



**Disclaimer:** The information in this document is provided in connection with Telecom Design products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Telecom Design products.

TELECOM DESIGN ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL TELECOM DESIGN BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF TELECOM DESIGN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Telecom Design makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Telecom Design does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Telecom Design products are not suitable for, and shall not be used in, automotive applications. Telecom Design products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© 2013-2014 Telecom Design S.A. All rights reserved. Telecom Design®, logo and combinations thereof, are registered trademarks of Telecom Design S.A. SIGFOX™ is a trademark of SigFox S.A.. Other terms and product names may be trademarks of others.

## 1 Overview

Thank you for choosing the TD1202 SIGFOX™ gateway module from Telecom Design!

This document provides a reference manual for the Telecom Design TD1202 module.

As an overview, this chapter gives the scope of this document. The document's organization is then detailed, followed by a list of relevant documents.

### 1.1 Scope

The TD1202 module provides an integrated dual AT command interpreter for interfacing with an external application over a serial link.

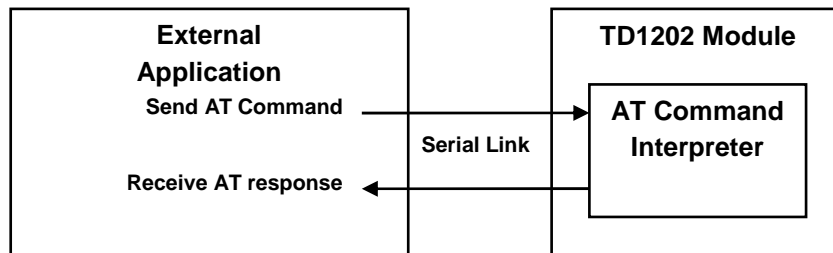


Figure 1- AT Command Interpreter

This guide focuses on the description of the commands and responses provided by the TD1202 module used in the communication with the external application.

Note: The “AT” command interpreter is only available since firmware revision SOFT1104.

### 1.2 Organization

Each section in this document covers a separate topic, organized as follow:

- Section 1 is this overview
- Section 2 is a Hayes-compatible AT command set reference
- Section 3 is a Hayes-compatible AT Sensor command set reference

### 1.3 Relevant Documents

This document provides a reference manual for the TD1202 SIGFOX™ Gateway module. Additional information on this module and on its dedicated evaluation board can be found in:

- *TD1202 Datasheet*
- *TD1202 EVB User's Guide*

The following standards are also referenced:

- *ITU-T Rec. T.50 (09/92) International Reference Alphabet (IRA)*
- *ISO/IEC 646:1991, Information technology — ISO 7-bit coded character set for information interchange*

## TABLE OF CONTENTS

---

<b><u>Section</u></b>	<b><u>Page</u></b>
<b>1 Overview.....</b>	<b>3</b>
1.1 Scope .....	3
1.2 Organization .....	3
1.3 Relevant Documents .....	3
<b>2 TD1202 Hayes-Compatible AT Command Set Reference.....</b>	<b>6</b>
2.1 TD1202 Factory Settings.....	6
2.2 AT Commands Presentation Rules.....	6
2.3 Information Responses and Result Codes .....	7
2.4 AT Command Syntax Format .....	7
2.5 Attention Command.....	9
2.6 Repeat Last Command A/ .....	10
2.7 Help ? .....	11
2.8 Echo Activation E .....	12
2.9 Information Display Control I .....	13
2.10 Result Code Suppression Q .....	15
2.11 Restart Banner Display S200.....	16
2.12 Format Control V.....	18
2.13 Extended Result Code X .....	19
2.14 Default Configuration Z.....	20
2.15 Restore Factory Settings &F .....	21
2.16 Display Configuration &V .....	22
2.17 Save Configuration &W .....	24
<b>3 TD1202 Hayes-Compatible AT LAN Command Set Reference .....</b>	<b>25</b>
3.1 Introduction to the TD LAN.....	25
3.2 TD LAN Address S400 .....	27
3.3 TD LAN Mask S401 .....	28
3.4 TD LAN Separator S402 .....	29
3.5 TD LAN Frequency S403 .....	31
3.6 TD LAN Power S404 .....	32
3.7 TD LAN Receive RF Message \$RL .....	33
3.8 TD LAN Send RF Message \$SL .....	35
<b>4 TD1202 Hayes-Compatible AT Sensor Command Set Reference .....</b>	<b>36</b>

---

4.1	Introduction to Sensor .....	36
4.2	Sensor and LAN .....	37
4.3	TD1202 Module Type .....	37
4.4	Examples .....	38
4.5	TD SENSOR – Module Type S500 .....	40
4.6	TD SENSOR – Device Class S501 .....	42
4.7	TD SENSOR – Battery Monitoring S502 .....	43
4.8	TD SENSOR – Temperature Monitoring S503.....	45
4.9	TD SENSOR – RSSI Monitoring S504 .....	47
4.10	TD SENSOR – Connection Monitoring S505.....	49
4.11	TD SENSOR – Switch Monitoring S506 .....	51
4.12	TD SENSOR – Boot Monitoring S507.....	53
4.13	TD SENSOR – Keep-Alive Monitoring S508 .....	54
4.14	TD SENSOR LAN – Send Data \$LD=.....	55
4.15	TD SENSOR LAN – Open Registration \$LR= .....	56
4.16	TD SENSOR LAN – Registration \$LR.....	57
4.17	TD SENSOR LAN – Get Address \$LA? .....	58
4.18	TD SENSOR LAN – LAN Reset \$LZ.....	59
4.19	TD SENSOR SEND - Data Phone \$DP.....	60
4.20	TD SENSOR SEND - Event Boot \$EB.....	61
4.21	TD SENSOR SEND - Event Connection \$EC.....	62
4.22	TD SENSOR SEND - Event RSSI \$ER .....	63
4.23	TD SENSOR SEND - Event Switch \$ES .....	64
4.24	TD SENSOR SEND - Event Temperature \$ET .....	65
4.25	TD SENSOR SEND - Event Battery \$EV .....	66
4.26	TD SENSOR SEND - Event Keep-Alive \$KA.....	67
4.27	TD SENSOR SEND - Service SMS \$SSMS.....	68
4.28	TD SENSOR SEND - Service Tweet \$STWT .....	69
4.29	TD SENSOR SEND - Raw \$RAW= .....	70
4.30	TD SENSOR SEND - Event Register \$REG .....	71

## 2 TD1202 Hayes-Compatible AT Command Set Reference

This section provides a reference for the Hayes-compatible “AT” commands implemented by the TD1202 Module.

### Notes:

- The “AT” command interpreter is only available since firmware revision SOFT1104. In older revisions, including SOFT1093, the “AT” command interpreter is not functional.
- Because of the required replacement of SIGFOX protocol V0 by SIGFOX protocol V1, from firmware version SOFT1278 and above, it is no longer possible to fit the “AT LAN” and “AT Sensor” Command Set into the TD1202 Flash memory. If your application requires these “AT LAN” or “AT Sensor” Command Sets, you should use the footprint-compatible TD1208 module instead.

### 2.1 TD1202 Factory Settings

A serial link handler is set with the following default values (factory settings):

- LVTTL electrical level
- Speed 9600 bps
- 8 data bits
- 1 stop bit
- No parity
- No hardware/software flow control

The following AT command interpreter settings are set up as factory defaults:

- Character echo is activated (“**ATE1**”)
- Format control is set to verbose (“**ATV1**”)
- Result codes are activated (“**ATQ0**”)
- Extended result codes are activated (“**ATX1**”)

### 2.2 AT Commands Presentation Rules

The AT commands are presented in the present documentation as follows:

- A ‘Description’ section provides general information on the AT command (or response) behavior
- A ‘Syntax’ section describes the command and response syntaxes and all parameters description
- A ‘Parameters and Defined Storage’ section describes all parameters and values
- A ‘Parameter Storage’ presents the command used to store the parameter value and/or the command to restore the parameter value
- An ‘Examples’ section presents the real use of the described command
- A ‘Notes’ section can also be included indicating some remarks about the command use

Figures are provided where necessary.

The commands will be listed alphabetically.

## 2.3 Information Responses and Result Codes

If format control is set to verbose (“**ATV1**”) and result codes are activated (“**ATQ0**”), the TD1202 module returns the “<CR><LF>^SYSSTART<CR><LF>” string upon device reset.

If command syntax is incorrect or the command contains wrong parameters, or the command cannot be executed successfully, the “**ERROR**” string is returned if format control is set to verbose (“**ATV1**”), or a “**4**” string is returned otherwise.

If the command line has been executed successfully, an “**OK**” string is returned if format control is set to verbose (“**ATV1**”), or a “**0**” string is returned otherwise.

## 2.4 AT Command Syntax Format

Command lines always start with “**AT**” and finish with a “<CR>” character, except for the “**A/**” command, and contain one or more commands.

Responses start and end with “<CR><LF>”, except for the “**ATV0**” TD1202 module response format and the “**ATQ1**” (result code suppression) commands.

In the following examples “<CR>” and “<CR><LF>” characters are intentionally omitted.

### 2.4.1 Basic AT Command Syntax Format

The syntax of basic commands is:

```
<command>[<number>]
```

Where “<command>” is either a single character, a “?” character (IA5 3/15), or the “&” character (IA5 2/6) followed by a single character. Characters used in “<command>” shall be taken from the set of alphabetic characters.

