RS232 Communication protocol Titrette®

Titrette® Firmware 4.xx

This document describes the data transmission between the Titrette® and the PC. With this description the Titrette® can be connected with each computer, which has a standard RS232-communication interface. On the other hand you have to develop your own software application.

Using the delivered program **BRAND - Titrette® Software 4.xx** saves you the software development. In this case, you are not required to read this document!

Communication cable

Please use only original cables delivered by BRAND.

9 pin D-Sub female connector:

Pin	Direction (PC view)	Description
1	-	Not connected
2	In	RxD (PC receives data)
3	Out	TxD (PC sends data)
4	Out	DTR (Data Terminal Ready)
5	-	GND (ground)
6	-	Not connected
7	-	Not connected
8	-	Not connected
9	-	Not connected

During opening of the RS232-communication interface by the PC, DTR (Pin 4) has to be switched to positive potential. The Titrette® recognizes a connection by the DTR-level. The Titrette® will not send any data if the DTR is at negative potential or unconnected.

The DTR-signal is also used to support the batteries of the Titrette®. At DTR output, PCs usually cannot drive currents more than a few milliamperes, but this is enough to extend the lifetime of the batteries enormously.

Practical experience showed us, that cable lengths up to 20 metres are usable without problems. Extension cables must have at least the above named 4 wires (Pins 2, 3, 4 and 5).

RS232-communication interface parameters

The transmission parameters are fixed by the Titrette® and have to be set as follows:

Baudrate: 9600 Baud
Data Bits: 8
Stop Bits: 2
Parity: none

Establishing a communication connection

Open the RS232 serial interface within your application after you established a cable connection. Make sure, that the DTR-connection (Pin 4 of female connector) is switched to positive potential. The Titrette® is sending the control character RDY (see also 'packet transmission') for confirmation.

Packet Transmission

For the data transmission it is essential to cut the incoming, continuous data stream into data packets. For this cutting purpose, so called control characters are used. The user data is put into an 'envelope' made from control characters. The following control characters are used:

STX = 2;
ETX = 3;
EOT = 4;
ACK = 6;
NAK = 21dec = 15hex;
RDY = 135dec = 87hex;
EVT = 146dec = 92hex;
RST = 153dec = 99hex;

In the chosen transmission logic, the user data must not include control characters, otherwise the transmission logic is ambiguous.

Within the user data of a data packet, the position of the transmitted character determines its meaning. For example the event **Double Click "CLEAR":** In this event the characters 5 to 24 after the control character STX represent the serial number of the connected instrument.

Data types and the associated formats

A) Byte (8 Bit)

To avoid control characters in the user data, and further more make sure that the whole range (0…255 respectively. -128…+127) can be transmitted, a byte is split into 2 nibbles (4 Bit), and each nibble is send as ASCII-character. One nibble has the range 0 … 15. This range is sent as ASCII-character ‚0'… ‚9' and ‚A' … ‚F', most significant nibble of a byte first. This complies with the hexadecimal notation of the value. 2 characters are always sent (leading zeros).

Example1:   The value 93dec should be sent as byte. 93dec = 5Dhex. The ASCII characters ‚5' and ‚D' are transmitted.

Example 2: The negative value -3 should be sent. For transmission the two's complement is formed: -3dec = FDhex = ASCII ‚F' and ‚D'.

B) 16 Bit Integers

The value is split into 4 nibbles and they are transmitted the same way as a byte; always 4 characters (leading zeros), most significant first. Negative values are transformed into the two's complement.

Example 3: 785dec = 0311hex = ASCII ‚0' and ‚3' and ‚1' and ‚1'.

C) 32 Bit Integer

Analog to the 16 Bit integers; the only difference: 8 nibbles for each value.

D) Text

The ASCII-values of the characters in the text are transmitted as bytes. The ASCII-value 0 is sent as termination.

Example 4: "Brand" = 42hex, 72hex, 61hex, 6Ehex, 64hex and 00hex. So the string "4272616E6400" equates to the string "Brand".

