

TITEL: **Serial Protocol Remote Mode**

PROJECT: VIAFLO_Pipettes

AUTHOR: M.Camastral

DOCUMENT NAME: SPEC_Protocol_RemoteMode_V12.docx

Document Change History

Vers.	Status:	Name:	Date:	Change:
V00	Draft	MCa	17. Sept. 2014	Document issued
	Release	MCa	5. Nov. 2014	Update, Release
V01	Draft	MCa	12. Nov. 2014	SetAction extended
	Release	MCa	08. May 2015	Release
V02	Draft	MCa	20. June 2015	Volume factor 5000ul SC
V03	Draft	RPi	05. Okt 2015	Added new action command for VOYAGER II
	Released	RPi	08. Okt 2015	Release
V04	Draft	TWe	03. Nov 2015	New commands: Abort, SetScreen, SetBrightness. New State: Error Spacer.
	Released	TWe	20. Nov 2015	Release
V01.05.00.00	Released	TWe	22. Apr 2016	Update for Release Pipet Firmware V3.31
V05	Released	IMe	11. Jan. 2017	Document revision without DLL information (back to the Vxx prefix) Add example for message frame.
V07	Released	PSo	10. Feb. 2017	Add new action commands
V09	Released	MCa	27. Aug. 2018	Update model number, Released
V10	Released	MCa	20. Nov. 2018	Bug fix model number FW 03.xx
V11	Released	TWe	02. March 2020	Update Status Code Add new action "Get Battery Info"
V12	Released	MCa	04. June 2020	Correction Byte# SetAction command

Table of Contents

1	Introduction	2
1.1	Purpose	2
1.2	Abbreviations, Acronyms and Definitions	2
1.3	Setup connection Host – Pipette	2
1.3.1	Bluetooth	2
1.3.2	Wire	2
2	Serial Protocol	3
2.1	Messages frame from Host to Pipette (Input)	3
2.2	Responses from Pipette to Host (Output)	4
2.3	Status code	4
2.4	Data flow control	4
3	Message types	4
3.1	Type 1 (0x01): Get Info	4
3.2	Type 2 (0x02): Get Action Status	5
3.3	Type 3 (0x03): Get Calibration Factor	6
3.4	Type 4 (0x04): Set Calibration Factor	6
3.5	Type 5 (0x05): Set Action	7

INTEGRA

3.6	Type 6 (0x06): Exit Remote mode (Disconnect)-----	8
3.7	Type 7 (0x07): Power Off -----	8
3.8	Type 8 (0x08): Abort Command-----	8
3.9	Type 9 (0x09): Set Screen-----	9
3.10	Type 10 (0x10): Set Brightness -----	9
3.11	Type 11 (0x11): Get Battery Info-----	9
4	Example procedure 'Set Action' -----	10
5	Example Hex-code messages -----	11

1 Introduction

1.1 Purpose

This document explains the serial communications between the Integra VIAFLO pipette and an external Host (PC) for remote control. The communication works over Quasi-RS232 (2.7V).

This document describes the DLL Version 01.09.00.00. This version works with INTEGRA RemoteControl Version 03.21.00.00 and with the Pipet Firmware V04.21 or higher.

1.2 Abbreviations, Acronyms and Definitions

Abbr. / Def.	Description
Byte#	Byte number counted from start of message (w/o STX)
Host	Master device (generally a PC) who sends commands to the pipette

1.3 Setup connection Host – Pipette

The connection between the host and the Integra pipette is established over Bluetooth or wire.

Activate the communication mode: Go to the “Toolbox” and enter the menu “Communications”, select “Remote Mode” and press “OK”.

Note: In the Remote Mode the pipettes automatically Power Off is deactivated! The device stays in the mode until you press 'Disconnect', send an 'Exit Remote' or 'Power Off' command or the battery is discharged.

1.3.1 Bluetooth

For this insert a Bluetooth module in to the pipette.

Setup a Bluetooth connection on the PC. The Pipette and PC need to be paired. Please refer to the help of your PC to learn how to pair Bluetooth devices. The following pairing code might be required: 12345.