“<number>” may be a string of one or more characters from “0” through “9” representing a decimal integer value. Commands that expect a “<number>” are noted in the description of the command. If a command expects “<number>” and it is missing (“<command>” is immediately followed in the command line by another “<command>” or the termination character), the value “0” is assumed. If a command does not expect a “<number>” and a number is present, an “**ERROR**” is generated. All leading “0”s in “<number>” are ignored by the TD1202 AT command interpreter.

### 2.4.2 S-parameters

Commands that begin with the letter “s” constitute a special group of parameters known as “S-parameters”. These differ from other commands in important respects.

The number following the “s” indicates the “parameter number” being referenced. If the number is not recognized as a valid parameter number, an “**ERROR**” result code is issued.

Immediately following this number, either a “?” or “=” character (IA5 3/15 or 3/13, respectively) shall appear. “?” is used to read the current value of the indicated S-parameter; “=” is used to set the S-parameter to a new value.

```
S<parameter_number>?
S<parameter_number>=[<value>]
S<parameter_number>=?
```

# TD1202 RM

If the “?” is used, the TD1202 module transmits a single line of information text to the external application. The ranges of returned values are given in the description of each S-parameter.

If the “=” is used, the new value to be stored in the S-parameter is specified in decimal following the “=”. If no value is given (i.e., the end of the command line occurs or the next command follows immediately), the S-parameter specified may be set to 0, or an “**ERROR**” result code issued and the stored value left unchanged. The ranges of acceptable values are given in the description of each S-parameter.

If the “=?” is used, the TD1202 module transmits a single line of information text to the external application, giving the ranges of accepted values as given in the description of each S-parameter.

## 2.4.3 Extended AT Command Syntax Format

The syntax of extended commands is:

```
$<command>=[<value1>[,<value2>]...]  
$<command>?
```

Where “\$” is an ISO/IEC 646 (US) code 24 character, and “<command>” is made up of multiple characters taken from the set of alphabetic characters.

The first form is used as an “action” command. In this form, the “<command>” is followed by an “=” character (IA5 3/13) and zero or more “<valuex>”, separated by “,” characters (IA5 2/12). If no value is given (i.e., the end of the command line occurs or the next command follows immediately), the extended command parameter specified may be set to 0, or an “**ERROR**” result code issued and the stored value is left unchanged.

“<valuex>” may be a string of one or more alphanumeric characters from “0” through “9”, “a” to “z” or “A” to “Z”. The range of acceptable values is given in the description of each extended command.

The second form is used as a “read” command. In this form, the “<command>” is followed by a “?” character (IA5 3/15). In this form, the TD1202 AT command interpreter transmits a single line of information text to the external application. For extended commands defined in this specification, the text portion of this information consists of one or more alphanumeric characters from “0” through “9”, “a” to “z” or “A” to “Z”. The range of possible values is given in the description of each extended command.

## 2.4.4 AT Command Concatenation

Concatenation of multiple AT commands on the same line is not supported in the current firmware version.



## 2.5 Attention Command

### 2.5.1 Description

This command always returns OK.

### 2.5.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.5.3 Syntax

*Action Command*

**AT**

OK

*Read Command*

None

*Test Command*

None

### 2.5.4 Parameters and Defined Values

None

### 2.5.5 Parameter Storage

None

### 2.5.6 Examples

Command	Responses
AT	OK

## 2.6 Repeat Last Command A/

### 2.6.1 Description

This command repeats the last command of the open session. Only the A/ command itself cannot be repeated.

### 2.6.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.6.3 Syntax

*Action Command*

**A/**

*Note: the response depends on the previous command*

*Read Command*

None

*Test Command*

None

### 2.6.4 Parameters and Defined Values

None

### 2.6.5 Parameter Storage

None

### 2.6.6 Examples

Command	Responses
ATI	Telecom Design TD1202 OK
A/ <i>Note: Repeat last command</i>	Telecom Design TD1202 OK

### 2.6.7 Notes

Unlike all other AT commands, this command is executed immediately after the last “/” character is received, without waiting for a finish “<CR>” character.

## 2.7 Help ?

### 2.7.1 Description

This command displays a summary of the available AT commands.

Note: This command is a Telecom Design extension to the basic AT command set.

### 2.7.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.7.3 Syntax

*Action Command*

**AT?**

*Note: The response depends on the list of available commands*

*Read Command*

None

*Test Command*

None

### 2.7.4 Parameters and Defined Values

None

### 2.7.5 Parameter Storage

None

### 2.7.6 Examples

Command	Responses
<b>AT?</b> <i>Note: display a summary of available AT commands</i>	<list of commands> OK <i>Note: The response depends on the list of available commands</i>

## 2.8 Echo Activation E

### 2.8.1 Description

This command is used to determine whether the TD1202 module echoes characters received by an external application (DTE) or not.

### 2.8.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.8.3 Syntax

*Action Command*

**ATE[<n>]**

OK

*Read Command*

None

*Test Command*

None

### 2.8.4 Parameters and Defined Values

Parameter	Value	Description
<n>		<b>Echo Activation Parameter</b>
	0	Characters are not echoed
	1	Characters are echoed (default value)

### 2.8.5 Parameter Storage

The <n> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 2.8.6 Examples

Command	Responses
<b>ATE0</b> <i>Note: characters are not echoed</i>	OK <i>Note: Done</i>
<b>ATE1</b> <i>Note: characters are echoed</i>	OK <i>Note: Done</i>

## 2.9 Information Display Control I

### 2.9.1 Description

This command causes the product to transmit one or more lines of specific information text.

### 2.9.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.9.3 Syntax

*Action Command*

**ATI[<n>]**

Depending on <n>

OK

*Read Command*

None

*Test Command*

None

### 2.9.4 Parameters and Defined Values

Parameter	Value	Description
<n>		<b>Information Display Control Parameter</b>
	0	Display manufacturer followed by model identification (default value). Always returns the string "Telecom Design TD1202".
	5	Display firmware release date. Returns a string "mm+yyyy", where "mm" is a 2-digit month number and "yyyy" is a 4-digit year.
	7	Display RF serial number (SIGFOX™ ID) as a variable 4 to 8-digit hexadecimal number.
	10	Display baseband unique ID as a 16-digit hexadecimal number.
	11	Display hardware revision number as a 2-digit hexadecimal number.
	13	Display firmware revision number as a string "SOFTxxx", where "xxx" is a 4-digit number.
	21	Display RF chip part number as a string.
	25	Display RF chip ROM ID as a decimal number: <div> <div></div> 2: Revision 0B <div></div> 3: Revision 1B </div>
	26	Display module temperature in °C as a decimal number.
	27	Display module idle power supply voltage as a decimal number with 2 decimal places separated by a dot character.
	28	Display module RF active power supply voltage as a decimal number with 2 decimal places separated by a dot character. <b>Note:</b> the value is acquired during every SIGFOX™ RF transmission. If no value is available, the "0.00" value is returned. This value is the recommended value to monitor a battery supply, as it provides the supply voltage with the maximum load.

# TD1202 RM

---

## 2.9.5 Parameter Storage

None

## 2.9.6 Examples

Command	Responses
<b>ATI0</b> <i>Note: Manufacturer and model identification</i>	Telecom Design TD1202 OK <i>Note: Command valid</i>
<b>ATI5</b> <i>Note: result firmware release date</i>	M08+2012 OK <i>Note: August 2012</i>
<b>ATI7</b> <i>Note: serial number (SIGFOX™ ID)</i>	1058 OK <i>Note: Serial number 1058</i>
<b>ATI10</b> <i>Note: baseband unique ID</i>	209531004F62DFF9 OK <i>Note: baseband unique ID 209531004F62DFF9</i>
<b>ATI11</b> <i>Note: hardware revision number</i>	0F OK <i>Note: hardware 0F</i>
<b>ATI13</b> <i>Note: firmware revision number</i>	SOFT1100 OK <i>Note: firmware SOFT1100</i>
<b>ATI21</b> <i>Note: RF chip part number</i>	Si4461 OK <i>Note: RF chip is Si4461</i>
<b>ATI25</b> <i>Note: RF chip revision number</i>	2 OK <i>Note: RF chip revision is 0B</i>
<b>ATI26</b> <i>Note: module temperature</i>	21 OK <i>Note: module temperature is 21°C</i>
<b>ATI27</b> <i>Note: module idle power supply voltage</i>	3.25 OK <i>Note: module idle power supply voltage is 3.25 V</i>
<b>ATI28</b> <i>Note: module RF active power supply voltage</i>	3.22 OK <i>Note: module RF active power supply voltage is 3.22 V</i>

## 2.10 Result Code Suppression Q

### 2.10.1 Description

This command determines whether the TD1202 module sends result codes or not.

### 2.10.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.10.3 Syntax

*Action Command*

**ATQ[<n>]**  
[OK]

*Read Command*

None

*Test Command*

None

### 2.10.4 Parameters and Defined Values

Parameter	Value	Description
<n>		<b>Result Code Suppression Parameter</b>
	0	The TD1202 module transmits result codes (default value)
	1	Result codes suppressed and not transmitted

### 2.10.5 Parameter Storage

The <n> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 2.10.6 Examples

Command	Responses
<b>ATQ0</b> <i>Note: the TD1202 transmits result codes</i>	<b>OK</b> <i>Note: Command valid</i>
<b>ATQ1</b> <i>Note: result codes are suppressed and not transmitted</i>	<i>Note: No response</i>

## 2.11 Restart Banner Display S200

### 2.11.1 Description

This command configures, queries or disables the restart banner display.

