

Clarity Controls

LC UNI


LC

ENG

Code/Rev.: M042/24C
Date: 17.1.2011

Phone: +420 251 013 400
Fax: +420 251 013 401
clarity@dataapex.com
www.dataapex.com

DataApex Ltd.
Podohradská 1
155 00 Prague 5
The Czech Republic

Clarity[®], DataApex[®] and [®] are trademarks of DataApex Ltd. Microsoft[®] and Windows[™] are trademarks of Microsoft Corporation.
DataApex reserves the right to make changes to manuals without prior notice. Updated manuals can be downloaded from www.dataapex.com.

Author: DM

Contents

1 LC UNI Control Module	1
2 Customizing the LC UNI control module	2
2.1 Strategy for profile development	2
2.2 LC UNI Profile.....	3
2.2.1 Connection.....	3
2.2.2 Command Format	5
2.2.3 Init.....	8
2.2.4 Set Pressure Limits.....	10
2.2.5 Set Flow.....	12
2.2.6 Error Status.....	14
2.2.7 Get Values.....	16
3 Troubleshooting	18
3.1 LC UNI control specifics and limitations.....	19
4 Appendices	20
4.1 List of variables.....	20
4.2 ASCII code table.....	22

To facilitate the orientation in the **LC UNI** manual and **Clarity** chromatography station, different fonts are used throughout the manual. Meanings of these fonts are:

Instrument (blue text) marks the name of the window, to which the text refers.

Open File (italics) describes the commands and names of fields in **Clarity**, parameters that can be entered into them or a window or dialog name (when you already are in the topic describing the window).

WORK1 (capitals) indicates the name of the file and/or directory.

ACTIVE (capital italics) marks the state of the station or its part.

The bold text is sometimes also used for important parts of the text and the name of the **Clarity** station. Moreover, there are text sections written in format other than normal text. These sections are formatted as follows:

Note: Notifies the reader of possibly interesting information.

Caution: Warns the user of possibly dangerous or very important information.

█ Marks the problem statement or trouble question.

Description: Presents any closer information on the problem, describes its causes etc.

Solution: Marks the response to the question, presents a procedure how to remove it.

1 LC UNI Control Module

The **DataApex LC UNI Pump Control** is universal LC driver designed to function with **LC Pumps** controlled by text commands using the COM port (RS232).

Any controlled pump must be able to change flow upon external command. It is not possible to use this module to control pumps that need to receive the whole gradient table at once.

Complete list of pumps supported at the time by existing **LC UNI** profiles is available in the **D004-Clarity-Controls-List-of-Controlled-instruments** datasheet, the particular profiles for these pumps can be found in the C:\CLARITY\UTILS\UNI_DRIVERS\ directory of the **Clarity** installation directory. It is also possible to create your own profile for controlling other pumps, as described in the chapter "**Customizing the LC UNI control module**" on pg 2.

2 Customizing the LC UNI control module

As an addition to predefined profiles, **LC UNI** module enables the user to operate any pump that is controlled by serial communication and has a command for controlling the flow. This means that if you know the communication protocol of your pump, you can make your own control module or modify some profile already existing.

Note: It is advisable to read the chapter "**Troubleshooting**" on pg **18**. in this manual before starting to make your new LC pump profile.

2.1 Strategy for profile development

As the communication protocol of a pump is usually considerably complex, it is advisable to use the following sequence of steps during the profile development:

1. Set the communication parameters to match those of the pump.
2. Start only with the *Init* command first. Commands well suited for *Init* command test are the version request or pump status.
3. Try to establish the communication.
4. Add the other functions stepwise.

Note: It is a good idea to save the profile often.

2.2 LC UNI Profile

2.2.1 Connection

The **UNI Pump Control Setup** dialog consists of seven tabs which describe the behavior of the pump based on the loaded profile and allow to develop new profiles either from the scratch, or by changing the already existing profiles. This topic only describes the **Connection** tab needed for setting the communication with the pump, other tabs necessary for the development and functionality of the UNI pump profile are described in the chapter "**Customizing the LC UNI control module**" on pg 2.

