Description	Technical specification for interface between the SA3001 Surface Analyst and other devices through a TCP socket connection		
Author	Rick Koval		
Distribution	Individuals implementing the socket communications.		
Doc. History	**Revision**	**Date**	**Change**
	1	2021-04-22	First release
	2	2021-05-28	Update purge timing wording because of variable hardware Change MC command reference for image rotation
	3	2021-06-03	Update timestamps to show milliseconds
	4	2021-11-01	Correct Time example in GetLastPCHK>
	5	2021-11-09	Add note on rates to section 1 and rename to Control API

1. Introduction

This specification describes a command/response communication channel for integrators or end customers to use for the purpose of automating the operation of a Surface Analyst in a production or laboratory setting. It is expected that the Surface Analyst local touchscreen is still utilized for configuration settings, setting up surface profiles, and many maintenance functions.

NOTE: Important points to keep in mind when using the Control API and Data API:

- Using the Data API will have an impact on overall inspection rates possible.
  - When using the control API and Data API at the same time a sustained rate over 600 inspections per hour may not be possible.
- The Data API is not a real time communications channel – so standard TCP timeouts should take this into account with the expectation that any individual command/response may take up to 30 seconds or longer to complete for complex queries or when the system is under heavy load.
- Heavy use of the Data API can also impact the responsiveness of the Control API.

2. Structure of the protocol and document

There are two general types of communication commands, those which are used to control a normal production sequence, and those commands which are only used for troubleshooting or debugging purposes. In this document the two types will be separated but the commands are not limited to the functions described (so the troubleshooting type commands can be used in production if they prove to be useful).

ABBREVIATIONS:

- SA - SA3001 Surface Analyst
- RD - Remote device. This is the device which is connected through the Ethernet connection to the SA

KEY:

- Green items are the normal operation of the commands.
- Orange items indicate a possible path for a retry or automatic correction taking place.
- Red items indicate an exception has occurred which disrupts the normal operational flow.
- Blue items are cancellation commands which only need to be used if a long running command needs to be cancelled. Because the cancellation commands fall outside of the normal command sequence it is possible for the communications to overlap and extra responses be received by the RD. Care should be

taken to provide a robust communications handling routine which can gracefully handle these types of situations.

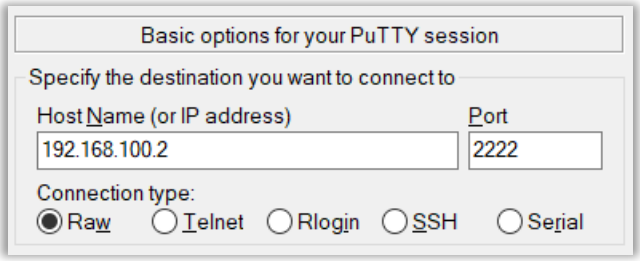
3. Ethernet connection details

The SA connects through a USB to Ethernet adaptor which gets the IP address from a DHCP server.

- The SA accepts TCP commands on port 2222. This port can be changed if needed by using the MC command “tcpc:x”, contact BTG Labs for assistance.
- The data in each packet sent from the RD contains a single string using ASCII (UTF8) encoding. All commands sent from the RD are terminated with the '>' character followed by a CrLf pair (except for binary image data packets).
- The SA will send reply TCP packets that are ASCII (UTF8) to the opened connection. Images are binary and packaged in PNG format. Some commands will echo the command immediately then return a second result packet when the request has been completed.
- Commands should only be issues when the SA is in measurement mode, or in the main menu system. Commands should not be issued and may not be responded to when any popup message is present, the menu system is in any sub-menus (profiles, about, etc.) or any user interface sequence (cartridge change, manual performance check, etc.)
- This protocol is designed so the RD is the master which will initiate all communications. After a command initiation there is a sequence of commands between the RD and the SA which should be completed before starting a new command sequence. Because the SA does not initiate a communications sequence it is important that the RD regularly request a status message to determine if there are pending issues which need to be addressed such as needing service due to a cartridge being empty or recommended performance checks.
- The SA will use the IP address assigned from a DHCP server on the network used, for a fixed IP address the DHCP server should be configured to always allocate a fixed IP address to the MAC address of the SA. The SA can initially be hooked up to a router or switch to initially determine the MAC address and most USB to Ethernet adaptors have the MAC address printed on the device. Options for DHCP servers include using a router with DHCP server capability, or PC options like http://www.dhcpserver.de/ which can be configured to give a consistent IP address to the SA.
    - A sample configuration file for the DHCP server for windows application (V2.5.1 – “dhcpsrv.ini”) is shown below for reference. This file shows the RD Ethernet board configured to 192.168.100.1 and sets the SA to always have an IP address of 192.168.100.2 (Change these addresses to match your specific configuration)