### 2.11.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.11.3 Syntax

*Action Command*

**ATS200=[<display>]**

OK

*Read Command*

**ATS200?**

<display>

OK

*Test Command*

**ATS200=?**

0..1

OK

### 2.11.4 Parameters and Defined Values

Parameter	Value	Description
<display>		<b>SIGFOX™ Keepalive Period Parameter</b>
	0	Disable restart banner display (default value)
	1	Enable restart banner display

### 2.11.5 Parameter Storage

The <display> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 2.11.6 Examples

Command	Responses
<b>ATS200?</b>	0 OK <i>Note: restart banner display is disabled</i>
<b>ATS200=1</b> <i>Note: enable the restart banner display</i>	OK <i>Note: Command valid</i>
<b>AT&amp;W</b> <i>Note: save parameter value in Flash memory</i>	OK <i>Note: Command valid</i>
<b>ATZ</b> <i>Note: restart the TD1202 module</i>	OK ^SYSSTART <i>Note: Command valid and restart banner is displayed</i>



## 2.11.7 Notes

The restart banner is only displayed if this parameter is enabled (“**ATS200=1**”), format control verbosity is enabled (“**ATV1**”) and result code suppression is disabled (“**ATQ0**”) in Flash memory during restart.

## 2.12 Format Control V

### 2.12.1 Description

This command determines whether the TD1202 module response format uses or not the header characters <CR><LF>, and whether the result codes are provided as numeric or verbose.

### 2.12.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.12.3 Syntax

*Action Command*

**ATV<n>**

OK

*Read Command*

None

*Test Command*

None

### 2.12.4 Parameters and Defined Values

Parameter	Value	Description	
<n>	0	Format Control Parameter	
		Information Responses	Result Code
		<text><CR><LF>	<numeric code><CR>
	1 (default value)	(default value)	
		Information Responses	Result Code
		<CR><LF> <text><CR><LF>	<CR><LF> <verbose code><CR><LF>

### 2.12.5 Parameter Storage

The <n> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 2.12.6 Examples

Command	Responses
<b>ATV0</b> <i>Note: the TD1202 transmits limited headers and trailers and numeric result codes</i>	0 <i>Note: Command is valid (0 means OK)</i>
<b>ATV1</b> <i>Note: result the TD1202 transmits full headers and trailers and verbose response text</i>	OK <i>Note: Command valid</i>

## 2.13 Extended Result Code X

### 2.13.1 Description

This command determines whether the TD1202 module sends extended result codes or not.

Note: this command is only included for compatibility reason, as the TD1202 module does not feature any result code requiring an extension.

### 2.13.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.13.3 Syntax

*Action Command*

**ATX[<n>]**

OK

*Read Command*

None

*Test Command*

None

### 2.13.4 Parameters and Defined Values

Parameter	Value	Description
<n>		<b>Extended Result Code Parameter</b>
	0	The TD1202 module transmits normal result codes
	1	The TD1202 module transmits extended result codes (default value)

### 2.13.5 Parameter Storage

The <n> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 2.13.6 Examples

Command	Responses
<b>ATX0</b> <i>Note: the TD1202 transmits normal result codes</i>	OK <i>Note: Command valid</i>
<b>ATX1</b> <i>Note: the TD1202 transmits extended result codes</i>	OK <i>Note: Command valid</i>

## 2.14 Default Configuration Z

### 2.14.1 Description

This command restores the configuration profile from non-volatile memory (Flash).

Note: As a Telecom Design extension, this command also resets the TD1202 module to its power-on state.

### 2.14.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.14.3 Syntax

*Action Command*

**ATZ**

OK

*Read Command*

None

*Test Command*

None

### 2.14.4 Parameters and Defined Values

None

### 2.14.5 Parameter Storage

None

### 2.14.6 Examples

Command	Responses
ATS302?	14 OK <i>Note: Default value is in Flash memory</i>
ATS302=10 <i>Note: change transmission in volatile memory only</i>	OK <i>Note: Command valid</i>
ATZ	OK <i>Note: Command valid, will restart immediately</i>
ATS302?	14 OK <i>Note: Default value set back from Flash memory</i>

## 2.15 Restore Factory Settings &F

### 2.15.1 Description

This command is used to restore factory settings from Flash memory.

### 2.15.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.15.3 Syntax

*Action Command*

**AT&F**

OK

*Read Command*

None

*Test Command*

None

### 2.15.4 Parameters and Defined Values

None

### 2.15.5 Parameter Storage

None

### 2.15.6 Examples

Command	Responses
<b>AT&amp;F</b> <i>Note: Asks for restoring the factory settings</i>	<b>OK</b> <i>Note: Done</i>

### 2.15.7 Notes

For each parameter, the section “Parameter Storage” specifies which default values can be restored using **AT&F**. The parameters are restored in RAM and in Flash memory, overwriting the profile set with **AT&W**.

## 2.16 Display Configuration &V

### 2.16.1 Description

This command is used to display the TD1202 module configuration.

### 2.16.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

### 2.16.3 Syntax

#### Action Command

##### AT&V

Telecom Design TD1202

Hardware Version: <hardware\_revision>

Software Version: <firmware\_revision>

S/N: <serial\_number>

ACTIVE PROFILE

E<val1> V<va2> Q<val3> X<val4> S200:<val5> S300:<val6> S301:<val7> S302:<val8>

S303:<val9>

OK

*Note: For each <valx> parameter value, please refer to the corresponding command.*

#### Read Command

None

#### Test Command

None

### 2.16.4 Parameters and Defined Values

Parameter	Value	Description
<hardware_revision>		<b>Hardware Revision Number Parameter</b>
	2 ASCII hex digits	The TD1202 module hardware revision number
<firmware_revision>		<b>Firmware Revision Number Parameter</b>
	"SOFTxxxx"	The TD1202 module firmware revision number, with "x" being an ASCII-coded digit
<serial_number>		<b>Serial Number Parameter</b>
	8 ASCII hex digits	The TD1202 module serial number
<valx>		<b>Active Profile Parameters</b>
		For each "<valx>" parameter value, please refer to the corresponding command

### 2.16.5 Parameter Storage

None

## 2.16.6 Examples

Command	Responses
<b>AT&amp;V</b> <i>Note: Display active parameters in RAM</i>	Telecom Design TD1202 Hardware Version: 0F Software Version: SOFT1100 S/N: 00001058 ACTIVE PROFILE E1 V1 Q0 X1 S200:0 S300:24 S301:2 S302:14 S303:1 OK <i>Note: Done for echo.</i>

2.17 Save Configuration &W

2.17.1 Description

This command writes the active configuration into a non volatile memory (Flash).

2.17.2 Compatibility

This command is available in all firmware revisions since SOFT1104.

2.17.3 Syntax

Action Command

AT&W

OK

Read Command

None

Test Command

None

2.17.4 Parameters and Defined Values

None

2.17.5 Parameter Storage

None

2.17.6 Examples

Command	Responses
ATS302=10 <i>Note: change transmission power in volatile memory only</i>	OK
AT&W	OK
ATZ <i>Note: reset the TD1202 module</i>	OK
ATS302?	10 OK <i>Note: Default value set back from Flash memory</i>



## 3 TD1202 Hayes-Compatible AT LAN Command Set Reference

This section provides a reference for the Hayes-compatible “AT” LAN commands implemented by the TD1202 Module.

**IMPORTANT NOTICE:** Because of the required replacement of SIGFOX protocol V0 by SIGFOX protocol V1, from firmware version SOFT1278 and above, it is no longer possible to fit the AT LAN Command Set into the TD1202 Flash memory. If your application requires this AT LAN Command Set, you should use the footprint-compatible TD1208 module instead.

### 3.1 Introduction to the TD LAN

The TD LAN is an energy-efficient local network that enables devices to communicate low-bandwidth data (up to 17 bytes per second) over the air.

In order to save power, the TD LAN operates as a time division duplexing (TDD) network where the device alternately transmits and receives data packets over the same radio channel. And although not fixed by the firmware and that these roles may be swapped at any time, it is assumed that one single device is in permanent receive mode, while the other devices are in transmit upon request mode, resulting in a non symmetric, star-topology network.

#### 3.1.1 RF Parameters

The TD LAN network operates over the license-free ISM 869 MHz radio band (868.0 to 869.7 MHz). The devices use a single 25 kHz narrow-band channel to transmit data @ 9600bps using a GFSK modulation.

The operating frequency can be set using an AT command for a particular application.

The transmit power can also be adjusted using an AT command, in order to reduce the transmit power consumption or increase the radio range when required.

#### 3.1.2 Network Addressing

Each device is assigned a logic 24-bit logic address and a corresponding 24-bit address mask. The logic address is transmitted into the RF frames that are sent and matched by the receiver after applying the address mask to it. The default null address and full address mask (all bits set to 1) ensure that receiving is enabled by default.

However, the recommended setup is to have a full address mask (all bits set to 1) for the transmitter and a partial address mask (not all bits are set to 1) for the receiver, both devices having a common address field (i.e. same “subnet”) over the partial address mask bits. This addressing scheme provides a way for transmitters to access the receiver when they are using the same “subnet”, and for the receiver a way to acknowledge a particular frame has been received to the original transmitter only.

Here is an example for a 4-bit subnet mask:

- Receiver: Address 0x5ED709, Mask 0xF00000

Address	0	1	0	1	1	1	1	0	1	1	0	1	0	1	1	1	0	0	0	0	1	0	0	1
Mask	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

- Transmitter: Address 0x55C344, Mask 0xFFFFF

Address	0	1	0	1	0	1	0	1	1	1	0	0	0	0	1	1	0	1	0	0	0	1	0	0
Mask	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

### 3.1.3 Time Windowing

Having a device in permanent receive mode would put a heavy constraint on its average power consumption. Thus, a time windowing method is used where a device in receive mode is only listening to incoming RF traffic the time required to capture a frame, then the radio is turned off for a fixed 1 second period, achieving a power reduction that is proportional to this duty cycle.

