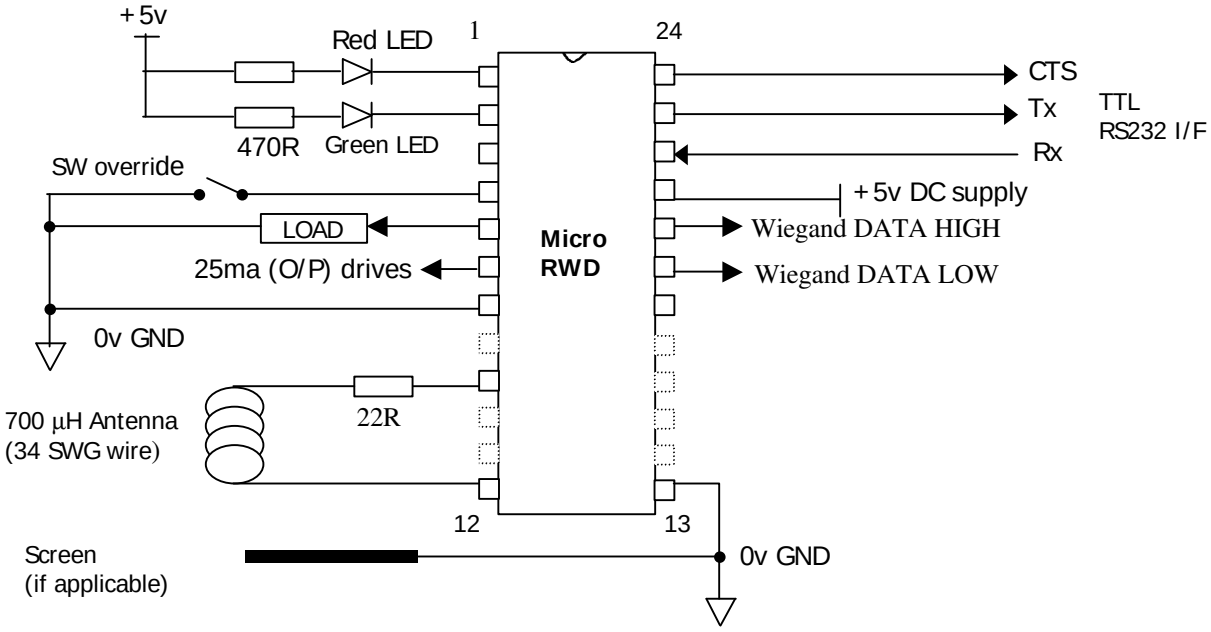
H1W200102.pdf

10 Pages
Last Revised 30/01/02

Micro RWD H1 "Wiegand" Output Version

This version of the Micro RWD product behaves in the same manner as the standard Micro RWD Hitag 1 reader with the additional feature of having variable length "Wiegand" data outputs. Also, the OP2 and OP3 outputs are now active high and idle in a low state. The solution only needs a 600-700µH antenna coil connected and 5v DC supply to be a fully featured read/write solution. The Wiegand DATA high and DATA low signals form the common protocol that has been a standard for many years on door entry and access control readers. The usual TTL serial interface is also supported along with the simple commands to read and write tag data and programme the RWD's internal parameters.
The diagram below shows the pinout configuration for the Hybrid chip version; two pins are used to output the DATA high and DATA low signals according to the "Wiegand" protocol. The Micro RWD "Wiegand" version can replace existing door entry systems with a fully compatible contactless smart card solution.

Micro RWD Hybrid chip connections



The Micro RWD "Wiegand" version 1.20 supports 4 to 34 bit output protocols (2 to 32 bits of data). The particular data length required is programmed into the RWD internal EEPROM byte 1 (second location). The value loaded is rounded down to an even number and a zero parameter effectively turns the Wiegand output OFF. Values less than 4 are treated as zero and values greater than 34 are set to the maximum value of 34.

