



# TagTracer<sup>®</sup> Multi

## Protocol Description

Firmware Version 0.12b

# Contents

<b>1</b>	
<b>OVERVIEW.....</b>	<b>3</b>
<b>2 HARDWARE</b>	
<b>INSTALLATION.....</b>	<b>3</b>
2.1 TAGTRACER MULTI DESKTOP.....	3
2.2 TAGTRACER MULTI OEM.....	3
2.3 PIN ASSIGNMENT TAGTRACER MULTI DESKTOP.....	3
2.4 TAGTRACER INDUSTRIE MULTI.....	3
<b>3 SOFTWARE</b>	
<b>INSTALLATION.....</b>	<b>3</b>
<b>4 PROTOCOL DESCRIPTION</b>	
<b>0.12B.....</b>	<b>4</b>
4.1 OPERATION PARAMETERS.....	4
4.2 OPERATION MODES.....	4
4.3 SUPPORTED TRANSPONDERS AND PREFIXES.....	5
4.4 COMMAND SET.....	6
<b>5 TRANSPONDER</b>	
<b>OVERVIEW.....</b>	<b>7</b>
5.1 EM4002 AND EM4102.....	7
5.1.1 Features.....	7
5.1.2 Memory Map.....	7
5.2 EM4050 AND EM4150.....	7
5.2.1 Features.....	7
5.2.2 Memory Map.....	7
5.2.3 Access Conditions.....	9
5.3 HITAG 1.....	10
5.3.1 Features.....	10
5.3.2 Restrictions.....	10
5.3.3 Memory Map.....	10
5.4 HITAG 2.....	11
5.4.1 Features.....	11
5.4.2 Restrictions.....	11
5.4.3 Memory Map.....	11
<b>6 PROGRAMMING EXAMPLES.....</b>	<b>13</b>
6.1 VISUAL BASIC.....	13
6.2 C/C++.....	14

## 1 Overview

This document describes usage of TagTracer® Multi. TagTracer® Multi is a reader/writer device for use with several different transponder types. TagTracer® Multi is available as OEM module for hardware integrator and as Desktop version which can be directly connected to a PC.

**Note:** Within this document we use the words *Tag* and *Transponder* with identical meaning.

## 2 Hardware Installation

### 2.1 TagTracer Multi Desktop

The hardware installation of the TagTrace Multi Desktop is very simple. The DSUB9 connector is plugged into a suitable COM port of a PC, the Western Modular connector is plugged into the appropriate connector of the TagTracer Desktop and finally, the power supply (wall cube type) is plugged into a 230 V mains outlet.

### 2.2 TagTracer Multi OEM

The hardware installation of the TagTracer Multi OEM requires more work. It needs a +5 V DC supply and a level shifter for the RS-232 signals. Please refer to Chapter 5.4.3.2 for a sample application circuit and to Chapter Fehler: Verweis nicht gefunden for the pin assignment for the OEM module. *Please keep in mind that reversing the power supply polarity will destroy the OEM module.*

### 2.3 Pin Assignment TagTracer Multi Desktop

Pin Number WM-Plug	Function
1	V <sub>in</sub> 9 ... 12 V AC/DC
2	GND
3	RxD (PC → TagTracer Multi)
4	TxD (TagTracer Multi → PC)
5	free
6	V <sub>in</sub> 9 ... 12 V AC/DC

### 2.4 TagTracer Industrie Multi

The hardware installation of the TagTrace Industrie Multi CDC-version is very simple. The USB-connector is plugged into a suitable COM port of a PC. You have to download the suitable PC driver from our Website.

## 3 Software Installation

The software is included in the development kit and on our Website: [www.ZeitControl.de](http://www.ZeitControl.de). The drivers can be found under the tab "Support" and "Downloads" and are called: USB CDC Driver.

The software setup is based on Windows® Installer. If the Windows® Installer is already installed on your system, you can proceed immediately, otherwise the Windows® Installer will be installed on your PC. To install the Windows® Installer, administration privileges are required.