On the other hand, when in transmit mode, a device has to send radio frames for a duration that is at least equal to the receiver's time window plus 2 frame receive time in order to be seen by the receiver.

### 3.1.4 Data Transmission

From 1 to 17 data bytes may be transmitted into a single message frame, which is acknowledged by the receiver. If this acknowledgement is not received within 2 seconds, the message frame is resent up to 2 times before giving up.

The data bytes to transmit must be given using 2 hexadecimal digits per byte; each byte may be separated by space or tab characters for clarity.

Upon reception, the data bytes are provided using the same 2 hexadecimal digit per byte format. By default, no space character is inserted between data bytes, but an AT command allows to specify if space separators must be inserted between every byte, word or long word.

## 3.2 TD LAN Address S400

### 3.2.1 Description

This command configures or queries the TD LAN address.

### 3.2.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.2.3 Syntax

*Action Command*

**ATS400=[<address>]**

OK

*Read Command*

**ATS400?**

<address>

OK

*Test Command*

**ATS400=?**

0..16777215 or 0x000000..0xFFFFFF

OK

### 3.2.4 Parameters and Defined Values

Parameter	Value	Description
<address>		<b>TD LAN Address Parameter</b>
	0..16777215 or 0x000000..0xFFFFFF	TD LAN 24-bit address (default value is 0)

### 3.2.5 Parameter Storage

The <address> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 3.2.6 Examples

Command	Responses
<b>ATS400?</b>	000000 OK <i>Note: TD LAN address is 0</i>
<b>ATS400=1</b> <i>Note: set the TD LAN address to 1</i>	OK <i>Note: Command valid</i>

## 3.3 TD LAN Mask S401

### 3.3.1 Description

This command configures or queries the TD LAN address mask.

### 3.3.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.3.3 Syntax

*Action Command*

**ATS401=[<mask>]**

OK

*Read Command*

**ATS401?**

<mask>

OK

*Test Command*

**ATS401=?**

0..16777215 or 0x000000..0xFFFFFFFF

OK

### 3.3.4 Parameters and Defined Values

Parameter	Value	Description
<mask>		<b>TD LAN Address Mask Parameter</b>
	0..16777215 or 0x000000..0xFFFFFFFF	TD LAN 24-bit address mask (default value is 0xFFFFFFFF)

### 3.3.5 Parameter Storage

The <mask> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 3.3.6 Examples

Command	Responses
<b>ATS401?</b>	<b>FFFFFF</b> OK <i>Note: TD LAN address mask is 0xFFFFFFFF</i>
<b>ATS401=0x7FFFFFFF</b> <i>Note: set the TD LAN address mask to 0x7FFFFFFF</i>	OK <i>Note: Command valid</i>

## 3.4 TD LAN Separator S402

### 3.4.1 Description

This command configures or queries the TD LAN separator value.

### 3.4.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.4.3 Syntax

*Action Command*

**ATS402=[<separator\_count>]**

OK

*Read Command*

**ATS402?**

<separator\_count>

OK

*Test Command*

**ATS402=?**

0,1,2,4

OK

### 3.4.4 Parameters and Defined Values

Parameter	Value	Description
<separator_count>		<b>TD LAN Separator Count Parameter</b>
	0	No space separator between received TD LAN data bytes (default value)
	1	A space separator is inserted between each received TD LAN data bytes
	2	A space separator is inserted between each received TD LAN data words
	4	A space separator is inserted between each received TD LAN data long words

### 3.4.5 Parameter Storage

The <separator\_count> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

# TD1202 RM

## 3.4.6 Examples

Command	Responses
ATS402?	0 OK <i>Note: no space separator between TD LAN data bytes</i>
ATS402=1 <i>Note: add a space character between every TD LAN data bytes</i>	OK <i>Note: Command valid</i>

## 3.5 TD LAN Frequency S403

### 3.5.1 Description

This command configures or queries the TD LAN frequency value.

### 3.5.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.5.3 Syntax

*Action Command*

**ATS403=[<frequency>]**

OK

*Read Command*

**ATS403?**

<frequency>

OK

*Test Command*

**ATS403=?**

868000000..869700000 or 868.000000..869.700000

OK

### 3.5.4 Parameters and Defined Values

Parameter	Value	Description
<frequency>		<b>TD LAN Separator Count Parameter</b>
	868000000..869700000 or 868.000000..869.700000	Frequency in Hz or MHz (default value 869312500)

### 3.5.5 Parameter Storage

The <frequency> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 3.5.6 Examples

Command	Responses
<b>ATS403?</b>	869312500 OK <i>Note: the TD LAN frequency is 869.3125 MHz</i>
<b>ATS403=869.3625</b> <i>Note: the TD LAN frequency is set to 869.3625 MHz</i>	OK <i>Note: Command valid</i>

## 3.6 TD LAN Power S404

### 3.6.1 Description

This command configures or queries the TD LAN RF power level in dBm.

### 3.6.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.6.3 Syntax

*Action Command*

**ATS404=[<dbm\_level>]**

OK

*Read Command*

**ATS404?**

<dbm\_level>

OK

*Test Command*

**ATS404=?**

-35..14

OK

### 3.6.4 Parameters and Defined Values

Parameter	Value	Description
<dbm_level>		<b>TD LAN Power Parameter</b>
	-35..14	TD LAN RF power level in dBm (default value is 14)

### 3.6.5 Parameter Storage

The <dbm\_level> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 3.6.6 Examples

Command	Responses
<b>ATS404?</b>	14 OK <i>Note: TD LAN power is 14 dBm</i>
<b>ATS404=10</b> <i>Note: set the TD LAN RF power to 10 dBm</i>	OK <i>Note: Command valid</i>



## 3.7 TD LAN Receive RF Message \$RL

### 3.7.1 Description

This command receives a TD LAN RF message or sets the TD LAN receive mode.

### 3.7.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.7.3 Syntax

*Action Command*

**AT\$RL=<mode>[,<timeout>]**

OK

*Read Command*

None

*Test Command*

None

### 3.7.4 Parameters and Defined Values

Parameter	Value	Description
<mode>		<b>TD LAN Receive RF Message Mode Parameter</b>
	0	Disable TD LAN RF receive mode (default value)
	1	Enable TD LAN RF synchronous single-frame receive mode, either infinite or until a timeout value is reached
	2	Enable TD LAN RF asynchronous multi-frame receive mode
<timeout>	0..4294967295	Only valid when <mode> is equal to 1: provides the timeout value in seconds to wait before aborting receive mode

### 3.7.5 Parameter Storage

None

### 3.7.6 Examples

Command	Responses
<b>AT\$RL=0</b> <i>Note: disable TD LAN RF receive mode</i>	OK <i>Note: Command valid</i>
<b>AT\$RL=1</b> <i>Note: enable TD LAN RF synchronous receive mode</i>	0102030405060708090a0b0c0d0e0f1011 OK <i>Note: Response is only obtained after receiving a valid TD LAN frame</i>
<b>AT\$RL=1,5</b> <i>Note: enable TD LAN RF synchronous receive mode for 5s maximum</i>	0102030405060708090a0b0c0d0e0f1011 OK <i>Note: Response is only obtained after receiving a valid TD LAN frame</i>
<b>AT\$RL=1,5</b> <i>Note: enable TD LAN RF synchronous receive mode for 5 seconds maximum</i>	ERROR <i>Note: No valid TD LAN frame received within 5 seconds</i>

## TD1202 RM

<b>AT\$RL=2</b> <i>Note: enable TD LAN RF asynchronous receive mode</i>	<b>OK</b> <i>Note: Command valid</i>
	<b>+RX_LAN=0102030405060708090a0b0c0d0e0f1011</b> <i>Note: Unsolicited response is obtained after receiving a valid TD LAN frame</i>
	<b>+RX_LAN=f1f2f3f4f5f6f7f8f9fafbfcfdfefff0001</b> <i>Note: Unsolicited response is obtained after receiving a valid TD LAN frame</i>

### 3.7.7 Notes

The message length is always 17 byte long.

## 3.8 TD LAN Send RF Message \$SL

### 3.8.1 Description

This command sends a TD LAN RF message.

### 3.8.2 Compatibility

This command is available in all firmware revisions since SOFT1134.

### 3.8.3 Syntax

*Action Command*

**AT\$SL=[<hex\_byte1>[[ ]..**<hex\_byte2>**]]..]**

OK

*Note: There can be from 1 to 17 **<hex\_byte>** parameter values, optionally separated by single or multiple space or tabulation characters.*

*Read Command*

None

*Test Command*

None

### 3.8.4 Parameters and Defined Values

Parameter	Value	Description
<b>&lt;hex_byte&gt;</b>		<b>TD LAN Send RF Message Parameter</b>
	00..FF	2-digit hexadecimal byte value ('0' to '9', 'a' to 'f' and 'A' to 'F' characters are valid) There can be from 1 to 17 <b>&lt;hex_byte&gt;</b> parameter values, optionally separated by single or multiple space (IA5 2/0) or tabulation (IA5 0/9) characters

### 3.8.5 Parameter Storage

None

### 3.8.6 Examples

Command	Responses
<b>AT\$SL=01</b> <i>Note: send a single-byte TD LAN RF frame</i>	OK <i>Note: Frame sent and receive acknowledgment received</i>
<b>AT\$SL=0102030405060708090A0B0C0D0E0F1011</b> <i>Note: send a 17-byte long TD LAN RF frame</i>	OK <i>Note: Frame sent and receive acknowledgment received</i>
<b>AT\$SL=0102030405060708090A0B0C0D0E0F1011</b> <i>Note: send a TD LAN RF frame</i>	ERROR <i>Note: No frame sent or no receive acknowledgment received</i>

### 3.8.7 Notes

The actual length of the sent message is always 17-byte long.