1.3.2 Wire

The communication is on the same interface as the USB stand / USB-cable is connected (back side of Pipette).

Note: The interface offers a quasi RS232 interface on a 2.7V level (not +/-12V).

INTEGRA

2 Serial Protocol

All Messages from Host to pipette are Master-Slave. All transmissions occur at:

Baud rate	115'200 bit/sec
Data bit	8
Parity bit	None
Stop bit	1
Handshake	None

NOTE: To allow "reserved" characters to be contained in binary messages an ESC byte is sent before each of the following bytes:

Character	Value (hex)
STX (start-of-text)	0x02
ETX (end-of-text)	0x03
ESC	0x1B

Multi byte values:

"4 Byte Data" (Buffer)			
Byte0 (Buffer[i+3])	Byte1	Byte2	Byte3 (Buffer[i])

C code would be:

value = (buffer[i] * 256 * 256 * 256) + (buffer[i+1] * 256 * 256) + (buffer[i+2] * 256) + buffer[i+3];

"2 Byte Data" (Buffer)	
Byte0 (Buffer[i+1])	Byte1 (Buffer[i])

C code would be:

value = (buffer[i] * 256) + buffer[i+1] ;

2.1 Messages frame from Host to Pipette (Input)

The length is the number of bytes between the STX and the EXT excluding any leading ESC character.

Name	STX	Length	Checksum	Sequence number	Resend flag	Message type	Data (Body)	ETX
Default	0x02							0x03
Bytes	1	2	1	2	1	2	n	1
Byte#		0	2	3	5	6	8	

Details:

<STX>	start-of-text
<2 byte binary length>	The length is the number of bytes between <STX> and <ETX> excluding any leading <ESC> characters.
<1 byte binary checksum>	The sum of all bytes between <STX> and <ETX> excluding any leading <ESC> characters. Result of the sum is truncated to just the lower 8 bits and subtracted from 256.
<2 byte binary sequence number>	This number is generated by Host to identify each message and should be echoed by the pipette.
<1 byte binary resend flag>	This is normally a 0 but set to 1 if the message is a repeat of the previous transmission. The sequence number will be the same as the previous message
<2 byte binary message type>	Message type
<n byte binary data>	This is the data to be read from or written to the pipette.
<ETX>	end-of-text

Example "Get Info" command:

INTEGRA

Name	STX	Length	Checksum	Sequence number	Resend flag	Message type	Data (Body)	ETX
Default	0x02	0x00,0x08	0xF6	0x00,0x01	0x00	0x00,0x01	-	0X03

2.2 Responses from Pipette to Host (Output)

The responses from the pipette have nearly the same format to the sent messages (except error code).

Name	STX	Length	Checksum	Sequence number	Resend flag	Message type	Status code	Data (Body)	ETX
Default	0x02						0		0X03
Bytes	1	2	1	2	1	2	2	n	1
Byte#		0	2	3	5	6	8	10	

Details:

<STX>	start-of-text
<2 byte binary length>	The length is the number of bytes between <STX> and <ETX> excluding any leading <ESC> characters.
<1 byte binary checksum>	The sum of all bytes between <STX> and <ETX> excluding any leading <ESC> characters. Result of the sum is truncated to just the lower 8 bits and subtracted from 256.
<2 byte binary sequence number>	This number is generated by Host to identify each message and should be echoed by the pipette.
<1 byte binary resend flag>	This is normally a 0 but set to 1 if the message is a repeat of the previous transmission. The sequence number will be the same as the previous message
<2 byte binary message type>	Message type
<2 byte status code>	Status code
<n byte binary data>	This is the data to be read from or written to the pipette.
<EXT>	end-of-text

2.3 Status code

Value:	Status:	Comment:
0	Command accepted	
1	Unknown message type	
2	Command value/parameter out of range	At least one value is out of range or invalid number of parameters / bytes
3	Hardware error	Use GetActionStatus to get more information.
4	Command not accepted	Use GetActionStatus to get more information.

2.4 Data flow control

No Flow Control, all the commands will be responded within 100ms.