The RWD operates by using the 32 bit (4 byte) unique serial number from Page 0 of the Hitag 1 transponder memory to create the required DATA high, DATA low Wiegand outputs.
The 4 to 34 bit variable Wiegand output (2 to 32 bits data field) is taken as the first N bits (most significant bit first) of the serial number.

The RWD also operates as a standard Hitag 1 reader and the TTL level serial interface can be used to read and write data for any of the Hitag 1 pages. Note that the READ and WRITE PAGE commands are fully implemented as with the standard MicroRWD H1 version but the READ and WRITE BLOCK commands have been removed for reasons of code optimisation. In this manner the MicroRWD H1 "Wiegand" version is a highly flexible solution that will report the H1 serial number as Wiegand outputs whenever a valid transponder enters the RF field and can also be used to read/write any other memory area of the H1 transponder. OP3 is also set high for two seconds after the Wiegand output to operate a buzzer and to allow for separation of repeated data frames.

Wiegand Output Protocol

The Wiegand protocol (26 bit mode) itself is made up of a leading even parity bit (for b0 - b11), 24 bits of data (from transponder data) and a trailing odd parity bit (for b12- b23). The 30 bit mode has the same format except 28 bits are used to form the data sequence.

For example:-

H1 serial number (Page 0) data (Hex): 04 60 22 12

Wiegand 26 bit sequence:- E (b0 --------- b11) (b12 -------- b23) O

E (0 4 6 0 2 2) O

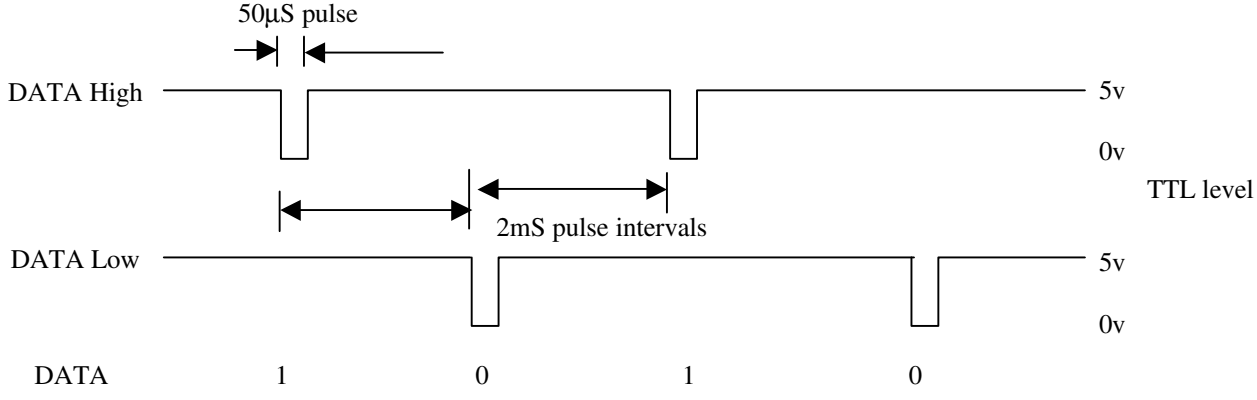
1 0000 0100 0110 0000 0010 0010 1

Where E is EVEN parity bit for bit 0 to 11 and O is ODD parity bit for bits 12 to 23

The serial number of Hitag 1 transponders is unique and is set during manufacture, so the Wiegand data will be different for each tag.

The complete Wiegand bit sequence is output whenever the tag is within the RWD's antenna field and the tag has been validated. This output is independent of the normal TTL serial interface which responds to received commands and replies with the data as requested. The physical Wiegand protocol is asynchronously transmitted as low going 50 µS pulses on the appropriate DATA low or DATA high pins. These pulses are separated by 2mS periods. The Wiegand sequence is output a single time whenever a valid tag enters the RF field for the first time. The data frame is then followed by a two second delay (OP3 output pulsed high) before normal operation continues (NO Wiegand output if selected length is ZERO).