## 4 TD1202 Hayes-Compatible AT Sensor Command Set Reference

This section provides a reference for the Hayes-compatible “AT” Sensor commands implemented by the TD1202 Module.

**IMPORTANT NOTICE:** Because of the required replacement of SIGFOX protocol V0 by SIGFOX protocol V1, from firmware version SOFT1278 and above, it is no longer possible to fit the AT Sensor Command Set into the TD1202 Flash memory. If your application requires this AT Sensor Command Set, you should use the footprint-compatible TD1208 module instead.

### 4.1 Introduction to Sensor

Sensor is a way to automatically send and process most relevant information from your TD1202 to a Web Service without having to worry about hardware monitoring, data transmission and message decoding.

Any TD1202 Module can monitor information concerning its battery level, its temperature level and state of any connected switch. Moreover in the case of a LAN, information concerning connection status and signal level can also be monitored.

Sensor is providing data interpretation and action triggering for a whole set of pre-defined messages. Thus each message can, according to its type and associated payload, trigger an action that can either be pre-defined or configured online via the Sensor Web Interface. Message types are the following.

1. **Register:** this message allows the module to register on Sensor in order to enable proper data processing. This message should only be sent once.
2. **Event:** this message is sent when a specific event occurs. All event messages are monitoring related. Following events can be emitted:
  - Boot
  - Battery status (low, ok)
  - Temperature status (low, ok, high)
  - Switch status (on, off)
  - Connection status (lost,ok)
  - Signal level status (low,ok)

Note that all these events are being sent automatically if their respective monitoring is enabled.

3. **Data:** this message is used to carry data information such as sensor measurements, phone numbers, passwords, etc...
4. **Service:** this message is used to trigger a pre-defined action on Sensor. Already defined services are:
  - Sending a SMS (up to 9 ASCII characters). SMS will be sent to phone numbers registered with the corresponding Data frame or via the Sensor Web Interface.
  - Tweeting a message (up to 9 ASCII characters). Tweets will be sent to a previously defined Twitter account via the Sensor Web Interface.
5. **Keep-Alive:** this message is sent to confirm the Module activity to Sensor.
6. **Raw:** this message can be used to send raw information to Sensor. Up to 10 bytes can be sent.

In order to look at all the received frames please go to the Sensor Web Interface: <https://developers.insgroup.fr/>

## 4.2 Sensor and LAN

### 4.2.1 Introduction

Alongside being able to send data over the Sigfox Network, the TD1202 Module is also capable of short range radio communication. Therefore modules can communicate between each other and become a LAN.

Sensor LAN is organized in a Gateway which manages a set of Devices. Up to 15 Devices can be connected to a Gateway. Thus a Gateway represents a network with one specific address. Any Device willing to join this network must first register to the Gateway. If the Gateway is not already managing 15 Devices, it will issue an address to the Device.

Devices can only register successfully if the Gateway has previously being set into registration mode. Note that as long as registration is opened on the Gateway side, already registered Devices can NOT communicate with the Gateway anymore. Once registration of the Devices is done, the Gateway should therefore close the registration for normal activity to restart.

Once a Device has obtained an address it can only communicate with the Gateway it had registered on. In a LAN, only a Gateway can send Sensor messages. If a Device has to send a message to Sensor it must send it to the Gateway which will forward it to Sensor.

Having a LAN allows to monitor Device connection and signal level status. In order to perform connection monitoring, a local device will send a local keep-alive message on a customizable interval. The Gateway will then make sure the Device has sent the keep-alive information on time.

Eventually a Device might want the Gateway to process some kind of information and reply back. Communication between a Device and its Gateway can be achieved by using the LAN Data command. Default behaviour for the Gateway is to echo the data back.

### 4.2.2 Collision Handling

One must be aware that a Gateway can receive a message from only one Device at a time. Furthermore, the Gateway cannot receive any message at all when emitting in Sigfox (up to 6 seconds). This can lead to collision issues when Devices try to communicate with the Gateway.

Concerning Event frames, a collision handler will ensure data delivery by repeating the frame until acked by the Gateway.

Concerning other frame types, no collision handling is provided and the user should always checked wether a frame has been received or not.

## 4.3 TD1202 Module Type

Depending on your application requirements, three behaviours can be associated to your module. These last are called Module type and can be one of the following:

- Transmitter: a transmitter is a standalone module. That is to say it can send messages to the Sigfox network on its own. This is the default behaviour for a TD1202 module.
- Device: a device is a module that can not directly send messages to Sensor. It must rely on a Gateway which will forward all its messages.

# TD1202 RM

- Gateway: a gateway is a transmitter that can also handle up to 15 Devices by forwarding their messages to Sensor and monitoring individually their connection and signal level status.

## 4.4 Examples

### 4.4.1 Transmitter with battery level, temperature and boot monitoring

<b>AT\$REG</b> OK	Note: Register the Transmitter on Sensor
<b>ATS502=1,2400,2900</b> OK	Note: Enable battery monitoring with a 2.4V low level and 2.9 OK level
<b>ATS503=1,3600,-5,35</b> OK	Note: Enable temperature monitoring every hours with a -5°C low level and a 35°C high level.
<b>ATS507=1,</b> OK	Note: Enable boot monitoring
<b>AT&amp;W</b> OK	Note: Save configuration in flash

### 4.4.2 Gateway with boot monitoring

<b>ATS500=1</b> OK	Note: Define Module Type as Gateway
<b>AT&amp;w</b> OK	Note: Save configuration into Flash memory.
<b>ATZ</b> OK	Note: Reboot the Module.
<b>AT\$REG</b> OK	Note: Register the Gateway on Sensor
<b>ATS507=1,</b> OK	Note: Enable boot monitoring
<b>AT&amp;w</b> OK	Note: Save configuration into Flash memory.

## 4.4.3 Device Registration on a Gateway with connection monitoring

Device:

**ATS500=0** Note: Define Module Type as Device  
OK

**AT&w** Note: Save Module type into Flash memory.  
OK

**ATZ** Note: Reboot the Module.  
OK

Gateway:

**AT\$LR=1** Note: Open registration on the Gateway.  
OK

Device:

**AT\$LR** Note: Device registration on the gateway  
OK

Gateway:

**AT\$LR=0** Note: Close registration on the Gateway.  
OK

Device:

**AT\$REG** Note: Register the Device on Sensor  
OK

**ATS505=1,300** Note: Enable connection monitoring with a  
OK 5mn checking interval.

**AT&w** Note: Save configuration into Flash memory.  
OK

## 4.5 TD SENSOR – Module Type S500

### 4.5.1 Description

This command configures or queries the TD SENSOR module type.

### 4.5.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.5.3 Syntax

*Action Command*

**ATS500=<type>**

OK

**AT&W**

OK

**ATZ**

OK

*Read Command*

**ATS500?**

<type>

OK

*Test Command*

**ATS500=?**

Device: 0, Gateway: 1, Transmitter: 2

OK

### 4.5.4 Parameters and Defined Values

Parameter	Value	Description
<type>		<b>TD SENSOR Type Parameter</b>
	0	Change the module behaviour to Device
	1	Change the module behaviour to Gateway
	2	Change the module behaviour to Transmitter (default value)

### 4.5.5 Parameter Storage

The <type> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 4.5.6 Examples

Command	Responses
<b>ATS500?</b>	0



	OK <i>Note: the Module is a Device</i>
<b>ATS500=1</b> <b>AT&amp;W</b> <b>ATZ</b> <i>Note: Change the Module behaviour to Gateway</i>	OK OK OK  <i>Note: Commands valid</i>

## 4.5.7 Notes

Changing the Module Type requires to save the new configuration to Flash and to reboot the Module to be effective. Therefore the following set of AT commands should always be executed when a Module Type has to be modified.

*Full List of Commands to properly update the Module Type*

**ATS500=<type>**

OK

**AT&w**

OK

**ATz**

OK

## 4.6 TD SENSOR – Device Class S501

### 4.6.1 Description

This command configures or queries the TD SENSOR Device Class. It allows differentiating several kinds of Modules.

### 4.6.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.6.3 Syntax

*Action Command*

**ATS501=<class>**

OK

*Read Command*

**ATS501?**

<class>

OK

*Test Command*

**ATS501=?**

0..65535

OK

### 4.6.4 Parameters and Defined Values

Parameter	Value	Description
<class>		<b>TD SENSOR Class Parameter</b>
	0..65535	Change the Device Class

### 4.6.5 Parameter Storage

The <class> parameter value is stored in Flash memory using the **AT&W** command. The default value can be restored using the **AT&F** command.

### 4.6.6 Examples

Command	Responses
<b>ATS501=0x1234</b> <i>Note: Change the Device Class to 0x1234</i>	OK <i>Note: Command valid</i>

## 4.7 TD SENSOR – Battery Monitoring S502

### 4.7.1 Description

This command configures or queries the TD SENSOR battery monitoring status.