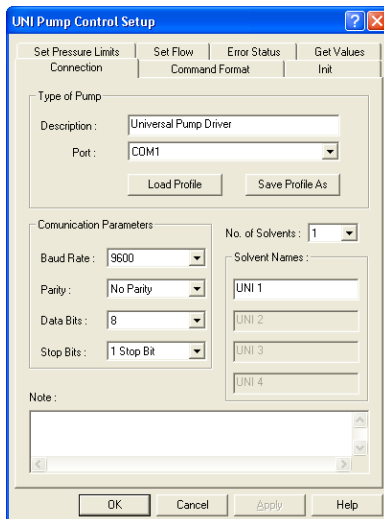


Fig 1: UNI Pump Control Setup - Connection tab

Type of Pump

The LC UNI pump control module enables to load the profile from a file or store the current profile as a new file. The *Type of Pump* section allows for better orientation in the currently opened profile.

Description

Holds the description of the profile, usually noting the type of the pump it is usable for.

Port

Sets the COM port to which the LC Pump is connected.

Load Profile

Loads the stored LC UNI profile (a file with the *.UNI suffix). Predefined profiles are stored in the UTILS subdirectory of the **Clarity** directory (C:\CLARITY\UTILS by default).

Save Profile As

Saves the currently opened LC UNI profile under a new name. Select the filename and folder where the profile should be saved in the **Save As** dialog.

Communication Parameters

The communication settings depend on the pump being connected. Check your LC Pump documentation for appropriate communication parameters.

Note: The communication parameters including the COM *Port*, with the exception of the *Baud Rate*, are not stored in the profile. When using the profile to control the pump, the communication parameters set in the **UNI Pump Control Setup** dialog must match those set in the pump.

Baud Rate

Sets the Baud Rate.

Parity

Sets the parity.

Data Bits

Sets the Data Bits.

Stop Bits

Sets the Stop Bits.

No. of Solvents

Sets the number of controlled solvents. Based on this selection, appropriate fields in *Solvent Names* section will be enabled.

Caution! If possible, add only the solvents that will be used in the gradient. The gradient table might otherwise not be able to calculate the missing information (e.g. percentages, total flow, etc.).

Solvent Names

Sets the names for individual solvents.

Note

Allows to add the user comments on the profile. The field is well suited for example for the information on which firmware version or **Clarity** version was the profile invented etc.

2.2.2 Command Format

The communication with the pump is generally performed by text commands. In order for the pump to understand these commands it has to be sent in a proper format.

Caution: The format (shell) of the commands has to be set in hexadecimal mode. For more details on hexadecimal format please see the chapter "**ASCII code table**" on pg 22.

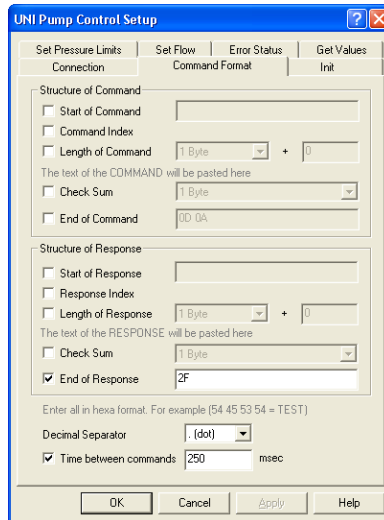


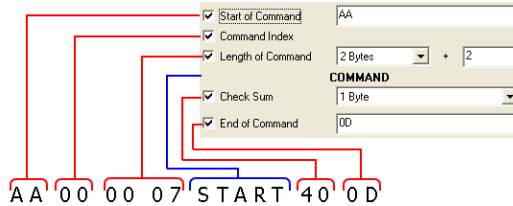
Fig 2: UNI Pump Control Setup - Command Format

The **Command Format** tab is divided in two sections:

- The *Structure of Command* section sets the parameters of commands sent from **Clarity** to the LC Pump.
- The *Structure of Response* section sets the format of the commands sent from the LC Pump to **Clarity**.