Wiegand Protocol Timing Diagram



MicroRWD Hitag 1 operation

The MicroRWD H1 (Wiegand) version is also a complete read and write tag acceptance solution for Hitag 1 RF transponders. The module provides internal EEPROM memory for holding lists of authorised identity codes, a manual override switch facility and has LED drives to give visual indication of acceptance as well as having the Wiegand protocol outputs.

The RWD also has a TTL level RS232 interface that allows a host system to communicate with the RWD if necessary, so that system features can be customised, configurations changed and tag read/write data handled by the host system.

The Hitag 1 transponders have significantly more memory than most other tags and provide 2048 bits (256 bytes) of EEPROM memory arranged as partitioned 32 bit pages. An area of 1536 bits (192 bytes) is open for general user data. The communication protocol (handled entirely by the RWD) supports multiple tags in the RF field and data integrity and security is ensured using extensive CRC (Cyclic Redundancy Check) methods and highly encrypted data storage.

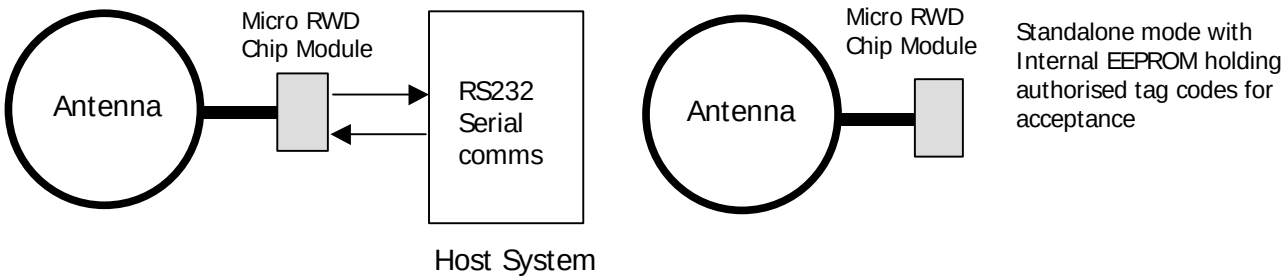
The MicroRWD is essentially a proximity system and a Read/Write range of up to 20cm can be achieved with the same level of reliable communication and EMC resilience. The unique AST (Adaptive Sampling) feature allows the RWD to continually adjust and re-tune the sampling to allow for inductive changes in the RF field, an essential feature for real-world reliability and robust operation. The communication protocol with the tags can achieve 4k bits/second of data transfer and the total time to read a 32 bit page, including reading of the serial number, selecting the tag and the read operation takes less than 100ms.

The MicroRWD can be easily integrated into almost any application; when power (5v DC) is first applied to the board the red and green LEDs flash once to indicate successful power-up. The device can also check for broken or shorted antenna and can even detect very badly tuned antennas, these problems are indicated by the red LED flashing continuously until the fault has been rectified.

The MicroRWD will normally have the red LED lit until a valid card or tag is brought into the RF field. If the tag is accepted as valid then the green LED is lit and the output driver (OP2) is switched on. This provides up to 25ma of drive current for operating a relay etc. In addition, a switch input is provided for overriding the tag reading operation and switching the output drive directly. When the green LED is lit the tag serial number is also output as Wiegand protocol on the DATA high and DATA low outputs (OP0, OP1) and the OP3 line is pulsed high for a two second period (to operate a buzzer and allow separation of Wiegand data frames). This Wiegand data is only output a single time while the valid card is in the RF field. If the Wiegand data length is ZERO (as programmed into the RWD internal memory) then there is no Wiegand output and OP3 remains low. In this case the RWD serial commands still operate in the usual manner.

(Hitag 1 is a trademark of Philips Semiconductors)

The Micro RWD has two basic modes of operation:-



Remote mode (connected to a host computer or microcontroller) and Standalone mode.

