Command Line Interface

Document Revision: 1.4
Document Date: 31.May.2020

Purpose

This command line interface is intended for developers who want to seamlessly embed MediCollector functionality into their applications. Using this interface, you can quickly and easily acquire data from one of our supported medical devices and use that data within your own application. When the command line is executed, it will query the specified medical device and return the latest values for all numeric signals into a text file for you.

Please note the following:

- This interface is available in all versions of our products. It is recommended that developers first download and test the interface using our MediCollector BEDSIDE product because it has a GUI which is useful for testing. When ready for deployment, it can be replaced with MediCollector CLI which is a trimmed down version without a GUI.
- The data is outputted to a tab-delimited text file and saved to the path specified by the programmer. The text file will have 2 rows. The first row contains the names of each signal. And the second row contains the signal values.
- The first column is always the timestamp (unless the TESTONLY parameter is TRUE). The timestamp is in the following fixed format: YYYY.MM.DD hh:mm:ss.000. The time is in 24-hour time and retrieved from the system clock on your PC. And the time always has 3 digits after the decimal point.
- If you run this command line using an unlicensed copy of MediCollector (e.g. in time-limited demo mode), it will only return a valid value for one randomly chosen signal. All other signals/columns will be replaced with the word "DEMO". This will allow you test the software. To remove this restriction, you need to purchase a valid license from our website and we will ship you a dongle to unlock the restriction.
- If an error occurs (e.g. cable is disconnected), the outputted text file will contain details about the error instead of the actual data. You will know if an error occurs because the output file will contain only 1 line of text, which will start with the word "Error".
- When you execute this command line, MediCollector will wait up to a maximum timeout of 15 seconds for data. If, for whatever reason, no data is received within that timeout, it will abort and generate an error.
- This interface will only retrieve NUMERIC values, such as HR, BR, RR, Pulse, SpO2, BP, ABP, Temp, etc. It will not retrieve waveforms such as ECG. Obviously, there is no point in retrieving a single instantaneous value of a high frequency waveform, such as ECG, as it is constantly changing and becomes meaningless if only 1 datapoint is returned.
- You can optionally run the command with the /TESTONLY parameter. When doing so, the command line will not actually retrieve data. Instead, it will just check to confirm that the specified device is detected on the specified port. If the device is detected, "OK" will be written to the target text file (and the Return Code will be 0). If the device is not detected, the relevant error message will be written to the text file (and the Return Code will be 1).
- Users should be careful when using the command line interface in the MediCollector SERVICE version of our products. If you attempt to run the command line while the service is running, you might get an error because the service is occupying the COM port.

Syntax

If using the MediCollector BEDSIDE or CLI product, you must run the *medicollector.exe* executable:

medicollector.exe "*PathToOutputFile*" **/device** "*DeviceName*" [**/port** *ComPort*] [**/nodialog**] [**/overwrite**] [**/testonly**]

If using the MediCollector SERVICE product, you should call the *medicollector_service.exe* executable:

medicollector_service "*PathToOutputFile*" **/device** "*DeviceName*" [**/port** *ComPort*] [**/nodialog**] [**/overwrite**] [**/testonly**]

Return Code

After execution, you can check for errors in 2 manners:

- The return code is stored in %errorlevel% after execution. The values are 0 = No error, 1 = Error.
- View the content of your target text file at *PathToOutputfile* for error message.

Parameters

PathToOutputFile	Required	This is the absolute path where the output file will be saved. If a file already exists at this path, an error will be generated. This is parameter is required. You must use double-quotes around this parameter if there is a space character in the path.
/device *DeviceName*	Required	This is the name of the device as seen in the list in MediCollector. This is parameter is required. You must use double-quotes around this parameter if there is a space character in it, as seen in the example below. Some options for this parameter are: • "Philips Intellivue Series" • "GE S/5 Series" •
/port *ComPort*	Sometimes Required	This is the COM port where the device is connected. This parameter is required if the specified device requires a COM port, otherwise the parameter is optional.
/nodialog	Optional	If present, this parameter will prevent a progress dialog box from appearing when the command line is called. This is parameter is optional.
/overwrite	Optional	If present, this will allow the *PathtoOutputFile* to be overwritten. If not present, and a file already exists at *PathToOutputFile*, then an error will be returned because the file will not be overwritten.
/testonly	Optional	If present, MediCollector will not retrieve data from the specified device. Instead, it will check to confirm the device is present on the specific port by attempting to communicate with the device. If the device is detected, "OK" will be written to the target text file (and the Return Code will be 0). If the device is not detected, the relevant error message will be written to the text file (and the Return Code will be 1).

Examples

Below is an example of how to retrieve data from a Philips Intellivue device on COM1 using our MediCollector BEDSIDE product:

medicollector "C:\temp\my output file.txt" /device "Philips Intellivue Series" /port COM1

Below is the same command executed using the MediCollector SERVICE product:

medicollector_service "C:\temp\my output file.txt" /device "Philips Intellivue Series" /port COM1

Below is an example of how to retrieve data from a GE S/5 device, such as a Carescape B850, on COM2:

medicollector "C:\temp\my output file.txt" /device "GE S/5 Series" /port COM2

Below is an example of how to retrieve data from the Simulated Device without showing a progress dialog and by overwriting the text file at the specified path:

medicollector "C:\temp\my output file.txt" /device "Simulated Device" /nodialog /overwrite

Below is an example of how to use the command line to check if a GE SOLAR device is attached to COM1 (e.g. using the */testonly* parameter). This command will not show a progress dialog and will overwrite the target text file if it exists:

medicollector "C:\temp\my output file.txt" /device "GE Solar/Dash Series" /nodialog /overwrite /testonly

Below are 2 commands which could be pasted into a batch file. The first command will run the command line interface and pause execution until data collection finishes. The second line returns the error code after execution is complete.:

START /wait medicollector "C:\temp\my output file.txt" /device "Simulated Device" /nodialog /overwrite
echo %errorlevel%

Output

The output is a tab-delimited text file that looks like this:

TIMESTAMP	HR	BR	RR	Temp	NBP(DIA)	NBP(SYS)	SpO2
2019.04.29 13:23:10.000	60	60	60	98	120	80	99

If running in demo mode, only one randomly selected cell will return a value. All others will be masked, as seen below. To remove this restriction, you will need to purchase a valid license.

TIMESTAMP	HR	BR	RR	Temp	NBP(DIA)	NBP(SYS)	SpO2
2019.04.29 13:23:10.000	DEMO	DEMO	DEMO	98	DEMO	DEMO	Demo

If an error occurs, the text file will contain the error message as seen below. You will know if an error occurs because the output file will contain only 1 line of text, which will start with the word “Error:”.

```
Error 123:  The cause of error is just an example blah blah.
```