### 4.7.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.7.3 Syntax

#### Action Command

```
ATS502=<enable>[,<level_low>,<level_ok>]
OK
```

#### Read Command

```
ATS502?
<enable>,<level_low>,<level_ok>
OK
```

#### Test Command

```
ATS502=?
0..1, 2100..3300, 2100..3300
OK
```

### 4.7.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Battery Monitoring Enable Parameter</b>
	0	Disable Battery Monitoring (default value)
	1	Enable Battery Monitoring
<level_low>		<b>TD SENSOR Battery Monitoring Level Low Parameter</b>
	2100..3300	Battery Low Level in millivolts. If the Battery level falls below the specified level, a Battery Low Event will be emitted.
<level_ok>		<b>TD SENSOR Battery Monitoring Level OK Parameter</b>
	2100..3300	Battery OK Level in millivolts. If the Battery level rises above the specified level, a Battery OK Event will be emitted. Note that a Battery OK Event will only be emitted if a Battery Low Event occurred previously.

### 4.7.5 Parameter Storage

All Battery Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.7.6 Examples

Command	Responses
<b>ATS502=1,2300,2800</b> <i>Note: enable battery monitoring. A Battery Low event will be emitted if the battery level falls below 2.3V and a battery OK will be emitted if, following the battery Low</i>	<b>OK</b> <i>Note: Command valid</i>

## TD1202 RM

<i>Event, the battery level rises above 2.8V.</i>	
<b>ATS502=0</b>	<b>OK</b>
<i>Note: Disable battery monitoring</i>	<i>Note: Command valid</i>

## 4.8 TD SENSOR – Temperature Monitoring S503

### 4.8.1 Description

This command configures or queries the TD SENSOR temperature monitoring status.

### 4.8.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.8.3 Syntax

#### Action Command

```
ATS503=<enable>[,<interval>,<level_low>,<level_high>]
OK
```

#### Read Command

```
ATS503?
<enable>,<interval>,<level_low>,<level_high>
OK
```

#### Test Command

```
ATS503=?
0..1, 0..4294967295, -30..85, -30.85
OK
```

### 4.8.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Temperature Monitoring Enable Parameter</b>
	0	Disable Temperature Monitoring (default value)
	1	Enable Temperature Monitoring
<interval>		<b>TD SENSOR Temperature Monitoring Interval Parameter</b>
	0..4294967295	Interval in seconds at which the Temperature should be checked.
<level_low>		<b>TD SENSOR Temperature Monitoring Level Low Parameter</b>
	-30..85	Temperature Low Level in Celsius degrees. If the Temperature level falls below the specified level, a Temperature Low Event will be emitted.
<level_high>		<b>TD SENSOR Temperature Monitoring Level High Parameter</b>
	-30..85	Temperature High Level in Celsius degrees. If the Temperature level rises above the specified level, a Temperature High Event will be emitted.

### 4.8.5 Parameter Storage

All Temperature Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.8.6 Examples

Command	Responses
ATS503=1,3600,-5,35 <i>Note: enable temperature monitoring by checking temperature level every hour. A Temperature Low event</i>	OK <i>Note: Command valid</i>

## TD1202 RM

<i>will be emitted if the temperature level falls below -5°C, a temperature High Event will be emitted if the temperature level rises above 35°C and following any of the two previous event if the temperature level goes back into the authorized range a Temperature OK Event will be emitted.</i>	
<b>ATS503=0</b> <i>Note: Disable temperature monitoring</i>	<b>OK</b> <i>Note: Command valid</i>

## 4.9 TD SENSOR – RSSI Monitoring S504

### 4.9.1 Description

This command configures or queries the TD SENSOR RSSI monitoring status. Module Type must be set to Device and keep-alive monitoring must be enabled to use this functionality.

### 4.9.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.9.3 Syntax

*Action Command*

**ATS504=<enable>[,<level\_low>,<level\_ok>]**

OK

*Read Command*

**ATS504?**

<enable>,<level\_low>,<level\_ok>

OK

*Test Command*

**ATS504=?**

0..1, -122..14, -122..14

OK

### 4.9.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR RSSI Monitoring Enable Parameter</b>
	0	Disable RSSI Monitoring (default value)
	1	Enable RSSI Monitoring
<level_low>		<b>TD SENSOR RSSI Monitoring Level Low Parameter</b>
	-122..14	RSSI Low Level in dBm. If the RSSI level falls below the specified level, a RSSI Low Event will be emitted.
<level_ok>		<b>TD SENSOR RSSI Monitoring Level High Parameter</b>
	-122..14	RSSI OK Level in dBm. If the RSSI level rises above the specified level, a RSSI OK Event will be emitted.

### 4.9.5 Parameter Storage

All RSSI Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.9.6 Examples

Command	Responses
<b>ATS504=1,-50,-10</b> <i>Note: enable RSSI monitoring. A RSSI Low event will be emitted if the RSSI level falls below -50dBm and if RSSI level then rises back above -10dBm a RSSI OK event</i>	OK <i>Note: Command valid</i>

## TD1202 RM

<i>will be emitted.</i>	
<b>ATS504=0</b>	<b>OK</b>
<i>Note: Disable RSSI monitoring</i>	<i>Note: Command valid</i>

### 4.9.7 Notes

RSSI monitoring requires the device to communicate with a Gateway. Therefore a Gateway or a Transmitter cannot monitor its own RSSI status. Hence to activate RSSI monitoring the Module type must be set to Device. Moreover RSSI status is only checked when a Gateway receives a message from a Device. Checking RSSI level at regular interval requires then to communicate with the Gateway at the very same interval (whichever the message type is). We therefore also require having Keep-Alive monitoring enabled in order to monitor RSSI.



## 4.10 TD SENSOR – Connection Monitoring S505

### 4.10.1 Description

This command configures or queries the TD SENSOR Connection monitoring status. Module Type must be set to Device to use this functionality.

### 4.10.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.10.3 Syntax

#### Action Command

**ATS505=<enable>[,<interval>]**

OK

#### Read Command

**ATS505?**

<enable>,<interval>

OK

#### Test Command

**ATS505=?**

0..1, 10..4294967295

OK

### 4.10.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Connection Monitoring Enable Parameter</b>
	0	Disable Connection Monitoring (default value)
	1	Enable Connection Monitoring
<interval>		<b>TD SENSOR Connection Monitoring Interval Parameter</b>
	10.. 4294967295	Connection checking interval in seconds. Due to transmission duration and risks of collisions the minimum checking interval is 10s. Minimum recommended interval when monitoring connection for several devices is 300 seconds.

### 4.10.5 Parameter Storage

All Connection Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.10.6 Examples

Command	Responses
<b>ATS505=1,300</b> <i>Note: enable Connection monitoring and check every 5 minutes if the connection is still alive. If it is not the case a Connection Lost event will be emitted by the gateway.</i>	<b>OK</b> <i>Note: Command valid</i>

## TD1202 RM

A Connection OK event will be emitted by the gateway if, following a Connection Lost event, a connection can be established.	
<b>ATS505=0</b> Note: Disable Connection monitoring	<b>OK</b> Note: Command valid

### 4.10.7 Notes

Connection monitoring requires the device to communicate with a Gateway. Therefore a Gateway or a Transmitter cannot monitor its own connection status. Hence to activate connection monitoring the Module type must be set to Device.

## 4.11 TD SENSOR – Switch Monitoring S506

### 4.11.1 Description

This command configures or queries the TD SENSOR Switch monitoring status.

### 4.11.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.11.3 Syntax

#### Action Command

```
ATS506=<enable>,<port>,<bit>[,<falling>,<rising>,<pull>,<state>]
OK
```

#### Read Command

```
ATS506?
<port>,<bit>,<falling>,<rising>,<pull>,<state>

..

<port>,<bit>,<falling>,<rising>,<pull>,<state>

OK

Note: return information only about enabled switch
```

#### Test Command

```
ATS506=?
0..1, 0..5, 0..15, 0..1, 0..1, 0..1, 0..1
OK
```

### 4.11.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Switch Monitoring Enable Parameter</b>
	0	Disable Switch Monitoring (default value)
	1	Enable Switch Monitoring
<port>		<b>TD SENSOR Switch Monitoring Port Parameter</b>
	0..5	Port on which the switch is connected.
<bit>		<b>TD SENSOR Switch Monitoring Bit Parameter</b>
	0..15	Bit on which the switch is connected.
<falling>		<b>TD SENSOR Switch Monitoring Falling Parameter</b>
	0	Disable sending Switch ON Message on falling edge
	1	Enable sending Switch ON Message on falling edge
<rising>		<b>TD SENSOR Switch Monitoring Rising Parameter</b>
	0	Disable sending Switch OFF Message on rising edge
	1	Enable sending Switch OFF Message on rising edge
<pull>		<b>TD SENSOR Switch Monitoring Pull Parameter</b>
	0	Disable pull-up on GPIO.

# TD1202 RM

	1	Enable pull-up on GPIO.
<state>		<b>TD SENSOR Switch Monitoring State Parameter</b>
	0	If pull is set to 1, pull the GPIO down.
	1	If pull is set to 1, pull the GPIO up.

## 4.11.5 Parameter Storage

All Switch Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

## 4.11.6 Examples

Command	Responses
<b>ATS506=1,2,15,1,1,1,1</b> <i>Note: enable Switch monitoring on port 2 bit 15 (USR1) for both edges with a pull-up. On a rising edge a Switch Off event will be emitted. On a falling edge a switch On event will be emitted.</i>	<b>OK</b> <i>Note: Command valid</i>
<b>ATS504=0,2,15</b> <i>Note: disable Switch monitoring on port 2 bit 15 (USR1)</i>	<b>OK</b> <i>Note: Command valid</i>

## 4.11.7 Notes

In order to receive Switch On or Switch Off event messages that corresponds to the reality, all switches must be connected to GND and have the GPIO pull-up enabled. Available GPIOs for switch monitoring and corresponding port and bit values can be found in the following table:

Pin Name	Port	Bit
SDA	0	0
SCL	0	1
USR0	1	13
DAC0	1	11
USR2	2	0
USR3	2	1
USR4	2	14
USR1	2	15
ADC0	3	6

## 4.12 TD SENSOR – Boot Monitoring S507

### 4.12.1 Description

This command configures or queries the TD SENSOR boot monitoring status.