1) Remote mode involves connecting to a host serial interface. This is where the stored list of authorised identity codes can be empty, effectively authorising any HT1 transponder for subsequent read/write operations. A simple serial protocol allows a host system to communicate with the Micro RWD in order to program new authorised identity codes, change encryption seed code and perform read/write operations to the tag itself.
2) Standalone mode is where the HT1 tag identity codes are checked against a stored list of authorised codes. If an identity code is matched, the output drive and Green LED are enabled. Effectively standalone mode occurs when there is no host system communicating with the Micro RWD.

Supported transponder types

The Micro RWD is designed to communicate with Hitag1 transponders configured in R/W Public mode. Setting the HT1 to any other configuration will render them inoperable with this system. **Note: Only the HT1 ICS30 02x Hitag silicon is fully supported for WRITE/ READ operations. The earlier HT1 ICS30 01x silicon (made obsolete early 1997) is only supported for READ operations.**

The operation of the Micro RWD and Hitag 1 transponders is described in more detail at the end of this document.

The identification codes described in this text are regarded as the first four bytes (serial number or page 0) of the tag memory array.

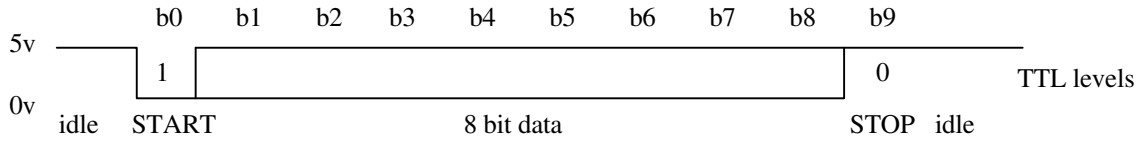
Serial Interface

This is a basic implementation of RS232. The Micro RWD does not support buffered interrupt driven input so it must control a BUSY (CTS) line to inhibit communications from the host when it is fully occupied with tag communication. It is assumed that the host (such as a PC) can buffer received data.
Tx, Rx and RTS signals from the Micro RWD are all TTL level and can be converted to +/- 10v RS232 levels using an inverting level converter device such as the MAX202 (note the inversion of the TTL levels).

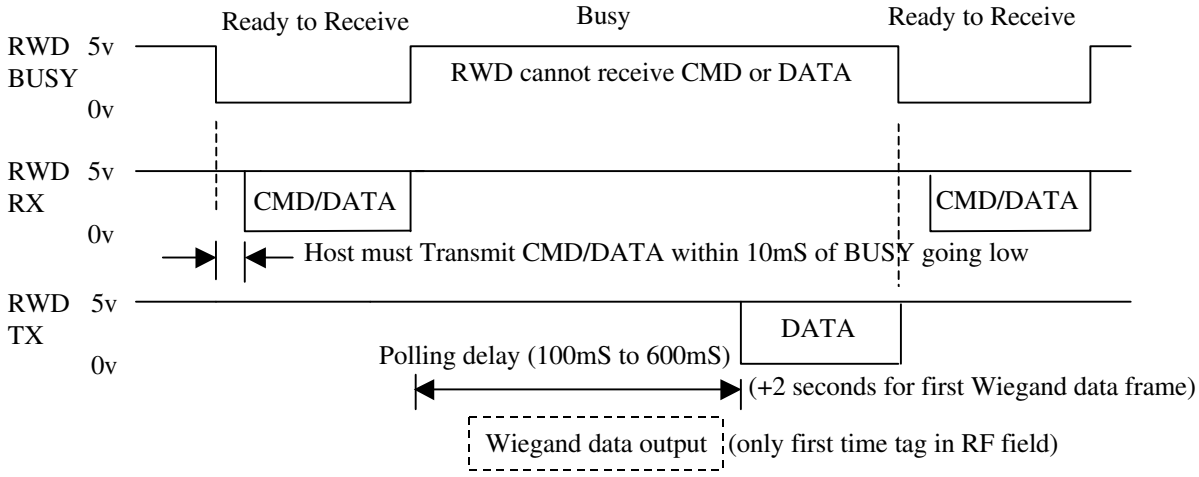
The serial communication system and protocol allows for a 10ms 'window' every Tag polling cycle indicated by the BUSY line being low. During this 'window' the host must assert the first start bit and start transmitting data. The BUSY goes high again 10ms after the last stop bit is received. NOTE that only one command sequence is handled at a time.