Setup is started automatically once the CD labeled "TagTracer-Multi Software Setup" is inserted into your computers' CD-ROM drive. If setup does not start automatically just run Setup.exe from CD.

If you accept the license agreement, the TagTracer Multi software will be installed on your PC. If you don't change the target directory, a directory "TagTracer Multi" in your common program's directory is created.

Once Software is installed you can find all documents, demo software and sample code within your Start menu through Start, Programs, TagTracer Multi.

## 4 Protocol Description 0.12b

This description is based on the software release 0.12b

### 4.1 Operation Parameters

The TagTracer® Multi is controlled through a serial interface. The OEM module uses TTL level serial interface. The desktop version of TagTracer® Multi uses RS232 level interface and such can be directly connected to a computer serial port.

Communication parameters used are 9600,8,N,1. In detail:

- 9600 baud
- 1 start bit
- no parity
- 8 data bits
- 1 stop bit

No handshake, neither hardware nor XONOFF, is used.

The device does not echo received characters.

### 4.2 Operation Modes

The TagTracer® Multi has two operation modes. Following a reset the device is in auto detect mode. In this mode the device scans for transponders in RF field. If a transponder is detected the transponder serial number prefixed by a character identifying the transponder type is transmitted through the serial interface. This transmission is repeated as long as the transponder stays inside the RF field or the device is switched into command mode. If in auto detect mode any character sent to TagTracer® Multi will cause the device to switch to command mode.

### 4.3 Supported Transponders and Prefixes

Following transponders are supported by TagTracer® Multi:

Prefix	Tag	Properties	Available Commands
U	EM400x/410x	5 Bytes Read only	x,c,v,p
T	EM4050	32 x 4 Bytes Read write, Password	x,r,w,l,v,p
H	HITAG2	7 x 4 Bytes Read write, Password	x,r,w,l,v,p
h	HITAG1	64 x 4 Bytes Read write0	x,r,w,v,p

## 4.4 Command Set

Commands are accepted in command mode only. If the reader is in auto detect mode any character sent to the device will cause it to switch into command mode. Once switched into command mode the reader will respond with "S". Now you may send commands as described below.

Commands and responses are transmitted using ASCII characters. Numbers are also encoded using ASCII characters. All numbers use hexadecimal notation. Such each byte must be encoded using two hexadecimal characters.

All commands are case insensitive. E.g. "c" is equivalent to "C". Commands do not require to be terminated by <CR><LF>.

Responses from TagTracer® Multi are terminated by <CR><LF> (0Dh, 0Ah). All responses from the module use uppercase letters, except for the prefix.

For proper operation with read/write transponders the transponder has to be recognized in auto detect mode before issuing any command.

Function	Command to Reader	Answer from Reader
Change mode to auto detect mode.	<b>C</b>	none
Read a word or page	<b>RXX</b> Where XX is word or page to read in hexadecimal notation. Example: R02	Transponder type prefix followed by content of word or page. E.g.: T20210000 or <b>N</b> if no data available.
Write a word or page  <b>Note:</b> <ul style="list-style-type: none"> <li>Does a read after write to make sure, data was written, The read data is replied in the answer from the reader</li> <li>Protected words and password require login before writing</li> <li>Writing to none readable words or pages (e.g. passwords) returns W0000000000 because memory is not readable</li> </ul>	<b>WXXDDDDDDDD</b> Where XX is word or page to write and DDDDDDDD is data to write. Both use hexadecimal notation. Example (write A1B2C3E4h to page 10h): W10A1B2C3E4	<b>W</b> followed by data read from word or page previously written. E.g. if write operation example was successful: WA1B2C3E4 If write has failed <b>N</b> is returned.
Login for protected operations  <b>Note:</b> Login needs to be performed only once as long as no logout takes place. Logout is through the Reset function, by removing the tag from the field or by turning off the RF field.	<b>LPPPPPPP</b> Where PPPPPPPP is the password in hexadecimal notation. Example: L00000000	<b>L</b> if successful or <b>N</b> if failed.  <b>Note:</b> Depending on transponder type an invalid login (password) may not be detected at once ( <b>N</b> response) instead next read/write operation will fail.
Reset the TagTracer® Multi device. (does a logout and restarts reader)	<b>X</b>	Device ID and version. E.g.: MULTITAG 0.12a
Power saving mode (turns RF field off)  <b>Note:</b> The RF-field gets turned on when the next command is received The CPU stays on	<b>P</b>	<b>P</b>