Checksum Byte

To ensure a certain degree of data security, a checksum byte is built over a data packet. This checksum byte is sent with the data packet. With this checksum byte, your application has the possibility to check the packet transmission. The following easy algorithm is used for building the checksum byte:

All transmitted 8-bit-characters (bytes) of a data packet from STX (exclusive) to ETX (inclusive) are **bitwise exclusive or** (XOR). The checksum byte is sent next to ETX. The receiver of the data packet has to XOR the incoming data in the same way, **including** the checksum byte. If everything is correct, the result equates to zero!

Examples are shown in the description of the individual events!

Events

The following events cause the Titrette® to send a data packet to the PC. Your application must watch the input data at the COM-port, the data is transferred accordingly to the user action with the Titrette®.

- The Titrette® user has pressed the button **CLEAR** 2 times in quick succession.
- The Titrette® user enters/returns from menu mode.
- The Titrette® user has changed the calibration value in the menu (**CAL**).
- The Titrette® user has changed the next calibration date in the menu (**GLP**.
- The Titrette® user has changed the automatic power-off time in the menu (**APO**).

- The Titrette® user has changed the decimal place settings in the menu (**dP**).

In the course of this document these events are described in detail. Please note, that the Titrette® does not send any data if the symbol for the computer connection in Titrette®-display is missing (square with enter character). If this is the case, please check the cable connection and your application.

1. *Double click on the CLEAR-button of the Titrette®.*

In addition to the titration volume the Titrette® is sending the instrument data, which gives information about the instrument status and identifies the individual instrument. So, each measurement point can be verified exactly. The following data are transmitted additional to the titration volume:

A.) Serial number (10 characters)
B.) Instrument type respectively nominal capacity (NV10, NV25 or NV50)
C.) Adjustment value (CAL)
D.) Next calibration date

The acceptor for these events have to be confirmed by the PC, so the user of Titrette® can estimate the result of the action, if the PC screen is not visible to the user. If there is no confirmation the Titrette® switches to pause mode. Then the pause and connection symbols are flashing in the display. The user must reactivate the instrument by pressing the pause button of the Titrette®. The titration value remains in the display, the transmission attempt can be repeated (double click "CLEAR").

The data packet transmitted from the Titrette® byte by byte:

Pos	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Characters 1-4: event type: Double click on CLEAR
2	‚5'	0x35	53	
3	‚1'	0x31	49	
4	‚='	0x3d	61	
5	‚3'	0x33	51	Instrument Number 10 ASCII-characters.
6	‚0'	0x30	48	Example 09F0815
				1. ASCII ‚0' = 30hex
7	‚3'	0x33	51	
8	‚9'	0x39	57	2. ASCII ‚9' = 39hex
9	‚4'	0x34	52	
10	‚6'	0x36	54	3. ASCII ‚F' = 46hex
11	‚3'	0x33	51	
12	‚0'	0x30	48	4. ASCII ‚0' = 30hex
13	‚3'	0x33	51	
14	‚8'	0x38	56	5. ASCII ‚8' = 38hex
15	‚3'	0x33	51	
16	‚1'	0x31	49	6. ASCII ‚1' = 31hex
17	‚3'	0x33	51	
18	‚5'	0x35	53	7. ASCII ‚5' = 35hex
19	‚0'	0x30	48	

20	,0'	0x30	48	8. Termination byte 0
21	,F'	0x46	70	
22	,F'	0x46	70	9. Not used; Filler byte FFhex
23	,F'	0x46	70	
24	,F'	0x46	70	10. Not used; Filler byte FFhex
25	,3'	0x33	51	Instrument type respectively. Nominal capacity (NV25=25; NV50=50) Example NV50 = 32hex
26	,2'	0x32	50	
27	,0'	0x30	48	Titration volume in µl (32 Bit Integer) Example 23,854 ml = 23854µl = 00005D2Ehex
28	,0'	0x30	48	
29	,0'	0x30	48	
30	,0'	0x30	48	
31	,5'	0x35	65	
32	,D'	0x44	68	
33	,2'	0x32	50	
34	,E'	0x46	69	
35	,0'	0x30	48	Adjustment value CAL (16 Bit Integer) Example +145µl = 0091hex (Negative Adjustment value in the two's complement, e.g., -23µl = FFE9hex)
36	,0'	0x30	48	
37	,9'	0x39	57	
38	,1'	0x31	49	
39	,0'	0x30	48	Next calibration date GLP (Year past 2000) Example 2009 = 09hex
40	,9'	0x39	57	
41	,0'	0x30	48	Next calibration date GLP (Month; 01=January) Example August = 08hex
42	,8'	0x38	56	
43	ETX	0x03	3	Control character ETX
44		0x00	0	Checksum over characters 1-43 (Byte by byte XOR) „051=3039463038313500FFFF3200005D2E00910908" and ETX = 0hex
45	RDY	0x87	135	Control character RDY