The universal communication module is designed to be able to format the following arbitrary formatting properties of the communication:

An example of the command structure



The shell of the command (red) is entered in hexmode and its format will be common to all commands. The individual commands (blue) will be sent enclosed in the shell.

"2 Bytes + 2" = The length will be represented by 2 Bytes and it will include the length of the COMMAND + CHECKSUM + END

Fig 3: The UNI Pump command structure

Start of Command/Response

Formatting characters introducing each command.

Command/Response Index

A byte containing the information about the ordinal number of the command.

Note: This *Command Index* is used in certain types of pumps (e.g. Flux).

Length of Command/Response

One or two Bytes carrying the information about the length.

- The first field decides whether the information about the length will be sent in 1 Byte or in 2 Bytes
- The + X figure can contain following values:
 - 0 - when the length value reflects only the length of the command
 - 1 - when the length includes the command + one byte of checksum
 - 2 - when the length includes the command + one byte of Check Sum + one byte of End (or length of the command + 2 bytes of the Check Sum – it depends on the type of the Check Sum specified in the following field).

Check Sum

Selects the type of the checksum. The options determine the length of the Check Sum value (1 or 2 Bytes) and parts of the command/response and its format from which the checksum will be calculated

(*COMMAND/RESPONSE* , *Length of Command/Response* , *Command/Response Index* , *Start of Command/Response*). Available options are:

- 1 Byte, 2 Bytes - The one or two bytes of the check sum will be calculated from the *COMMAND/RESPONSE* only.

- *1 Byte with Length, 2 Bytes with Length* - The one or two bytes of the check sum will be determined from the *COMMAND/RESPONSE* and from the *Length of Command/Response*.
- *1 Byte with Index & Length, 2 Bytes with Index & Length* - The same as previous option + *Command/Response Index*
- *1 Byte with Start & Index & Length, 2 Bytes with Start & Index & Length* - The same as previous option + *Start of Command/Response*.

End of Command/Response

Selects the formatting characters ending each command.

Note: In most cases the ending characters are <CR> <LF>, which corresponds to 0D 0A in hexa mode.

Decimal Separator

Selects the sign used as decimal separator that will be used in the body of the *COMMAND* for formatting of the numbers. Available options are either *.* (*dot*) or *,* (*comma*).

Time between commands

Sets a minimum time interval in milliseconds, after which the next command is sent to pump. The field is necessary for pumps that are not able to accept a new command before the previous one was performed.

2.2.3 Init

The **Init** tab provides parameters for testing the connection with the LC pump and for sending the initializing set of commands during the opening of the **Clarity Instrument** window and for sending the closing commands during closing of the **Clarity Instrument** window.

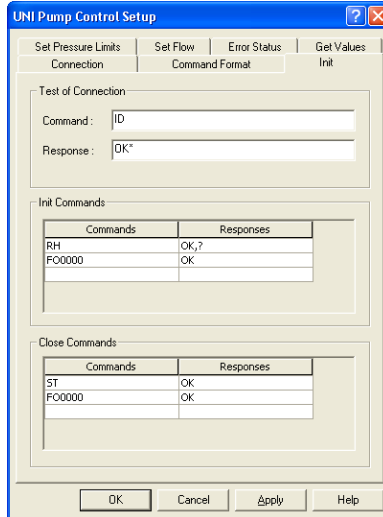


Fig 4: UNI Pump Control Setup - Init

Test of Connection

This section serves for the testing of the connection to the pump. This test is performed whenever the **Clarity Instrument** on which the pump is configured opens.