Get Version	<b>V</b>	Device ID and version. E.g.: MULTITAG 0.12a
-------------	----------	--

## 5 Transponder Overview

In following chapters you will find a short overview about each supported transponder type. This information may be incomplete, but should be enough for most cases. More detailed information is available from the silicon manufacturers of transponder chips used.

### 5.1 EM4002 and EM4102

#### 5.1.1 Features

- 40 bit serial number (read only)

#### 5.1.2 Memory Map

There is no special memory organization within this tag. It is recommended to use this chip in auto detect mode of TagTracer® Multi. TagTracer® Multi will return the 40 bit serial number following the prefix **U** when in auto detect mode.

In opposite to other ZeitControl EM4102 reader (like Minimagic) the number is shown in a bit reversed orientation. Please check the "Programming example" chapter 6 for some conversion routines.

### 5.2 EM4050 and EM4150

#### 5.2.1 Features

- 1 kBit of EEPROM organized in 32 words of 32 bits.
- 32 bit serial number (read only)
- 32 bit device ID (read only)
- Password protection

#### 5.2.2 Memory Map

Word	Function
00h	Password (write only)
01h	Protection Word
02h	Control Word
03h .. 1Fh	User Memory
20h	Device Serial Number (read only)
21h	Device ID (read only)

##### 5.2.2.1 Password

The transponder password will be required for use with login command. Factory default password is **00000000**. You can change the password by write to word 00h (**W00DDDDDDDD**). Changing password requires a successful login to the transponder. So you must have executed login using current password (e.g. **L000000000**) before. Because password is read protected the reader will always respond with **W0000000000** when writing to word 0 of EM4050 or EM4150. Password change will take place immediately, so be careful.

Example (change password from 00000000h to 42424242h):

- ✱ L00000000
- ✱ L<CR><LF>
- ✱ W0042424242
- ✱ W00000000<CR><LF>

#### 5.2.2.2 Protection Word

The protection word specifies which parts of memory should be read protected or write inhibited. Read protected memory can be read only if a successful login was executed. Write inhibited memory cannot be written. To write to write inhibited memory you must first change the protection word to remove write inhibited condition for this memory location.

You can change the protection word by write to word 01h (**W01FFLLIJJ**). Changing protection word requires a successful login to the transponder. Changes does not take place immediately. First the transponder must be reset, e.g. by reset the device (**R**).

The protection word is organized as follows (with regards to hexadecimal notation used with TagTracer® Multi):

*FFLLIJJ*

With

- *FF* first word to be read protected.
- *LL* last word to be read protected
- *//* first word to be write inhibited
- *JJ* last word to be write inhibited

Example (make user memory from word 3 to word 7 read protected and write inhibit memory from word 3 to word 10):

- ✱ W010307030A
- ✱ W0307030A <CR><LF>

#### 5.2.2.3 Control Word

The control word specifies which memory should be transferred in standard read mode. Standard read mode is not used by this device, but changing this may affect the function of a transponder when using with different reader devices.

The control word also specifies if password protection is active, such that a login is required to access read protected memory and to write to user memory. Please note: To write password, protection word or control word a login is always required, even if password check is disabled.

Further the control word specifies the read after write mode (on or off).

You can change the control word by write to word 02h (**W02FFLLUOUU**). Changing control word requires a successful login to the transponder. Changes does not take place immediately. First the transponder must be reset, e.g. by reset the device (**R**).