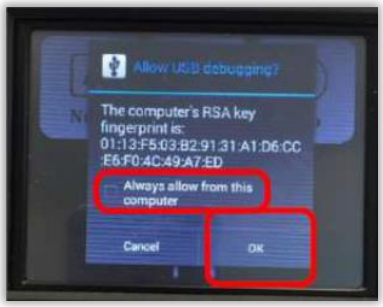
```
[General]
SUBNETMASK=255.255.255.0
LEASETIME=32000000 ; approx. 1 year
ROUTER_1=192.168.100.1 ; Network card to bind to the DHCP server
DNS_1=192.168.100.1 ; Network card to bind to the DHCP server

[Settings]
IPBIND_1=192.168.100.1 ; Network card to bind to the DHCP server
IgnoreUnknownClients=1 ; Ignore all other clients except those below

[00-50-B6-07-19-C9] ; Surface Analyst USB to Ethernet adaptor MAC Address
IPADDR=192.168.100.2 ; IP address to use when opening the port on the SA
```

- A terminal program can be used on a laptop for testing the connection. As an example, the application “Putty” can be used (https://www.putty.org/) with settings similar to the following (change address to match your specific configuration):



4. ADB through Ethernet

It is possible to send ADB commands through the ethernet connection. These commands work directly with the operating system and can sometimes be useful during troubleshooting to retrieve files. To use ADB commands through ethernet the SA device needs to already have recognized the host and received authorization to communicate with it. To do this the host needs to be hooked up to the SA using a micro USB cable directly to the SA and the host. When this connection is first made the following screen will show up on the screen and the checkbox should be checked before clicking ok. Until this step is complete the ADB commands through ethernet will return errors such as "error: device offline".



5. Recommended SA settings

To operate the SA in a remote manner there are many settings which should be changed to allow operation without operator intervention. All of these items would normally be pre-configured on a system purchased for remote operation.

- Depending on the mounting of the inspection head the image may need to be rotated for inspection head movements to match movements on the screen. MC commands for common options are "dr:270" (270 degrees), "dr:90" (90 degrees), "dr:0" (0 degrees), and "dr:180" (180 degrees). This is not required for system operation as it only impacts the orientation of the images for human readability.
- External communications mode: Menu page B, "Send to MC" option to send command "rmt" which will result in a popup "RobotMode enabled". This will disable performance check reminders as well as other popup messages. It is the responsibility of the RD to initiate performance checks on a regular basis to ensure the SA is operating accurately and properly.
- Auto login: Menu page 7, "Autologin" parameter should be set to "Enable". This will make the SA go directly to the measurement screen upon startup.
- Auto shutdown: Menu page 7, "Autoshutdown Time" parameter should be set to 0. This will keep the SA from shutting off during periods of inactivity.
- Auto dimming: Menu page B, "Send to MC" option to send command "dim:0" which will result in the popup "ScreenDimmingSeconds set to 0". This will disable the operating system automatic dimming of the screen when user input is not detected.
- Pump timeout: Menu page 6, "Pump Timeout" parameter should be set to 999. This will keep the pump pressurizing the system from timing out during periods of inactivity. It is the responsibility of the RD to turn the pump off when measurements are not required for a long period of time.
- Preview timeout: Menu page 6, "Preview Timeout" parameter should be set to 0. This will keep the system in the measurement screen without returning to the menu system during periods of inactivity. It is

the responsibility of the RD to turn the pump off when measurements are not required for a long period of time.