### 4.12.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.12.3 Syntax

*Action Command*

**ATS507=<enable>**

OK

*Read Command*

**ATS507?**

<enable>

OK

*Test Command*

**ATS507=?**

0..1

OK

### 4.12.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Boot Monitoring Enable Parameter</b>
	0	Disable Boot Monitoring (default value)
	1	Enable Boot Monitoring

### 4.12.5 Parameter Storage

All Boot Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.12.6 Examples

Command	Responses
<b>ATS507=1</b> <i>Note: enable Boot monitoring. A Boot event will be sent each time the Module boots.</i>	OK <i>Note: Command valid</i>
<b>ATS507=0</b> <i>Note: Disable Boot monitoring</i>	OK <i>Note: Command valid</i>

### 4.12.7 Notes

To be effective this command should always be followed by the **AT&W** command.

## 4.13 TD SENSOR – Keep-Alive Monitoring S508

### 4.13.1 Description

This command configures or queries the TD SENSOR Keep-Alive monitoring status.

### 4.13.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.13.3 Syntax

#### Action Command

```
ATS508=<enable>[,<interval>]  
OK
```

#### Read Command

```
ATS508?  
<enable>,<interval>  
OK
```

#### Test Command

```
ATS508=?  
0..1, 1..255  
OK
```

### 4.13.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>TD SENSOR Keep-Alive Monitoring Enable Parameter</b>
	0	Disable Keep-Alive Monitoring (default value)
	1	Enable Keep-Alive Monitoring
<interval>		<b>TD SENSOR Keep-Alive Monitoring Interval Parameter</b>
	1..255	Keep-Alive monitoring interval in hours. A Keep-alive Sensor frame will be sent every interval if enable is set to 1.

### 4.13.5 Parameter Storage

All Keep-Alive Monitoring parameters values can be stored in Flash memory using the **AT&W** command. The default values can be restored using the **AT&F** command.

### 4.13.6 Examples

Command	Responses
<b>ATS508=1,24</b> <i>Note: enable Keep-alive monitoring. A Keep-Alive frame will be sent every 24hours. Note that information about battery temperature level will be sent with the frame.</i>	OK <i>Note: Command valid</i>
<b>ATS508=0</b> <i>Note: Disable Keep-Alive monitoring</i>	OK <i>Note: Command valid</i>

## 4.14 TD SENSOR LAN – Send Data \$LD=

### 4.14.1 Description

This command sends a Data Message to the Gateway. Module type must be set to Device and a Local Register must have previously been issued for this command to be successful.

### 4.14.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.14.3 Syntax

Action Command

**AT\$LD=<hex\_byte1> [[ ]..<hex\_byte2>..]**

RX: <hex\_byte1> ... <hex\_byte15>

OK

*Note: There can be from 1 to 16 <hex\_byte> parameter values, optionally separated by single or multiple space or tabulation characters.*

Read Command

None

Test Command

None

### 4.14.4 Parameters and Defined Values

Parameter	Value	Description
<hex_byte>		<b>SENSOR LAN Send Data Value Parameter</b>
	00..FF	2-digit hexadecimal byte value ('0' to '9', 'a' to 'f' and 'A' to 'F' characters are valid) There can be from 1 to 16 <hex_byte> parameter values, optionally separated by single or multiple space (IA5 2/0) or tabulation (IA5 0/9) characters

### 4.14.5 Parameter Storage

None

### 4.14.6 Examples

Command	Responses
<b>AT\$LD=01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10</b> <i>Note: sends 16 bytes to the gateway. Note that whichever is the length of your data, 16 bytes will always be sent. Also note that default behavior for the Gateway is to echo the message back. Please be aware that due to acknowledgment protocol, the gateway's reply will only contain 15 bytes thus removing the last byte from the original message.</i>	<b>RX: 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F</b> <b>OK</b>

## 4.15 TD SENSOR LAN – Open Registration \$LR=

### 4.15.1 Description

This command enables or disables Devices registration on the Gateway. Module type must be set to Gateway for this command to be issued. A maximum of 15 Devices can register on a unique Gateway.

### 4.15.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.15.3 Syntax

*Action Command*

**AT\$LR=<enable>**

OK

*Read Command*

None

*Test Command*

None

### 4.15.4 Parameters and Defined Values

Parameter	Value	Description
<enable>		<b>SENSOR LAN Open Registration Enable Parameter</b>
	0	Devices Registration disabled (default value)
	1	Devices Registration enabled

### 4.15.5 Parameter Storage

None

### 4.15.6 Examples

Command	Responses
<b>AT\$LR=1</b> <i>Note: Devices are allowed to register on the Gateway.</i>	OK
<b>AT\$LR=0</b> <i>Note: Devices are not allowed to register on the Gateway.</i>	OK



## 4.16 TD SENSOR LAN – Registration \$LR

### 4.16.1 Description

This command allows a Device to register on the Gateway. Module type must be set to Device for this command to be issued. Any Device must first register to the Gateway to obtain a LAN address and be able to send messages to Sensor.

### 4.16.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.16.3 Syntax

*Action Command*

**AT\$LR**

OK

*Read Command*

None

*Test Command*

None

### 4.16.4 Parameters and Defined Values

None

### 4.16.5 Parameter Storage

None

### 4.16.6 Examples

Command	Responses
<b>AT\$LR</b> <i>Note: Try to Register on the Gateway</i>	OK <b>Note : Registration OK</b>

4.17 TD SENSOR LAN – Get Address \$LA?

4.17.1 Description

This command displays current address and mask values for Sensor LAN.

4.17.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

4.17.3 Syntax

Action Command
<b>AT\$LA?</b> Address: 5B2D3F, Mask: FFFFFFFF OK

Read Command
None

Test Command
None

4.17.4 Parameters and Defined Values

None

4.17.5 Parameter Storage

None

4.17.6 Examples

Command	Responses
<b>AT\$LA?</b> <i>Note: Request LAN address and mask for Sensor.</i>	<b>Address: 5B2D3F, Mask: FFFFFFFF</b> OK

## 4.18 TD SENSOR LAN – LAN Reset \$LZ

### 4.18.1 Description

If the Module type is set to Device, this command resets the current LAN address. If the Module type is set to Gateway, this commands reset the list of Devices connected to the gateway.

### 4.18.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.18.3 Syntax

*Action Command*

**AT\$ LZ**

OK

*Read Command*

None

*Test Command*

None

### 4.18.4 Parameters and Defined Values

None

### 4.18.5 Parameter Storage

None

### 4.18.6 Examples

Command	Responses
<b>AT\$ LZ</b> <i>Note: Reset LAN configuration depending on Module type.</i>	OK

## 4.19 TD SENSOR SEND - Data Phone \$DP

### 4.19.1 Description

This command sends a Sensor message containing a phone number. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received.

### 4.19.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.19.3 Syntax

*Action Command*

**AT\$DP=<index>,<cell\_phone>**

OK

*Read Command*

None

*Test Command*

None

### 4.19.4 Parameters and Defined Values

Parameter	Value	Description
<index>		<b>SENSOR Send Data Phone Index Parameter</b>
	1..4	Phone number index. Can be used to save several phone numbers.
<cell_phone>		<b>SENSOR Send Data Cell Phone Parameter</b>
	0.. 18446744073709551615	Cell Phone number up to 18 digits (leading zeros excluded).

### 4.19.5 Parameter Storage

None

### 4.19.6 Examples

Command	Responses
<b>AT\$DP=3,0033601020304</b> <i>Note: assign the cell-phone number 0033601020304 to cell phone n°3 on Sensor.</i>	OK

## 4.20 TD SENSOR SEND - Event Boot \$EB

### 4.20.1 Description

This command sends a Sensor message indicating a Boot Event. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.20.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.20.3 Syntax

*Action Command*

**AT\$EB**

OK

*Read Command*

None

*Test Command*

None

### 4.20.4 Parameters and Defined Values

None

### 4.20.5 Parameter Storage

None

### 4.20.6 Examples

Command	Responses
<b>AT\$EB</b> <i>Note: send a Boot Event to Sensor.</i>	OK

## 4.21 TD SENSOR SEND - Event Connection \$EC

### 4.21.1 Description

This command sends a Sensor message indicating a Connection Lost or Connection OK Event for a given device in a LAN. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.21.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.21.3 Syntax

*Action Command*

AT\$EC=<status>,<entry\_id>

OK

*Read Command*

None

*Test Command*

None

### 4.21.4 Parameters and Defined Values

Parameter	Value	Description
<status>		<b>SENSOR Send Event Connection Status Parameter</b>
	0	Connection Lost Event.
	1	Connection OK Event.
<entry_id>		<b>SENSOR Send Event Connection Entry ID Parameter</b>
	1..15	Address of the concerned device in the LAN. . Note that address 0 being reserved to the gateway it cannot be used for sending a Connection Event.