The control word is organized as follows (with regards to hexadecimal notation used with TagTracer® Multi):



## FFLLUOUU

With

- *FF* first word to be read in standard mode.
- *LL* last word to be read in standard mode
- *O* password and read after write option which may be:
  - **0** password check and read after write disabled
  - **1** password check enabled, read after write disabled
  - **2** password check disabled read after write enabled
  - **3** password check and read after write enabled
- *U* user available (no special meaning use as you like)

Example (enable password check):

✱ W0200000100A

★ W00000100 <CR><LF>

### 5.2.3 Access Conditions

#### 5.2.3.1 Read Access

The password (word 0) can never be read. All other location depend on configuration in protection word. Password check option as configured in control word has no meaning for read access. When reading read protected memory without login, 00000000 is returned instead of real memory content. No error condition is returned.

#### 5.2.3.2 Write Access

Read only sections like device serial can never be written.

Password (word 0), protection word (word 1) and control word (word 2) always require a login to be written.

If password check is enabled, write operations (to any word) are not possible without login.

Write inhibited words cannot be written even after login. To write to write inhibited memory you must first change the protection word to remove write inhibited condition for this memory location.

Read protection does not prevent write operation to a memory location. E.g. if memory word 3 to 7 is read protected and password check is disabled, then you cannot not read word 3 without login, but you can change word 3 without login.

## 5.3 Hitag 1

In following chapters you find all none confidential information available to use Hitag 1 transponders. For more detailed description of Hitag 1 transponders please contact Philips. More (password protected) information can be found on Philips Semiconductor web page (<http://www.semiconductors.philips.com/>). Since most of these information are confidential we cannot provide more detailed information.

### 5.3.1 Features

- 2 kBit of EEPROM organized in 16 blocks. Every block consists of 4 pages with 32 bit each.
- 32 bit serial number (read only)

### 5.3.2 Restrictions

Following restrictions apply when using Hitag 1 transponders with TagTracer® Multi.

- Crypto and authentication functions are not supported. Because of this secret pages cannot be accessed.
- Anti collision is not supported
- Passwords are not supported by Hitag 1

### 5.3.3 Memory Map

Following memory map shows a default configuration as delivered by Philips. Configuration may be changed by changing configuration page. Pages marked secret (including configuration page) cannot be accessed with TagTracer® Multi. This limitation exists, because access to secret pages would require a special crypto hardware which is not part of TagTracer® Multi.

Block	Page	Function	Access Restrictions
0h	00h	Serial Number	public (read only)
	01h	Configuration	public
	02h	Key A	secret
	03h	Key B	secret
1h	04h	Logdata 1B	secret
	05h	Logdata 0A	secret
	06h	Logdata 1A	secret
	07h	Logdata 0B	secret
2h .. 3h	08h .. 0Fh	User Data	secret
4h .. Fh	10h .. 3Fh	User Data	public

## 5.4 Hitag 2

### 5.4.1 Features

- 256 bit of EEPROM organized in 8 pages with 32 bit each
- 32 bit serial number (read only)
- Password protection

### 5.4.2 Restrictions

Following restrictions apply when using Hitag 2 transponders with TagTracer® Multi.

- Crypto mode, Mode A, Mode B and Mode C mode are not supported. Password mode must be used.
- Biphase coding for Hitag 2 mode is not supported.
- Anti collision is not supported by Hitag 2

### 5.4.3 Memory Map

Following description shows the memory map of Hitag 2 transponders in password mode. Password mode is the factory default configuration for Hitag 2 transponders. TagTracer® Multi does support password mode only. Access conditions to the pages of Hitag 2 can be configured using the configuration page.

Page	Function
0h	Serial Number (read only)
1h	Password RWD
2h	reserved (do not use)
3h	8 bit Configuration and 24 bit Password TAG
4h .. 7h	User Data

#### 5.4.3.1 Password RWD and Password TAG

Factory defaults are:

- Password RWD: 4D494B52h (“MIKR”)
- Password TAG: AA4854h

These two passwords are used for a mutual authentication between transponder and reader device. The passwords can be changed by changing the matching pages. Password RWD is contained in page 1. Password TAG is contained in 24 least significant bits of page 3. 8 most significant bits contain configuration settings and such must be left unchanged when changing password TAG.