- SmartDrop: Menu page 5, “SmartDrop” parameter should be set to “SmartDrop”, selected for all profiles, and the limit should be appropriately set (use default if another setting is not required).
- Prime requirement reminders: Menu page 10, “Valve Prime Time” parameters (both “Single shot Time” and “Double Shot Time” should be set to 0. This will keep reminder messages from popping up on the screen. It is the responsibility of the RD to prime the system before measurements are performed when the system has been sitting idle for a period of time.
- Warm up time: Menu page 10,”Warm Up Time” parameter should be set to 0. This will disable the warm up timer during startup.
- The “Messenger” application on the SA needs to be disabled to avoid a boot crash message (while the USB to Ethernet adaptor is plugged in). This can be accomplished through the Android desktop or by running an ADB command from a connected PC (ADB is installed by Surface Analyst Manager – SAM for short).
    - C:\>adb shell pm disable com.android.mms
        - Should get echo: “Package com.android.mms new state: disabled”
- Large cartridge mode: Menu page B, “Send to MC” option to send command “lct” which will result in a popup “largeCartridge enabled”. This will set the system to use the large cartridge system and associated serial number system.
- Time for pump to come up to pressure. If the pump takes a long time to come up to pressure then errors may be generated when entering measurement mode. MC Command “rmpt:x” can be used to change the timeout value. “x” is a value in milliseconds so a value of 18000 would set the timeout to 18 seconds (appropriate for a system with slow pressurization).
- Use DIO pin 7 for purge shot. This feature is used for demonstration purposes only. Menu page B, “Send to MC” option to send command “diopst” which will return “digital IO pin 7 purge shot mode enabled” to enable this feature. Toggling the DIO pin 7 on and off (must be for duration less than 2 seconds) will trigger the device to fire a purge shot if it is already in measurement mode. This mode can be disabled with the “Send to MC” option to send command “diopsf”.
- Use DIO pass/fail and ready output signals. This optional setting is generally used for customers who do not wish to use the Ethernet communications to automate the measurement process of a Surface Analyst. This setting will use up 3 of the user configurable outputs and require that the measure input be used to acknowledge results. NOTE: This setting requires updated firmware in the Surface Analyst so older devices will need to be updated before using this function. The most common settings used with this mode are as follows:
    - External communications mode: Menu page B, “Send to MC” option to send command “rmf” which will result in a popup “RobotMode disabled”. This will enable performance check reminders as well as other popup messages. By turning off robot mode the user is accepting that all interaction with the Surface Analyst will be through the touchscreen and resemble using a Surface Analyst as a hand-held device except for the ability to trigger a measurement, receive only a pass/fail response, and having a ready signal. The angle result is only available through the visual user interface on screen.
    - Enable DIO pass/fail mode: Menu page B, “Send to MC” option to send command “iort” which will result in a popup “DIO Results Enabled”.
    - Timeout settings above may be adjusted to remind the user of items like priming, surface check card usage and automatic exit of measurement mode. These settings should be checked to make sure they are appropriate for the level of automation expected.
- By default the outgoing messages will include CrLf characters to facilitate parsing on the remote device end. This can be changed to not include CrLf by using the MC command “crlff”. To turn on CrLf use the MC command “crlft”