NOTE also that the acknowledge or data reply from the MicroRWD to the host can take two seconds when a tag is brought into the RF field for the first time. This is to allow sufficient separation of the Wiegand data frames.

Transmitted or Received data byte, 9600 baud, 8 bit, 1 stop, No parity (104µS per bit)



RWD tag polling cycle and serial communication BUSY protocol



Command Protocol

The following commands are supported. The corresponding acknowledge code should be read back by the host and decoded to confirm that the command was received and actioned correctly. The serial bit protocol is 9600 baud, 8 bits, 1 stop, no parity (lsb transmitted first). The status flags returned in the Acknowledge byte are as follows:

```
b7  b6  b5  b4  b3 b2 b1 b0
1   1   1   1   1  1  1  1
          |   |   |  |  |   EEPROM error (Internal EEPROM write error)
          |   |   |  |  Tag OK (Tag ident code matched to list)
          |   |   |  Rx OK  (Tag  comms and acknowledgement OK)
          |   |   RS232  error (Host serial comms error)
          |   RELAY Enabled flag
          HTRC (or Antenna fault) error flag
```

Note that bits 6 and 7 are fixed 1's so that an acknowledge code of D6 (Hex) would generally indicate no errors with a matched (and authenticated) Tag present.
Note also that only the relevant flags are set after each command as indicated in the following specification.

Write Tag Page

Command to write 4 bytes of data to HT1 32 bit page. If the write was unsuccessful (invalid tag or out of field) then Status flags in acknowledge byte indicate error.

	B7							B0	
Command:	0	1	0	1	0	1	1	1	(0x57)
Argument1:	x	x	N	N	N	N	N	N	(N = HT1 page address 0-63)
Argument2:	D	D	D	D	D	D	D	D	(D = msb data to write to HT1)
Argument3:	D	D	D	D	D	D	D	D	
Argument4:	D	D	D	D	D	D	D	D	
Argument5:	D	D	D	D	D	D	D	D	(D = lsb data to write to HT1)
Acknowledge:	1	1	F	F	F	F	F	X	(F = Status flags)

Read Tag Page

Command to read 4 bytes of data from HT1 32 bit page. If the read was successful, indicated by acknowledge status flags then four bytes of tag data follow.

	B7							B0	
Command:	0	1	0	1	0	0	1	0	(0x52)
Argument1:	x	x	N	N	N	N	N	N	(N = HT1 page address 0-63)
Acknowledge:	1	1	F	F	F	F	F	X	(F = Status flags)

Data only follows if read was successful

Reply1:	D	D	D	D	D	D	D	D	(D = msb data read from HT1)
Reply2:	D	D	D	D	D	D	D	D	
Reply3:	D	D	D	D	D	D	D	D	
Reply4:	D	D	D	D	D	D	D	D	(D = lsb data read from HT1)

Tag_STATUS

Command to return Tag status.
The acknowledge byte flags indicate general Tag status.

	B7							B0	
Command:	0	1	0	1	0	0	1	1	(0x53)
Acknowledge:	1	1	F	F	F	F	F	X	(F = Status flags)

Message

Command to return product and firmware identifier string to host.

	B7 B0	
Command:	0 1 1 1 1 0 1 0	(0x7A)
Reply:	"b IDE RWD H1 WD (SECx V1.xx) DD/MM/YY" 0x00	

Returned string identifies author, product descriptor, project name, firmware version no. and date of last software change. Note that the string is always NULL terminated. The string begins with a unique lower case character that can be used to identify a particular version of Micro RWD.

NOTE that the serial communication uses hardware handshaking to inhibit the host from sending the Micro RWD commands while Tag interrogation is in progress.

NOTE also that for the Read Tag command, if an error flag has been set in the Acknowledge code then there will be NO following data.