For conformation send the following data packet to the Titrette®:
RST, EOT, STX, 31hex, 31hex, 30hex, ETX, 33hex
respectively decimal: 153, 4, 2, 49, 49, 48, 3, 51

After the reception of confirmation the Titrette® sends an ACK and a RDY
respectively decimal: 6, 135 again.

After this, the event is complete.

2. *Menu mode entry and exit*

After the entry into the menu mode (pressing and holding the CLEAR-button of the Titrette®) the instrument is reporting the instrument state in the same way it reports the return from menu mode.

One byte is send. Value = 0: return or 1: entry

Pos.	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Characters 1-4: event type: menu mode activated/completed
2	‚5'	0x35	53	
3	‚0'	0x30	48	
4	‚='	0x3d	61	
5	‚0'	0x30	48	=1 → menu mode entered
6	‚1'	0x31	49	=0 → menu mode exited
7	ETX	0x03	3	Control character ETX
8		0x0A	10	Checksum over „050=01" and ETX (Byte by byte XOR)
9	RDY	0x87	135	RDY

3. *Calibration Value changed (CAL)*

When the user has changed the CAL value, the Titrette® reports this event with a 16 bit integer. This integer value gives the correction for the nominal capacity in µl.

Pos.	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Character 1-4: Event type: Menu setting changed
2	‚5'	0x35	53	
3	‚2'	0x32	50	
4	‚='	0x3d	61	
5	‚B'	0x42	66	Key of Menu setting: BFhex = Adjustment value (CAL)
6	‚F'	0x46	70	
7	‚0'	0x30	48	New Adjustment value in hexadecimal form. Here 0x0091 = 145dec. = +145µl (always 4 characters with leading 0)
8	‚0'	0x30	48	
9	‚9'	0x39	57	
10	‚1'	0x31	49	
11	ETX	0x03	3	Control character ETX
12		0x05	5	Checksum over „052=BF0091" and ETX (Byte by byte XOR)
13	RDY	0x87	135	Control character RDY

4. *Next Calibration Date changed (GLP)*

When the user has changed the GLP date, the Titrette® reports this new date with 1 byte for the year (since 2000) and 1 byte for the month (1: January, … , 12:december).

Pos.	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Characters 1-4: Event type: Menu setting changed
2	‚5'	0x35	53	
3	‚2'	0x32	50	
4	‚='	0x3d	61	
5	‚F'	0x46	70	Key of Menu setting: FDhex = Calibration date (GLP)
6	‚D'	0x44	68	
7	‚0'	0x30	48	Value in hexadecimal form. Here 907 = July2009
8	‚9'	0x39	57	
9	‚0'	0x30	48	(always 4 characters with leading 0)
10	‚7'	0x37	55	
11	ETX	0x03	3	Control character ETX
12		0x05	5	Checksum over „052=FD0907" and ETX (Byte by byte XOR)
13	RDY	0x87	135	Control character RDY

5. *Auto Power Off changed (APO)*

When the user has changed the Auto Power Off time, the Titrette® reports this new data as 16 bit integer in steps of 15 seconds. To get the new APO-time in minutes, divide the value by 4.