3 Message types

3.1 Type 1 (0x01): Get Info

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
--------	--------	-------	----------

INTEGR

8-9	2	Status code	
10	1	Firmware version major	
11	1	Firmware version minor	
12-13	2	Hardware Version (Electronics)	
14-17	4	Serial Number	
18-19	2	Model number*	

*The model number depends on FW version. It is different between version 03.xx and 04.xx!

Table model number:

Model #	FW 03.xx	FW 04.xx
0	None	12.5 µl SC
1	12.5 µl MC	12.5 µl MC 8ch
2	12.5 µl Voyager 8ch	12.5 µl MC 12ch
3	12.5 µl Voyager 12ch	12.5 µl MC 16ch
4	125 µl MC	12.5 µl VOYAGER 8ch
5	125 µl Voyager 8ch	12.5 µl VOYAGER 12ch
6	125 µl Voyager 10ch	50 µl SC
7	125 µl Voyager 12ch	50 µl MC 8ch
8	300 µl MC	50 µl MC 12ch
9	300 µl Voyager 4ch	50 µl MC 16ch
10	300 µl Voyager 5ch	50 µl VOYAGER 8ch
11	300 µl Voyager 6ch	50 µl VOYAGER 12ch
12	300 µl Voyager 8ch	125 µl SC
13	300 µl Voyager 10ch	125 µl MC 8ch
14	1250 µl MC	125 µl MC 12ch
15	1250 µl Voyager 4ch	125 µl MC 16ch
16	1250 µl Voyager 5ch	125 µl VOYAGER 8ch
17	1250 µl Voyager 6ch	125 µl VOYAGER 12ch
18	1250 µl Voyager 8ch	300 µl SC
19	12.5 µl SC	300 µl MC 8ch
20	125 µl SC	300 µl MC 12ch
21	300 µl SC	300 µl VOYAGER 4ch
22	1250 µl SC	300 µl VOYAGER 6ch
23	5000 µl SC	300 µl VOYAGER 8ch
24	STEP1100 (for testing)	1250 µl SC
25	50 µl SC	1250 µl MC 8ch
26	50 µl MC	1250 µl MC 12ch
27	-	1250 µl VOYAGER 4ch
28	-	1250 µl VOYAGER 6ch
29	-	1250 µl VOYAGER 8ch
30	-	5000 µl SC
31	-	STEP1100 (for testing)

3.2 Type 2 (0x02): Get Action Status

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	
10-11	2	Action status	
12-13	2	Hardware error code	

INTEGRA

Action status:

Value:	Status:	Comment:
0	Ready	
1	Wait for BlowIn	Send 'BlowIn' as next
2	Wait for RUN Key	
3	Busy	An action is executed
4	Pipette not homed	Send 'Home pipette' as next
5	User abort	Action or 'Wait RUN key' aborted
6	Error Spacer	
7	Battery too low	Pipet version 04.21 or higher

Hardware error code:

Value:	Error:	Comment:
0	No hardware error	
5	Critical error: ADC overrun	Device inoperable!
18	Critical error: Battery voltage too high	
20	Critical error: Overload charge current	
21	Critical error: Vref out of range	
30	Critical error: SW/HW incompatible	
98	Critical error: μ C Quartz failed (HSE)	

3.3 Type 3 (0x03): Get Calibration Factor

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	
10-11	2	Calibration factor 'Pipet'	multiplied by 10'000
12-13	2	Calibration factor 'Repeat'	multiplied by 10'000

3.4 Type 4 (0x04): Set Calibration Factor

Data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Calibration factor 'Pipet'	multiplied by 10'000
10-11	2	Calibration factor 'Repeat'	multiplied by 10'000

Calibration Range: 9000 – 11000

4 decimal places so the actual calibration factors are 0.9000 – 1.1000 (Default 1.0000)

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

Settings will be stored in the non volatile memory.