Program EEPROM

The Micro RWD has some internal EEPROM for storing system parameters such as passwords and authorised identity codes. This command sequence allows individual bytes of the EEPROM to be programmed with new data. Note that due to the fundamental nature of these system parameters, incorrect data may render the system temporarily inoperable.

	B7 B0	
Command:	0 1 0 1 0 0 0 0	(0x50)
Argument1:	x x N N N N N N	(N = EEPROM memory location 0-127)
Argument2:	D D D D D D D D	(D = data to write to EEPROM)
Acknowledge:	1 1 X F X X X F	(F = Status flags)

Internal EEPROM memory map

Byte 0: Tag Polling Rate (x 2.5 ms) (default 100ms)
Byte 1: Wiegand data length 4 to 34 bit (even number), 0x00 = OFF (NO Wiegand output), default = 30 bit
Byte 2: Reserved (Checksum)
Byte 3: Encryption ON/OFF control byte (0x00 = OFF)

Byte 4:) 32 bit Encryption seed (M.S byte)
Byte 5:)
Byte 6:)
Byte 7:) (L.S byte)

Byte 8: Reserved
Byte 9: Reserved
Byte 10: Reserved
Byte 11: Reserved

Start of authorised tag codes. List is terminated with FF FF FF FF sequence.
List is regarded as empty (all identity codes valid) if first code sequence in list is (FF FF FF FF).

Byte 12: 0xFF Empty list
Byte 13: 0xFF
Byte 14: 0xFF
Byte 15: 0xFF

Byte 16: (MSB) Tag identity code
Byte 17:
Byte 18:
Byte 19: (LSB)

Byte 20: (MSB) Tag identity code
Byte 21:
Byte 22:
Byte 23: (LSB)
-
-
Byte 127: Last Internal EEPROM location

Encryption Methodology

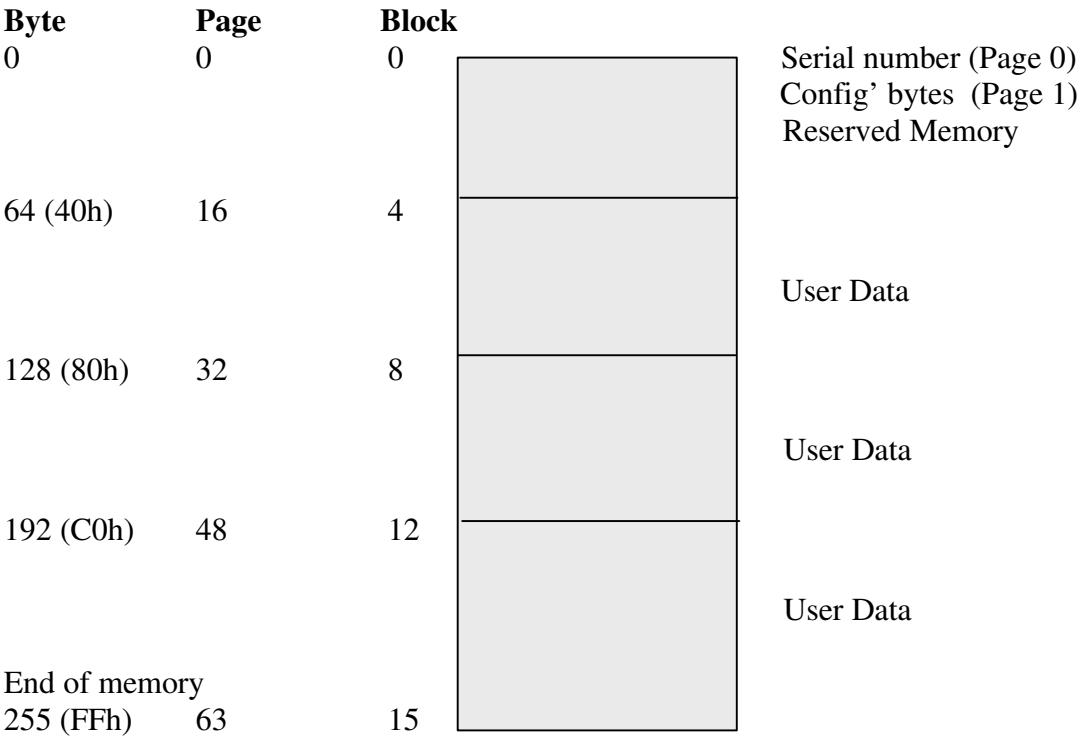
The Micro RWD H1 has a data encryption system that allows data to be stored in an encoded form that cannot be read as sensible data by any other Hitag 1 reader system.
On early versions of the Micro RWD (pre Ver. 1.20), data could be stored in the lower half of the transponder memory (from page 16) as plain data with no encryption applied and stored in the upper half (from page 32) in an encrypted form. The user could therefore choose the data storage area according to security requirements. From version 1.20 onwards the form of the stored data for the whole transponder memory (apart from the serial number, configuration and other data in blocks 0-3) is controlled by the Encryption Control byte in the Micro RWD internal EEPROM. If Encryption Control ON is selected then all data stored in the transponder from page 16 upwards will be encrypted, and if OFF is selected then all data is stored in standard format.
The method of encryption is unique and uses a “dynamic algorithm” which effectively makes the encoded data specific to a particular transponder and a set of encryption seed values stored in the Micro RWD internal EEPROM. This not only protects stored information but also prevents cloning of cards or copying of data. Information is encrypted when being stored and decrypted when being read, thereby making the process totally transparent to the user. Another Hitag 1 reader system would read encrypted data as random bytes with no meaning.
Users should program their own encryption seed values to fully customise their system.