6. List of commands (Quick Guide)

Main Commands:	
Align>	Return coordinates of fiducial with image
AlignNP>	Return coordinates of fiducial without image
ChangeCartridge(a)>	Start a cartridge change sequence
ContinuousPurge>	Purge air out of system
DeepPurge**>**	Purge using both open flow and frequency sweeps
DeleteResults>	Remove the results database to make room in the storage of the SA
DropCount>	Get the current user drop count
FactoryPurge>	Purge air out of system
GetDropNote>	Get the current value of the Drop Note field which will be stored with the data
GetInputPin(a)>	Get the status of a DIO input pin
GetLastImage(a)>	Get a specific image type from the last inspection performed
GetLastPCHK>	Get the date and time of the last Performance Check
GetOutputPin(a)>	Get the status of a DIO output pin
GetProfiles>	Get the list of Surface Profiles stored on the SA
GetScreen>	Get an image
GetStatus>	Get current status of various items
GoToMeasurement>	Set the SA into measurement mode ready to take a measurement
LoadProfile(a)>	Change the active Surface Profile on the SA
LogLastPCHK>	Get the data from the last Performance Check
Measure>	Start a measurement with a returned image
MeasureNP>	Start a measurement without a returned image
MeasureDiscreteStart>	Start a measurement sequence broken into parts to allow movement between steps.
PCHK(a)>	Start a performance check which will verify and calibrate the SA settings
PrimeShot>	Fill the nozzle to replace evaporation loss
PurgeDropCount>	Get the current number of drops used for maintenance and purging
SetDropNote(a)>	Change the current value of the Drop Note field which will be stored with the data
SetOutputPin(a,b)>	Set the value of a DIO output pin
ShutDown>	Power down the SA
TenShotPurge>	Purge air out of system
Extra commands:	
DSP(a,b,c)>	Start a custom dispense event
GetInfo**>**	Get the current SA configuration as displayed in the About screen
GetPRS>	Get the current settings and readings of the pump pressure
Ping>	Check connectivity and responsiveness of the communications
PumpOff>	Turn off the pump (should be done when the system is not in active use)
PumpOn>	Turn on the pump
RobotDebug>	Turn on logging for the communications
RobotModeOff>	Turn off full communications to allow users to utilize the touchscreen more effectively
RobotModeOn>	Turn on full communications and disable popup messages
ScanBarcode(a)>	Read the value of a barcode without using it in a performance check
SetFan(a)>	Set the fan temperature setpoint
SetPRS(a)>	Change the pressure setting without changing profiles
TCPLoggingOff>	Turn debug Ethernet logging off
TCPLoggingOn>	Turn debug Ethernet Logging on

7. Commands for normal production use

These commands alter the settings in the SA or command it to take an action.
NOTE: All SA responses will be terminated with CrLf except for binary image data (unless changed by MC commands crlff / crlft)

Perform an alignment by taking an image and returning target location

RD command	SA responses	Description
		Normal sequence
Align> *Or* **AlignNP>**	**Align(a,b,c,d,e,f,g,h)>**	*NOTE: AlignNP> initiates a standard alignment without returning the image packet. All other responses are the same.* Successful alignment: Packet 1: a = X coordinate of the center of the circle. Result is a decimal (2 digits after decimal) location in pixels (0-511). b = Y coordinate of the center of the circle. Result is a decimal (2 digits after decimal) location in pixels (0-511). c = Area of circle. Result is an integer size in pixels. d = Size of the image in bytes (which is the size of the next packet if not the NP version). e = Number of outlier points f = Compactness value g = Timestamp of the inspection - “yyyy-mm-ddThh:mm:ss.nnn” h = Detection flags (result code indicating if the drop was automatically found to be good or defective) The possible codes are as follows: BD_OUTLIERS BD_COMPACTNESS BD_OUT_OF_FOCUS BD_DROP_TOO_SMALL GD *Example response (out of focus error):* Align(127.58,144.02,22951,284519,0,0.99,2018-05-09T15:03:52.879,BD_OUT_OF_FOCUS)> *Example response (acceptable result)* Align(256.37,280.99,23712,285723,0,1,2018-05-09T15:03:52.879,GD)>
	a	Packet 2 (only returned for an “Align>” command): a = Image from inspection. This will be a binary packet encoded in PNG image format. The image is 480x480 pixels ***NOTE: This packet will not be sent if the “AlignNP>” version of the command initiated the process.***