Note: The version or status requests are recommended for initial test of connection. Use wildcard (*, ?) characters in the response field if the response may vary.

Command

Sets the command to be sent during the testing of the connection.

Note: The **LC UNI** control module waits for a response for the time set in the *Timeout* field specified in the **Error Status** tab. If it will not receive the expected response, it will generate an error message and the communication will not be established.

Response

Sets the expected response from the device that will confirm that the connection is successfully established. If the *Response* field is left empty, the driver will expect no response and won't wait for one.

Init Commands

This table serves for specifying the initialization commands that will be sent to the pump after the above-mentioned test of connection finished successfully. The *Commands* column defines the commands that will be sent to the device, while the *Responses* column holds the expected responses. If the *Responses* field for the given command is left empty, the driver will expect no response and won't wait for one.

Note: Initialization commands can be typically used to start the communication, set the type of heads, reset the device, etc.

Close Commands

This table serves for specifying the closing commands that will be sent to the pump when closing the **Clarity Instrument** window. The *Commands* column defines the commands that will be sent to the device, while the *Responses* column holds the expected responses. If the *Responses* field for the given command is left empty, the driver will expect no response and won't wait for one.

Note: Closing commands are typically used to finish the communication, switch off the remote control mode, etc.

2.2.4 Set Pressure Limits

The [Set Pressure Limits](#) tab serves for setting the pressure limits in the pump.

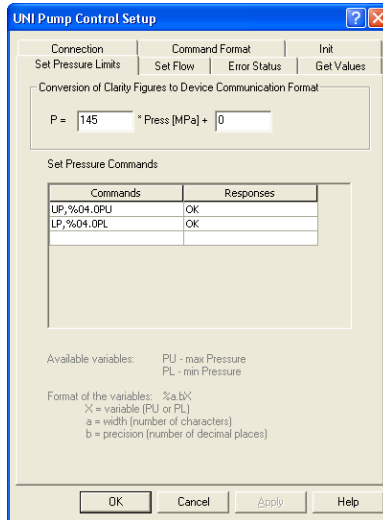


Fig 5: UNI Pump Control Setup - Set Pressure Limits

Conversion of Clarity Figures to Device Communication Format

This section serves for setting the calculation equation for pressure between **Clarity** and the pump figures. The edit fields in the equations present the means for setting the scale (first field) and the offset (second field) for the conversion of figures sent to the device. The control module uses MPa as pressure units.

Note: Let us assume that the pump receives the pressure in the form which tells that 0 is a pressure of 0 MPa, while getting the value of 350 would set the pressure to 35 MPa. The correct figures to enter would be 10 for the first field and 0 for the second field.

Set Pressure Commands

This table serves for specifying the commands for setting-up the minimum and maximum pressure allowed. The *Commands* column defines the commands that will be sent to the device, while the *Responses* column holds the expected responses. If the *Responses* field for the given command is left empty, the driver will expect no response and won't wait for one. The formatting and variables are described in the chapter "**Appendices**" on pg 20.

The following variables can be used in the commands:

PU: Upper pressure limit

PL: Lower pressure limit

Note: Both *PU* and *PL* variables are substituted by values entered into the respective fields of the [Gradient Options](#) dialog when the command is sent to the pump.

2.2.5 Set Flow

The **Set Flow** tab sets the flow units conversion ratio between **Clarity** and the pump, the commands necessary to set the flow of the pump and, if necessary, the commands to run and stop the pump(s).