### 4.21.5 Parameter Storage

None

### 4.21.6 Examples

Command	Responses
AT\$EC=0,13 <i>Note: send the Connection Lost Event for Device of which network address is 13.</i>	OK

## 4.22 TD SENSOR SEND - Event RSSI \$ER

### 4.22.1 Description

This command sends a Sensor message indicating a RSSI Level Low or RSSI Level OK Event for a given device in a LAN. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.22.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.22.3 Syntax

*Action Command*

**AT\$ER=<entry\_id>,<status>**

OK

*Read Command*

None

*Test Command*

None

### 4.22.4 Parameters and Defined Values

Parameter	Value	Description
<entry_id>		<b>SENSOR Send Event RSSI Entry ID Parameter</b>
	1..15	Address of the concerned device in the LAN. Note that address 0 being reserved to the gateway it cannot be used for sending a RSSI Event.
<status>		<b>SENSOR Send Event RSSI Status Parameter</b>
	0	RSSI Level Low Event.
	1	RSSI Level OK Event.

### 4.22.5 Parameter Storage

None

### 4.22.6 Examples

Command	Responses
<b>AT\$ER=1,0</b> <i>Note: send the RSSI Low Event for Device of which network address is 1.</i>	OK

## 4.23 TD SENSOR SEND - Event Switch \$ES

### 4.23.1 Description

This command sends a Sensor message indicating a Switch ON or a Switch OFF Event for a given switch connected to a device. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.23.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.23.3 Syntax

*Action Command*

**AT\$ES=<port>,<bit>,<status>**

OK

*Read Command*

None

*Test Command*

None

### 4.23.4 Parameters and Defined Values

Parameter	Value	Description
<port>		<b>SENSOR Send Event Switch Port Parameter</b>
	0..5	Port number for the concerned switch.
<bit>		<b>SENSOR Send Event Switch Bit Parameter</b>
	1..255	Bit number for the concerned switch.
<status>		<b>SENSOR Send Event Switch Status Parameter</b>
	0	Switch OFF Event.
	1	Switch ON Event.

### 4.23.5 Parameter Storage

None

### 4.23.6 Examples

Command	Responses
<b>AT\$ES=0,15,1</b> <i>Note: send the Switch ON Event for the switch connected to port 0, bit 15.</i>	OK



## 4.24 TD SENSOR SEND - Event Temperature \$ET

### 4.24.1 Description

This command sends a Sensor message indicating a Temperature Level Low, Temperature Level OK or Temperature Level High Event. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.24.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.24.3 Syntax

*Action Command*

**AT\$ET=<status>**

OK

*Read Command*

None

*Test Command*

None

### 4.24.4 Parameters and Defined Values

Parameter	Value	Description
<status>		<b>SENSOR Send Event Temperature Status Parameter</b>
	0	Temperature Low Event.
	1	Temperature OK Event.
	2	Temperature High Event.

### 4.24.5 Parameter Storage

None

### 4.24.6 Examples

Command	Responses
<b>AT\$ET=1</b> <i>Note: send a Temperature OK Event.</i>	OK

## 4.25 TD SENSOR SEND - Event Battery \$EV

### 4.25.1 Description

This command sends a Sensor message indicating a Battery Level Low, Battery Level OK Event. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.25.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.25.3 Syntax

Action Command

AT\$EV=<status>  
OK

Read Command

None

Test Command

None

### 4.25.4 Parameters and Defined Values

Parameter	Value	Description
<status>		<b>SENSOR Send Event Battery Status Parameter</b>
	0	Battery Low Event.
	1	Battery OK Event.

### 4.25.5 Parameter Storage

None

### 4.25.6 Examples

Command	Responses
AT\$EV=1 <i>Note: send a Battery OK Event. Note that for Battery OK Event, battery level will be measured and added to the Sensor Message.</i>	OK

## 4.26 TD SENSOR SEND - Event Keep-Alive \$KA

### 4.26.1 Description

This command sends a Sensor message indicating a Keep-Alive Event. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received. Please keep in mind that this command is provided for event simulation only. Event messages should be sent automatically if the corresponding monitoring is enabled. Please do not send any Event message in your application.

### 4.26.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.26.3 Syntax

*Action Command*

**AT\$KA**

OK

*Read Command*

None

*Test Command*

None

### 4.26.4 Parameters and Defined Values

None

### 4.26.5 Parameter Storage

None

### 4.26.6 Examples

Command	Responses
<b>AT\$KA</b> <i>Note: send a Keep-Alive Event to Sensor.</i>	OK

## 4.27 TD SENSOR SEND - Service SMS \$\$\$SMS

### 4.27.1 Description

This command sends an SMS to a pre-registered cell-phone number (either via the Sensor Web Interface or the Data Phone Message). If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received.

### 4.27.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.27.3 Syntax

*Action Command*

**AT\$\$\$SMS=<message>**

OK

*Read Command*

None

*Test Command*

None

### 4.27.4 Parameters and Defined Values

Parameter	Value	Description
<message>		<b>SENSOR Send Service SMS Message Parameter</b>
	Up to 9 ASCII bytes	ASCII Message, up to 9 characters. Note that commas cannot be sent using this command. Moreover sending 0 will not work (but will return OK).

### 4.27.5 Parameter Storage

None

### 4.27.6 Examples

Command	Responses
<b>AT\$\$\$SMS=whats up?</b> <i>Note: send the message "whats up?" as a SMS to a pre-registered cell-phone.</i>	OK

## 4.28 TD SENSOR SEND - Service Tweet \$STWT

### 4.28.1 Description

This command sends a Tweet to a pre-registered tweet account (via the Sensor Web Interface). If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received.

### 4.28.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.28.3 Syntax

*Action Command*

**AT\$STWT=<message>**

OK

*Read Command*

None

*Test Command*

None

### 4.28.4 Parameters and Defined Values

Parameter	Value	Description
<message>		<b>SENSOR Send Service Tweet Message Parameter</b>
	Up to 9 ASCII bytes	ASCII Message, up to 9 characters.

### 4.28.5 Parameter Storage

None

### 4.28.6 Examples

Command	Responses
<b>AT\$STWT=tweeting</b> <i>Note: tweet the message "tweeting" on a pre-registered Twitter account.</i>	OK

## 4.29 TD SENSOR SEND - Raw \$RAW=

### 4.29.1 Description

This command sends a Raw Sensor Message. If the Module Type is set to Device, then a Local Register and a Sensor Register command must have previously been issued for the message to be received. If the Module type is set to either Gateway or Transmitter a Sensor Register command must have previously been issued for the message to be received.

### 4.29.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.29.3 Syntax

*Action Command*

**AT\$RAW=<hex\_byte1> [[ ]..<hex\_byte2>..]**  
OK

*Note: There can be from 0 to 10 <hex\_byte> parameter values, optionally separated by single or multiple space or tabulation characters.*

*Read Command*

None

*Test Command*

None

### 4.29.4 Parameters and Defined Values

Parameter	Value	Description
<hex_byte>		<b>SENSOR Send Raw Message Byte Value Parameter</b>
	00..FF	2-digit hexadecimal byte value ('0' to '9', 'a' to 'f' and 'A' to 'F' characters are valid) There can be from 0 to 10 <hex_byte> parameter values, optionally separated by single or multiple space (IA5 2/0) or tabulation (IA5 0/9) characters

### 4.29.5 Parameter Storage

None

### 4.29.6 Examples

Command	Responses
<b>AT\$RAW=01 02 03 04 05 06 07 08 09 0A</b> <i>Note: send a raw message that can be decoded on Sensor by configuring properly the Web Interface.</i>	OK

## 4.30 TD SENSOR SEND - Event Register \$REG

### 4.30.1 Description

This command allows registering a Module on Sensor.

### 4.30.2 Compatibility

This command is available in all firmware revisions since SOFT1154.

### 4.30.3 Syntax

*Action Command*

**AT\$REG**

OK

*Read Command*

None

*Test Command*

None

### 4.30.4 Parameters and Defined Values

None

### 4.30.5 Parameter Storage

None

### 4.30.6 Examples

Command	Responses
<b>AT\$REG</b> <i>Note: Register the module on Sensor.</i>	OK

### 4.30.7 Note

This command must be issued once prior any other command in order to ensure proper message processing for the SIGFOX™-compatible commands implemented by the TD1202 Module.

# TD1202 RM

---

## DOCUMENT CHANGE LIST

### Revision 1.0

- First Release

### Revision 1.1

- Removed obsolete parameter in \$ST command

### Revision 2.0

- Added the TD LAN AT commands
- Added a “compatibility” sub-section for each command

### Revision 2.1

- Added a separate section for the TD LAN AT commands
- Added an introduction sub-section for the TD LAN AT commands

### Revision 3.0

- Removed Sigfox-related AT commands
- Added Sensor AT commands

### Revision 3.1

- Changed contact information

### Revision 3.2

- Added notice for takedown of AT LAN and AT Sensor Command Sets starting with firmware SOFT1278

### Revision 3.4

- Changed contact information

### Revision 3.5

- Corrected port for DAC0 GPIO



**NOTES:**

## CONTACT INFORMATION

### Telecom Design S.A.

Zone Actipolis II — 2 bis rue Nully de Harcourt  
33610 CANEJAN, France  
Tel: +33 5 57 35 63 70  
Fax: +33 5 57 35 63 71

Please visit the Telecom Design web page:

<http://www.telecomdesign.fr/>

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Telecom Design assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Telecom Design assumes no responsibility for the functioning of undescribed features or parameters. Telecom Design reserves the right to make changes without further notice. Telecom Design makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Telecom Design assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Telecom Design products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Telecom Design product could create a situation where personal injury or death may occur. Should Buyer purchase or use Telecom Design products for any such unintended or unauthorized application, Buyer shall indemnify and hold Telecom Design harmless against all claims and damages.

Telecom Design is a trademark of Telecom Design S.A.

SIGFOX™ is a trademark of SigFox S.A.

Other products or brand names mentioned herein are trademarks or registered trademarks of their respective holders.