RD command	SA responses	Description
		Possible Failure responses
Align> *Or* **AlignNP>**	**TM_ERROR_NOT_IN_PREVIEW>**	Failed alignment: The SA is not in the measurement mode.
Align> *Or* **AlignNP>**	**ERROR_ALIGN>**	Failed alignment: Unable to locate and measure the alignment target

Prepare and initiate a cartridge change

RD command	SA responses	Description
		Normal sequence
ChangeCartridge(a)>	**CC_SERIAL_OK>**	Starts the cartridge change process by validating the new cartridge serial number. The serial number is made up of 16 HEX digits. Dashes will be removed from the received string so embedded dashes are acceptable. Either upper or lower-case letters are acceptable. a = Cartridge serial number (16 HEX digits including 0-9 and A-F) printed on cartridge.
	CC_COMPLETE>	After the user has completed the change process the SA will respond with a complete message. This message will not be sent if the serial number is invalid or the cartridge is empty or the user cancels the process (see errors below).
		Invalid serial number entered
ChangeCartridge(a)>	**CC_INVALID_SERIAL>**	The serial number entered is not valid. Verify serial number was entered correctly.
		Cartridge has already been used and is empty
ChangeCartridge(a)>	**CC_ERROR_EMPTY>**	The cartridge serial number is valid but has already been used and is empty. Use a fresh cartridge.
		Cartridge change process was cancelled by the user on the screen
ChangeCartridge(a)>	**CC_USER_CANCEL>**	During the cartridge change process, the operator has the opportunity to cancel on screen. This error indicates that has happened and so the change process has been terminated.

Start a continuous purge (Approximately 140 drops – takes a medium amount of time)

RD command	SA responses	Description
		Normal sequence
ContinuousPurge>	**ContinuousPurge>**	Response is sent once purge is complete (medium amount of time)

Start a deep purge (NOT FOR GEM DROP APPLICATIONS) (approximately 1,200 drops – takes a long time)

RD command	SA responses	Description
		Normal sequence
DeepPurge>	**DeepPurge>**	A deep purge is started. This is a medium purge and will go through several sequences including open flow and frequency sweeps. NOTE: No purge performed when a GemDrop valve is used.
	DeepPurgeFinished>	The response will be sent when the purge is successfully completed (takes a long time). NOTE: This message will never be received if used on a GemDrop application.

Delete results database to make more room on the SA

RD command	SA responses	Description
		Normal sequence
DeleteResults>	**DeleteResults>**	The database on the SA will be deleted. It is recommended that the database be backed up before deletion if any of the data needs to be analyzed or archived. The database can be remotely backed up by using the ADB command: *"C:\>adb pull /sdcard/results.db [optional destination path here]"* NOTE: The amount of free space can be inquired through the GetStatus> command to determine if deleting the results database is required.

Get current user drop count

RD command	SA responses	Description
		Normal sequence
DropCount>	**DropCount(a,b)>**	Number of drops used for measurements out of the cartridge and the total number of measurement drops the cartridge holds. a = User drops used (integer) b = User drops available (integer) *Example response (542 user drops used out of a total of 1000):* DropCount(542,1000)>

Start a factory purge (NOT FOR GEMDROP APPLICATIONS) (approximately 13,000 drops – takes a really long time)

RD command	SA responses	Description
		Normal sequence
FactoryPurge>	**FactoryPurge>**	A factory purge is started. This is a long purge and will go through several sequences including resting periods. NOTE: This will be FactoryPurgeAborted> if on a GemDrop application
	FactoryPurgeFinished>	The response will be sent when the purge is successfully completed (a really long time) NOTE: This message omitted on a GemDrop application.
		Cancel a factory purge before it completes
CancelFactoryPurge>	**CancelFactoryPurge>**	Will attempt to cancel a factory purge – cancelling can take up to 30 seconds depending on the step taking place. Final Response is sent once purge has successfully cancelled.
	FactoryPurgeAborted>	Final response is sent once the purge has been successfully cancelled.