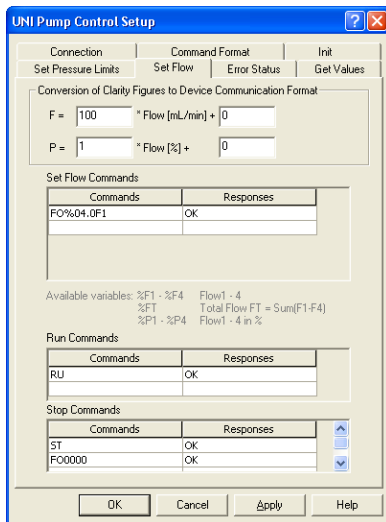


Fig 6: UNI Pump Control Setup - Set Flow

Conversion of Clarity Figures to Device Communication Format

This section serves for setting the calculation equations between **Clarity** and the pump figures. The edit fields in the equations present the means for setting the scale (first field) and the offset (second field) for the conversion of figures sent to the device.

Note: Let us assume that the pump receives the flow in the form which tells that 0 is a flow of 0 ml/min, while getting the value of 1000 would set the flow to 10 ml/min. The correct figures to enter would be 100 for the first field and 0 for the second field.

F and P are the values that the pump uses, counted from values taken from **Clarity**. Some pumps use different format for receiving and sending the flow.

The formatting conventions described in the chapter "**Appendices**" on pg 20. do not apply in the *Conversion of Clarity Figures to Device Communication Figures* section.

Set Flow Commands

This section sets the commands that will be used for setting the actual flow in the pump. The formatting and variables are described in the chapter

"Appendices" on pg 20. The following variables can be used in the commands:

%F1 - %F4: Flow of the solvent 1 - 4

%FT: Total Flow; $FT = \text{Sum}(F1-F4)$

%P1 - %P4: Flow of the solvent 1 - 4 in %

Note: On isocratic pumps that will be used in high pressure gradients, use **%F1** when getting the flow of the solvent. The **%FT** variable should only be used for stand alone pumps or in combination with **%Px** variables on low pressure gradient pumps.

Note:

On some pumps combined commands can be used to send more parameters, for example the Set Flow command can also eventually include the pressure limit parameters.

Run Commands

This table serves for specifying the commands that will be sent to the pump to start it - some pumps need not only to get the desired flow, but also to enter the running state. The *Commands* column defines the commands that will be sent to the device, while the *Responses* column holds the expected responses. If the *Responses* field for the given command is left empty, the driver will expect no response and won't wait for one.

Stop Commands

This table serves for specifying the commands for stopping of the pump - some pumps need not only to set the flow to zero, but also switching them from the running state to cease working. The *Commands* column defines the commands that will be sent to the device, while the *Responses* column holds the expected responses. If the *Responses* field for the given command is left empty, the driver will expect no response and won't wait for one.

2.2.6 Error Status

The **Error Status** tab sets the commands for testing potential error status of the device.

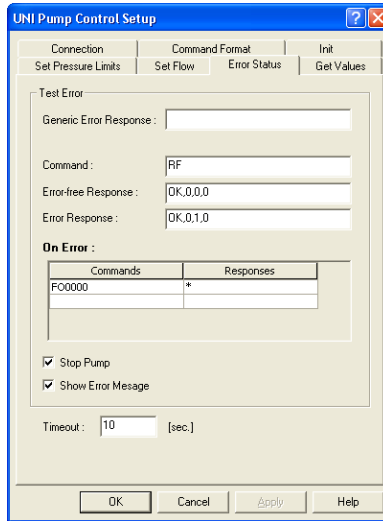


Fig 7: UNI Pump Control Setup - Error Status

Test Error

Generic Error Response

Sets the string for which all responses of the pump are checked. All responses from the device are checked for the specified string.

Command

Sets the command for checking the error status of the pump.

Note: Once the *Command* field has been filled in it is necessary to fill both the *Error-free Response* and *Error Response* fields.

Error-free Response

Sets the expected response in case that the pump does not issue any error status.

Error Response

Sets the expected response in case that the pump has issued an error status.

Note: The error response from the device is first checked against the *Error Response* field. Only if it does not match it would be checked against the *Error-free Response* field. Consequently the *Error-free Response* field can contain the wildcards (*, ?) that may be valid also for the *Error Response*.