INTEGRAL

3.5 Type 5 (0x05): Set Action

Data (body):

Byte#:	Bytes:	Info:	Comment:
8	1	Action: 1 = Aspirate 2 = Dispense ¹ 3 = Mix ¹ 4 = Purge ¹ 5 = BlowOut 6 = BlowIn 7 = Dispense with no BlowOut 8 = Home pipette ² 9 = Space (Voyager only) 10 = Home Spacer (Voyager only) 11 = Mix with no BlowOut 12 = Relative Mix, Aspirate first 13 = Relative Mix, Dispense first	
9	1	Speed	1...10 Note: for actions 'Aspirate', 'Dispense', 'Dispense with no BlowOut', 'Mix' and 'Purge' otherwise ignored, BlowOut / BlowIn are executed with the last used speed.
10-11	2	volume_value = real_volume [μl] * Factor[1/ μl]	Factor[1/ μl] depends on maximum volume of pipette model. ⁴ (for actions 'Aspirate', 'Dispense', 'Dispense with no BlowOut and 'Mix')
12	1	Mix cycles	1...30 Note: for Mix actions otherwise ignored
13	1	RUN confirmation	Value not 0: The pipette screen shows 'Press RUN' and the action will be executed after pressing RUN.
14-33	20	Message string ³	Message is showed on screen. Allowed characters: ASCII 32..255
34-35	2	Spacing ⁵	Spacing range is dependent upon pipette model, but all values are in $\frac{1}{10}$ mm. Note: for Space action only otherwise ignored

1. The last dispense or mix to volume = 0 or purge will add an automatic BlowOut.
The host must send a BlowIn before the next aspirate.
2. The 'Home Pipette' command set the speed to default (8).
3. The message is showed in a proportional font. The maximum of visible characters depends on the string content.
4. The factor and the min. - and max. value depends on type of pipette model:

Volume type	Factor[1/ul]	Min. volume_value	Max. volume_value
12.5ul	100	50	1250
50ul	100	100	5000
125ul	10	20	1250
300ul	10	50	3100
1250ul	10	250	12500
5000ul	10	1000	50000

INTEGRA

Example calculation for a volume of 250ul with a 300ul pipet:

$\text{volume_value[]} = \text{Factor}[1/\text{ul}] * \text{real_volume}[\text{ul}]$

$\text{volume_value[]} = 10/\text{ul} * 250\text{ul} = \underline{2500}$

5. Pipette model spacing dependencies

# of channels	Volume type	Spacing [mm]	
		Min.	Max.
4	300 / 1250	9.0	33.0
6	300 / 1250	9.0	19.8
8	12.5 / 50 / 125	4.5	14.1
8	300 / 1250	9.0	14.1
12	12.5 / 50 / 125	4.5	9.0

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

After starting an action read the action status periodical (Get Action Status).

3.6 Type 6 (0x06): Exit Remote mode (Disconnect)

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

After receiving the command the pipette sends the answer and terminates the remote mode.

The command is not accepted if the Action Status is 'Busy'.

3.7 Type 7 (0x07): Power Off

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

After receiving the command the pipette sends the answer, waits 200ms and switches off the power.

The command is not accepted if the Action Status is 'Busy'.

3.8 Type 8 (0x08): Abort Command

No data (body).

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

Note: Abort the "Wait for RUN Key" status or the commands: 'Aspirate', 'Dispense', 'Purge' and 'Mix'. After canceling an active command it is necessary to home the pipette.

INTEGRAL

3.9 Type 9 (0x09): Set Screen

Data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Number of Screen ¹	0 to 3

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

1. Screen Numbers
 - 0: Default remote control screen
 - 1: Custom screen 1
 - 2: Custom screen 2
 - 3: Black screen

Note: This value will not be stored. After a reset the default 'Remot Control' screen will be active.

3.10 Type 10 (0x10): Set Brightness

Data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Display Brightness	0 (Off) to 10

Response data (body):

Byte#:	Bytes:	Info:	Comment:
8-9	2	Status code	

Note: This value will not be stored. After a reset the brightness which is set in the pipet settings will be reloaded.

3.11 Type 1 (0x11): Get Battery Info

No data (body).