Pos.	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Characters 1-4: Event type: Menu setting changed
2	‚5'	0x35	53	
3	‚2'	0x32	50	
4	‚='	0x3d	61	
5	‚F'	0x46	70	Key of Menu setting: FEhex = Auto Power-off time (APO)
6	‚E'	0x45	69	
7	‚0'	0x30	48	Value in hexadecimal form. Here 0x001C = 28dec = 28 * 15 Sec. = 7 Minutes
8	‚0'	0x30	48	
9	‚1'	0x31	49	(always 4 characters with leading 0)
10	‚C'	0x43	67	
11	ETX	0x03	3	Control character n ETX
12		0x78	120	Checksum over „052=FE001C" and ETX (Byte by byte XOR)
13	RDY	0x87	135	Control character RDY

6. *Decimal Places changed (dP)*

When the user changes the display resolution (decimal places), the new resolution is coded in bit #3 in the transferred byte. If this bit is 1, 3 decimal places are set, if this bit is 0, 2 decimal places are set.

Pos.	ASCII	Hex	Dec	Comment
	EVT	0x92	146	Control character: an event occurred
0	STX	0x02	2	Control character start of packet transmission
1	‚0'	0x30	48	Characters 1-4: Event type: Menu setting changed
2	‚5'	0x35	53	
3	‚2'	0x32	50	
4	‚='	0x3d	61	
5	‚E'	0x46	70	Key of Menu setting: EFhex = Decimal places
6	‚F'	0x44	68	
7	‚0'	0x30	48	Value in hexadecimal form. Here 0x09. Bit #3 (0x08) provides information about the number of decimal places. 0 → 2 places, 1 → 3 places. Here bit #3 = 1, means 3 decimal places are actually displayed.
8	‚9'	0x39	57	
9	ETX	0x03	3	Control character ETX
10		0x02	2	Checksum over „052=EF09" and ETX (Byte by byte XOR)
11	RDY	0x87	135	Control character RDY

Request for data

The PC might send a data request to the Titrette®. The Titrette® completes such a request by sending an appropriate answer via RS232-interface.

How to send a request for data? Follow the step by step instructions:

1.) The PC sends the following request data packet:

Pos.	ASCII	Hex	Dec	Comment
	RST	0x99	153	Reset Interface function within Titrette®
0	EOT	0x04	4	Control byte End Of Text
1	‚0'	0x30	48	Bytes 1-3: Command code: In this example 17 = „017" = Fetch Titrette® display data without clearing display. Additional data is sent in the same way as with double click on CLEAR. EVT-Byte is replaced by ACK-Byte.
2	‚1'	0x31	49	
3	‚7'	0x37	55	
4	ENQ	0x05	5	End of data packet.

2.) The Titrette® answers immediately after reception of the request data packet with an ACK-Byte.

3.) Finally the Titrette® sends the requested data. The data format is different for all the various requests.

The following list describes the answers of the Titrette® according to all data requests:

1. **Request current volume (LCD) in the same data format as with double click on CLEAR-button. LCD is <u>not</u> cleared!**

   Command code „017"

   For a table containing the returned data from Titrette®, look at **Events → 1. Double click on the CLEAR-button of the Titrette®.** The data format is the same as for this event except that the EVT-byte is replaced by ACK-byte.

2. **Request current volume (LCD). No additional data is transmitted. LCD is cleared.**

   Command code „007"

Pos.	ASCII	Hex	Dec	Comment
	ACK	0x99	153	Acknowledge the request
0	STX	0x04	4	Control character STX.
1	‚0'	0x30	48	Characters 1-3: Confirmation of the request.
2	‚0'	0x30	48	Same format as with request reception. „007" =
3	‚7'	0x37	55	Send LCD volume, clear display.
4	‚='	0x3d	61	Start of user data.
5	‚0'	0x30	48	32-bit value current volume. E.g. 13.492ml =
6	‚0'	0x30	52	13492µl = 000034B4 hex.
7	‚0'	0x30	48	
8	‚0'	0x30	56	
9	‚3'	0x33	48	
10	‚4'	0x34	50	
11	‚B'	0x42	66	
12	‚4'	0x34	52	
13	ETX	0x03	3	Control character ETX.
14		0x78	120	Checksum over „007=000034B4" and ETX (bytewise XOR)
15	RDY	0x87	135	Control character RDY. Completes the request. No confirmation is necessary from PC side! Titrette® is ready for new requests.