A login using password RWD is required, before the transponder memory can be accessed. If the factory default password is set a login is not required in some cases. Still it is recommended to do a manual login for default password as well.

Password TAG is sent by Hitag 2 transponder for authentication to the reader device. TagTracer® Multi accepts all passwords.

### 5.4.3.2 Configuration

The 8 most significant bits of page 3 contain the configuration settings. These have following meaning:

Bit	Function
7	<p><b>Note: This bit is OTP, once set to 1 it cannot set back to 0</b>            default: 0            0: Page 1 and 2 read/write            1: Page 1 no read, no write. Page 2 read only when in password mode. Page 2 no read/write when in crypto mode.</p>
6	<p><b>Note: This bit is OTP, once set to 1 it cannot set back to 0</b>            default: 0            0: Page 3 read/write            1: Page 3 read only. Configuration byte and password TAG fixed.</p>
5	<p>default: 0            0: Page 4 and 5 read/write            1: Page 4 and 5 read only</p>
4	<p>default: 0            0: Page 6 and 7 read/write            1: Page 6 and 7 read only</p>
3	<p>default: 0            0: Password mode            1: Crypto mode (not supported by TagTracer® Multi)</p>
2, 1	<p>default: 11            0, 0: Public mode B, biphase coding (not supported by TagTracer® Multi)            0, 1: Public mode A, manchester coding (not supported by TagTracer® Multi)            1, 0: Public mode A, biphase coding (not supported by TagTracer® Multi)            1, 1: Hitag 2 mode</p>
0	<p>default: 0            0: Use manchester coding for Hitag 2 mode            1: Use biphase coding for Hitag 2 mode (not supported by TagTracer® Multi)</p>

## 6 Programming examples

### 6.1 Visual Basic

This function (SwapTransID) converts a Multitag 4102 Transponder ID to a Minimagic 4102 Transponder ID. This conversion is lossless.

usage is :

```
minimagicID = SwapTransID(multitagID)
```

```
Private Function TransConvert(value As String) As String
    Select Case value
        Case "0": TransConvert = "0"
        Case "1": TransConvert = "8"
        Case "2": TransConvert = "4"
        Case "3": TransConvert = "C"
        Case "4": TransConvert = "2"
        Case "5": TransConvert = "A"
        Case "6": TransConvert = "6"
        Case "7": TransConvert = "E"
        Case "8": TransConvert = "1"
        Case "9": TransConvert = "9"
        Case "A": TransConvert = "5"
        Case "B": TransConvert = "D"
        Case "C": TransConvert = "3"
        Case "D": TransConvert = "B"
        Case "E": TransConvert = "7"
        Case "F": TransConvert = "F"
    End Select
End Function

Public Function SwapTransID(value As String) As String
    Dim i As Integer
    Dim c As String
    Dim result As String

    For i = 1 To Len(value) Step 2
        c = Mid$(value, i + 1, 1)
        result = result + TransConvert(c)
        c = Mid$(value, i, 1)
        result = result + TransConvert(c)
    Next i

    SwapTransID = result
End Function
```

## 6.2 C/C++

This code segment can be inserted anywhere in the code (eg. after a read routine). The Minimagic transponder id has to be in the xbuf array ( at least: char xbuf[11]). Result after conversion will also be in the xbuf array.

Note that xbuf[0] is the Transponder type ID (in this case 'U').

```
{
    int i;

    for (i = 0; i < 5; i++)
    {
        const char conv[] =
        {
            // 0-9
            '0', '8', '4', 'C', '2', 'A', '6', 'E', '1', '9',
            // dummy
            '-', '-', '-', '-', '-', '-', '-',
            // A-F
            '5', 'D', '3', 'B', '7', 'F'
        };

        unsigned char c = xbuf[i*2 + 1];

        xbuf[i*2+1] = conv[xbuf[i*2 +2] - '0'];
        xbuf[i*2+2] = conv[c - '0'];
    }
}
```