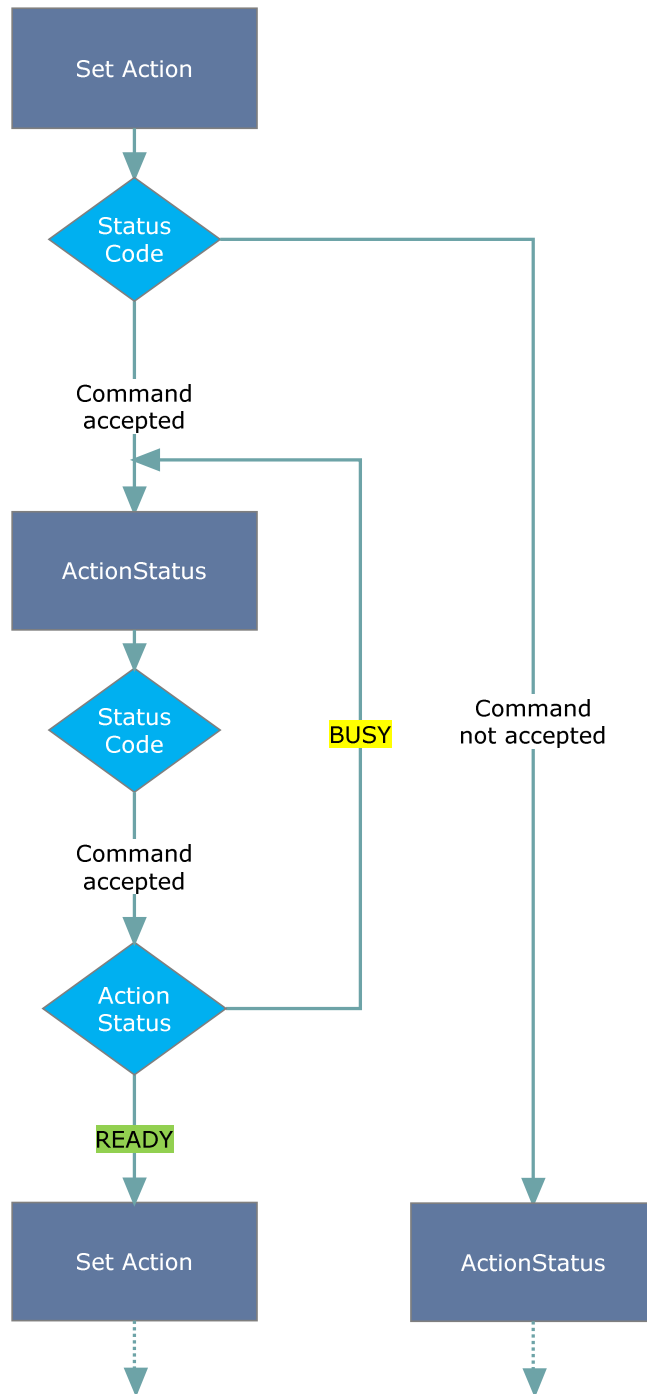
Response data (body):

Byte#:	Bytes:	Info:	Comment:
10	1	State of charge [%]	0 to 100 [%] 255 means the value could not be read out.
11	1	State Bits: Bit 0: External supply	

INTEGRA

4 Example procedure 'Set Action'

After a pipette action with the response status code "Command accepted" the pipette action runs until the action status "Ready" is received.



INTEGRA

5 Example Hex-code messages

- Aspirate: (Vol: 1000ul; Speed: 8; Mix Cycles: 3; Message: "Integra"; No RUN key confirm.)
02 00 24 76 00 00 00 00 05 01 08 1B 03 E8 1B 03 00 49 6E 74 65 67 72 61 20 20 20 20 20 20
20 20 20 20 20 20 00 00 03
- Mix: (Vol: 1000ul; Speed: 8; Mix Cycles: 3; Message: "Integra"; RUN key confirm.)
02 00 24 73 00 00 00 00 05 1B 03 08 1B 03 E8 1B 03 01 49 6E 74 65 67 72 61 20 20 20 20 20 20
20 20 20 20 20 20 00 00 03
- Purge: (Vol: 0ul; Speed: 5; Mix Cycles: 0; Message: "Integra"; No RUN key confirm.)
02 00 24 64 00 00 00 00 05 04 05 00 00 00 00 49 6E 74 65 67 72 61 20 20 20 20 20 20 20 20 20
20 20 20 00 00 03