3. **Request current volume (LCD). No additional data is transmitted. LCD is not cleared.**

Command code „008"

Pos.	ASCII	Hex	Dec	Comment
	ACK	0x99	153	Acknowledge of the request
0	STX	0x04	4	Control character STX.
1	‚0'	0x30	48	Characters 1-3: Confirmation of the request.
2	‚0'	0x30	48	Same format as with request reception. „008" =
3	‚8'	0x38	56	Send LCD volume, do <u>not</u> clear display.
4	‚='	0x3d	61	Start of user data.
5	‚0'	0x30	48	32-bit value current volume. E.g. 13.492ml =
6	‚0'	0x30	52	13492µl = 000034B4 hex.
7	‚0'	0x30	48	
8	‚0'	0x30	56	
9	‚3'	0x33	48	
10	‚4'	0x34	50	
11	‚B'	0x42	66	
12	‚4'	0x34	52	
13	ETX	0x03	3	Control character ETX.
14		0x77	119	Checksum over „008=000034B4" and ETX (bytewise XOR)
15	RDY	0x87	135	Control character RDY. Completes the request. No confirmation is necessary from PC side! Titrette® is ready for new requests.

4. **Request Serial number**

Command code „016"

Pos.	ASCII	Hex	Dec	Comment
	ACK	0x99	153	Acknowledge of the request
0	STX	0x04	4	Control character STX.
1	‚0'	0x30	48	Characters 1-3: Confirmation of the request.
2	‚1'	0x31	49	Same format as with request reception. „016" =
3	‚6'	0x36	54	Send serial number.
4	‚='	0x3d	61	Start of user data.
5	‚3'	0x33	51	Serial number 9 ASCII characters. E.g.
6	‚0'	0x30	48	"09F0815" 1. ASCII ‚0' = 30hex
7	‚3'	0x33	51	
8	‚9'	0x39	57	2. ASCII ‚9' = 39hex
9	‚4'	0x34	52	
10	‚6'	0x36	54	3. ASCII ‚F' = 46hex
11	‚3'	0x33	51	
12	‚0'	0x30	48	4. ASCII ‚0' = 30hex
13	‚3'	0x33	51	
14	‚8'	0x38	56	5. ASCII ‚8' = 38hex

15	,3'	0x33	51	
16	,1'	0x31	49	6. ASCII ,1' = 31hex
17	,3'	0x33	51	
18	,5'	0x35	53	7. ASCII ,5' = 35hex
19	,0'	0x30	48	
20	,0'	0x30	48	8. limiter 0 = 00hex
21	,F'	0x46	70	
22	,F'	0x46	70	9. not used; filler byte FFhex
23	ETX	0x03	3	Control character ETX.
24		0x0E	14	Checksum over „016=3039463038313500FF" and ETX (bytewise XOR)
25	RDY	0x87	135	Control character RDY. Completes the request. No confirmation is necessary from PC side! Titrette® is ready for new requests.

5. **Request firmware version**

Command code „001"

Pos.	ASCII	Hex	Dec	Comment
	ACK	0x99	153	Acknowledge of the request
0	STX	0x04	4	Control character STX.
1	,0'	0x30	48	Characters 1-3: Confirmation of the request.
2	,0'	0x30	48	Same format as with request reception. „001" =
3	,1'	0x31	49	Send firmware version of instrument and sensor
4	,='	0x3d	61	Start of user data.
5	,0'	0x30	48	16-Bit value firmware version of the instrument.
6	,4'	0x34	52	E.g. V4.08.
7	,0'	0x30	48	Main version 4, Sub version 08.
8	,8'	0x38	56	
9	,0'	0x30	48	16-Bit value firmware version sensor. E.g. V2.13.
10	,2'	0x32	50	Main version 2, Sub version 13.
11	,0'	0x30	48	
12	,D'	0x44	68	
13	ETX	0x03	3	Control character ETX.
14		0x75	117	Checksum over „001=0408020D" and ETX (bytewise XOR)
15	RDY	0x87	135	Control character RDY. Completes the request. No confirmation is necessary from PC side! Titrette® is ready for new requests.