On Error

Sets the commands to be sent to the pump after the error status has been detected. In the second column expected responses from the pump have to be stated.

Caution! When a response to an *On Error* command is recognized by *Error Response* or *Generic Error Response*, the communication may freeze in an endless cycle. To avoid this, use wildcard (*) in this field.

Stop Pump

Sets the behavior of the pump in case of the emerging error. When checked all running solvents will be stopped if error occurs.

Show Error Message

Enables the display of the error messages when error occurs.

Timeout

Sets a time (in seconds) for which the driver will wait for a response to a command. If the response has not been received within the specified time, a communication error will be generated.

2.2.7 Get Values

The **Get Values** tab is used to set the request format that will ask the pump for actual flow and pressure. These figures will be later displayed in the **Device Monitor** window and will be requested from the pump twice per second.

Note: If the pump or the LC UNI control module is busy performing other actions, the refresh rate will be slower.

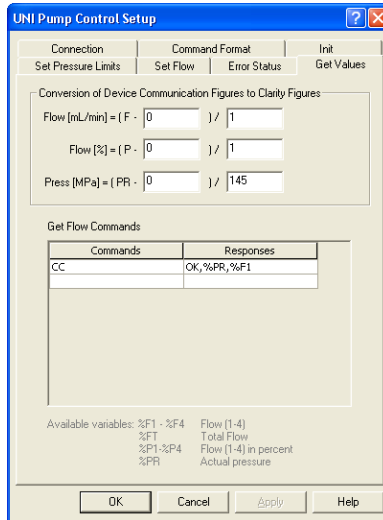


Fig 8: UNI Pump Control Setup - Get Values

Conversion of Device Communication Figures to Clarity Figures

This section serves for setting the calculation equations between the pump figures and **Clarity**. The edit fields in the equations present the means for setting the offset (first field) and the scale (second field) for the conversion of figures received from the device.

Note: Let us assume that the pump sends the flow in the form which means that 0 is a flow of 0 ml/min, while sending the value of 1000 would mean the flow of 10 ml/min. The correct figures to enter would be 0 for the first field and 100 for the second field.

The formatting conventions described in the chapter "**Appendices**" on pg 20. do not apply in the *Conversion of Device Communication Figures to Clarity Figures* section. The LC UNI control module will calculate any missing figures providing that it has enough information to calculate them (e.g. will calculate %FT - Total Flow if it has %F1 - %F4).

Flow [ml/min]

Sets the conversion equation of the flow figures between the pump and **Clarity** formats.

Note: Flow has to be expressed in ml/min regardless to the setting of units used for display in the **Clarity Device Monitor** dialog.

Flow [%]

Sets the conversion equation of the relative flow figures between the pump and **Clarity** formats.

Press [MPa]

Sets the conversion equation of the pressure figures between the pump and **Clarity** formats.

Note: Pressure has to be expressed in MPa regardless to the setting of units used for display in the **Clarity Device Monitor** dialog.

Get Flow Commands

This section sets the commands that will be used for gaining the actual flow and pressure figures from the pump. The following variables can be used in the commands:

%F1 - %F4: Flow of the solvent 1 - 4

%FT: Total Flow; $FT = \text{Sum}(F1-F4)$

%P1 - %P4: Flow of the solvent 1 - 4 in %

%PR: Actual Pressure

Note: On isocratic pumps that will be used in high pressure gradients, use **%F1** when getting the flow of the solvent. The **%FT** variable should only be used for stand alone pumps or in combination with **%Px** variables on low pressure gradient pumps.

3 Troubleshooting

When the remedy for some problem cannot be discovered easily, the recording of communication between **Clarity** and the pump can significantly help the **DataApex** support to discover the cause of the problem.