Method of Operation

The Micro RWD reader only allows full communication with Hitag 1 transponders if an initial level of security has been passed. The system works by firstly reading the tag serial number (identity code) which is the four bytes from page 0 (first page) of the tag memory. The Micro RWD internal EEPROM is then checked to see if this serial number is stored in the authorisation list located from byte 12 onwards. If the tag serial number is matched to a serial number stored in the Micro RWD or the list is empty then the tag has passed the validation test. If the Micro RWD has FF FF FF FF (hex) stored at EEPROM locations 12 to 15 then the list is treated as empty and all Hitag 1 tags are accepted through the validation test.

Full communication is only allowed if this initial security check has been passed (or the Micro RWD authorisation list is empty).

Hitag 1 Memory Map



Pages 16 to 63 are available for user data storage (192 bytes). It is advised not to use the memory locations below page 16 because these are used for configuration and a different mode of operation not supported on the standard Micro RWD. These are marked as reserved by Philips Semiconductors.

Hitag 1 Configuration Bytes

The Hitag 1 transponder mode and whether the memory pages are locked or open for read/write operations is controlled by the configuration bytes. These are bytes 0 and 1 of Page 1 of the Hitag 1 memory. Note that bytes 2 and 3 of the configuration page are not used and are currently available for read/write use.

Configuration Byte 0 (Page 1, byte 0)

b7	b6	b5	b4	b3	b2	b1	b0	
1	**1**							
								0 = Block 7 read only
								1 = Block 7 read/write
								0 = Block 6 read only
								1 = Block 6 read /write
								0 = Block 5 read only
								1 = Block 5 read/write
								0 = Block 4 read only
								1 = Block 4 read/write
								0 = Block 3 read only
								1 = Block 3 read/write (not used)
								0 = Block 2 read only
								1 = Block 2 read/write (not used)

Configuration Byte 1 (Page 1, byte 1)

b7	b6	b5	b4	b3	b2	b1	b0	
							1	
								Reserved
								Reserved
								Reserved
								0 = Configuration (Page 1) read only
								1 = Configuration (Page 1) read/write
								Reserved
								Reserved
								Reserved

Note that these configuration bits are OTP. Once they are set to read-only the Hitag 1 transponder is hardware protected and they can never be changed.

Note that the “Reserved” bits of Configuration Byte 1 must not be altered. The current value of byte 1 must be read first and the bits that can be changed, masked on before writing back.