The recording can be enabled by adding or amending the COMMDRV.INI file in the **Clarity** installation directory (C:\CLARITY by default). The file can be edited in any text editor (e.g. Notepad). Following section should be edited or added:

```
[COM1]
echo=on
textmode=on
filename=LC_UNI.txt ; %D or %d could be used to include a current
date (recommended in combination with the Reset=Off option
reset=off
```

Note: Instead of COM1 type the correct serial port used to communicate with the pump causing problems.

The created *.TXT files will greatly help in diagnosis of unrecognized errors and problems in communication.

3.1 LC UNI control specifics and limitations

■ The “Cannot establish communication with XXX pumps!” message appears while the Instrument window opens.

Description: The communication failed to initialize. The causes may be:

The communication parameters are incorrect

Solution: These parameters must be set the same in the pump and in the control module. Check your pump manual for corresponding settings.

Late or unexpected response from the pump

Solution: The communication is synchronous, i.e. the module waits for a reply for each command sent to the pump (unless specified otherwise). The response must be received within the specified *Timeout* time. Some pumps can respond quite slowly to some commands (for example *Reset* command or rapid flow rate changes).

■ Pump Communication Error is reported

Description: This message can appear after the initialization or anytime during a run. As a consequence of the Pump Communication Error the “Waiting, pump not ready” status in the **Instrument** window *Status Line* will be displayed. The reasons of the Pump Communication Error may be following:

Late or unexpected response from the pump - The communication is synchronous, i.e. the module waits for a reply for each command sent to the pump (unless specified otherwise). The response must be received within the specified *Timeout* time. Some pumps can respond quite slowly to some commands (for example *Reset* command or rapid flow rate changes). Also, when the response is not recognized (as a correct answer or global error response) the communication error will be generated.

Not accepted command - Unless the “not accepted” responses are used in the *Generic Error Response* field, such response will cause the communication error.

Solution: It is usually necessary to restart the **Clarity Instrument** window and/or the pump to recover from the error.

■ Pump Error is reported

Description: This message is generated, when a recognized error response has been detected. In some cases, the communication can be recovered by sending the method once more (using the *Send Method* and *OK* buttons from the **Method Setup** dialog) or using the *Resume Idle* command/button from the **Device Monitor** window.

Not accepted command - When the “not accepted” responses are used in the *Generic Error Response* field, such response will cause the Pump Error.

Pump Error Response - The pump *Error Response* will cause this message – usually it is due to the exceeding of pressure limits.

Solution: The running analysis must be stopped or aborted first. The reason for the error may need to be corrected on the pump.

4 Appendices

4.1 List of variables

The following variables are used through the whole control module. It means that the value determined in one dialog can be used in other dialogs of the **UNI Pump Control** module.

Format specification fields

A format specification, which consists of optional and required fields, has the following form:

`%[flags] [width] [.precision] [variable]`

Each field of the format specification is a single character or a number signifying a particular format option

The optional fields, which appear before the Variable, control other aspects of the formatting, as follows:

Flags

A flag directive is a character that justifies output and prints signs, blanks, decimal points, and octal and hexadecimal prefixes. More than one flag directive may appear in a format specification.

Tab 1: Flag Characters

Flag	Meaning	Default
0	If width is prefixed with 0, zeros are added until the minimum width is reached. If 0 and - appear, the 0 is ignored. If 0 is specified with an integer format (i, u, x, X, o, d) the 0 is ignored.	No padding.

Width

The width argument is a nonnegative decimal integer controlling the minimum number of characters printed. If the number of characters in the output value is less than the specified width, blanks are added to the left until the minimum width is reached.

If width is prefixed with **0**, zeros are added until the minimum width is reached.

The width specification never causes a value to be truncated. If the number of characters in the output value is greater than the specified width, or if width is not given, all characters of the value are printed.

Precision

The precision specification specifies a nonnegative decimal integer, preceded by a period (.), which specifies the number of decimal places.

Unlike the width specification, the precision specification can cause rounding of a floating-point value.

Variables

Variables are strings that will be automatically identified in the commands or responses and replaced by the current values.

Tab 2: Variables in commands sent

Command	Meaning
%PU	Upper (max) pressure limits
%PL	Lower (min) pressure limits
%F1 - %F4	Flows of individual solvents
%P1 - %P4	Flows of individual solvents in percent
%FT	Total Flow

The formatting conventions are applied to the sent commands.

Tab 3: Variables in responses received

Command	Meaning
%FT	Total Flow
%F1 - %F4	Flows of individual solvents
%P1 - %P4	Flows of individual solvents in percent
%PR	Current pressure
?	Arbitrary character (wildcard)
*	Arbitrary string (wildcard)

The formatting conventions are not applied to the received responses.

4.2 ASCII code table

Tab 4: List of available ASCII codes

HEXA	Char	Note
00	NUL	(Null char.)
01	SOH	(Start of Header)
02	STX	(Start of Text)
03	ETX	(End of Text)
04	EOT	(End of Transmission)
05	ENQ	(Enquiry)
06	ACK	(Acknowledgment)
07	BEL	(Bell)
08	BS	(Backspace)
09	HT	(Horizontal Tab)
0A	LF	(Line Feed)
0B	VT	(Vertical Tab)
0C	FF	(Form Feed)
0D	CR	(Carriage Return)
0E	SO	(Shift Out)
0F	SI	(Shift In)
10	DLE	(Data Link Escape)
11	DC1	(XON) (Device Control 1)
12	DC2	(Device Control 2)
13	DC3	(XOFF)(Device Control 3)
14	DC4	(Device Control 4)
15	NAK	(Negative Acknowledgement)
16	SYN	(Synchronous Idle)
17	ETB	(End of Trans. Block)
18	CAN	(Cancel)
19	EM	(End of Medium)
1A	SUB	(Substitute)
1B	ESC	(Escape)
1C	FS	(File Separator)
1D	GS	(Group Separator)
1E	RS	(Request to Send)(Record Separator)
1F	US	(Unit Separator)
20	SP	(Space)

HEXA	Char	Note
21	!	(exclamation mark)
22	"	(double quote)
23	#	(number sign)
24	\$	(dollar sign)
25	%	(percent)
26	&	(ampersand)
27	'	(single quote)
28	((left/opening parenthesis)
29)	(right/closing parenthesis)
2A	*	(asterisk)
2B	+	(plus)
2C	,	(comma)
2D	-	(minus or dash)
2E	.	(dot)
2F	/	(forward slash)
30	0	
31	1	
32	2	
33	3	
34	4	
35	5	
36	6	
37	7	
38	8	
39	9	
3A	:	(colon)
3B	;	(semi-colon)
3C	<	(less than)
3D	=	(equal sign)
3E	>	(greater than)
3F	?	(question mark)
40	@	(AT symbol)
41	A	
42	B	
43	C	
44	D	
45	E	
46	F	

HEXA	Char	Note
47	G	
48	H	
49	I	
4A	J	
4B	K	
4C	L	
4D	M	
4E	N	
4F	O	
50	P	
51	Q	
52	R	
53	S	
54	T	
55	U	
56	V	
57	W	
58	X	
59	Y	
5A	Y	
5B	[(left/opening bracket)
5C	\	(back slash)
5D]	(right/closing bracket)
5E	^	(caret/circumflex)
5F	_	
60	`	
61	a	
62	b	
63	c	
64	d	
65	e	
66	f	
67	g	
68	h	
69	i	
6A	j	
6B	k	
6C	l	

HEXA	Char	Note
6D	m	
6R	n	
6F	o	
70	p	
71	q	
72	r	
73	s	
74	t	
75	u	
76	v	
77	w	
78	x	
79	y	
7A	z	
7B	{	(left/opening brace)
7C		(vertical bar)
7D	}	(right/closing brace)
7E	~	(tilde)
7F	DEL	(delete)