Get the current drop note value

RD command	SA responses	Description
		Normal sequence
GetDropNote>	**GetDropNote(a)>**	This returns the current drop note. a = Current drop note. This is a text field which can contain any value a user types in.

Get status of an input bit (0-3)

RD command	SA responses	Description
		Normal sequence
GetInputPin(a)>	**GetInputPin(b,c)>**	Return the state of a physical input on the system. a = Pin number 0-3 of the input to be read The SA response contains the actual read values: b = Pin number 0-3 of the input which was read c = Status of the input. Valid status responses are: HIGH = input was read and is in a high state LOW = input was read and is in a low state ERROR_PIN = input pin number is not valid (must be 0-3) ERROR_IO = I/O board not found

Get a specific image type from the last inspection performed

RD command	SA responses	Description
		Normal sequence

RD command	SA responses	Description
GetLastImage(a)>	**GetLastImage(a,b)>**	This returns the type of image specified in the parameter from the last inspection performed. Only one type of image request is allowed for each call, multiple calls must be used to get a full set of images. A new call should not be made until the last call has completed and both packets have been returned. a = Image Type IMG_SUBTRACT = Subtract image (with no overlays) IMG_SUBTRACT_OV = Subtract image (with analysis overlay) IMG_DROP = Drop image (with no overlay except drop edges – no analysis) IMG_DROP_OV = Drop Image (with analysis overlay) IMG_SUBSTRATE = Substrate image Packet 1: a = Image Type (same value and meaning as command) b = Size of the image in bytes (which is the size of the next packet). If there is no image available then this value will be -1. *Example response:* GetLastImage(IMG_SUBTRACT,161005)>
	c	Packet 2: c = Requested image. This will be a binary packet encoded in PNG image format. The image is 480x480 pixels. NOTE: This packet will not be sent if packet 1 indicates a size of -1.

Get time and date of last performance check

RD command	SA responses	Description
		Normal sequence
GetLastPCHK>	**GetLastPCHK(a)>**	This returns the date and time of the last passed performance check a = Date and time of last performance check in the format “yyyy-mm-ddThh:mm:ss.nnn” *Example response:* GetLastPCHK(04-02-2018T14:41:57.492)>

Get status of an output bit (0-3)

RD command	SA responses	Description
		Normal sequence
GetOutputPin(a)>	**GetOutputPin (b,c)>**	Return the state of a physical output on the system. a = Pin number 0-3 of the output to be read The SA response contains the actual read values: b = Pin number 0-3 of the output which was read c = Status of the input. Valid status responses are: HIGH = output was read and is in a high state LOW = output was read and is in a low state ERROR_PIN = input pin number is not valid (must be 0-3) ERROR_IO = I/O board not found

Get a list of surface profiles available on the SA

RD command	SA responses	Description
		Normal sequence
GetProfiles>	**GetProfiles(a,b,…)>**	Returns the names of the profiles available on the SA. Each profile will be separated by a comma. The profile names are not guaranteed to be returned in any specific order. a = Name of the first profile. b = Name of the second profile. … = Rest of the profile names separated by commas NOTE: If Dynamic Detection is not enabled on the device then any profiles with Dynamic Detection enabled will not be listed in the response.

Get the current image

RD command	SA responses	Description
		Normal sequence
GetScreen>	**GetScreen(a)>**	This can be used to implement a live video stream when the SA is ready for a measurement by repeatedly sending this command. Commands cannot overlap with the responses (wait for one command to finish before starting another) so do not implement this call on a simple timer unless there is a check for the last call to be completed. Packet 1: a = Size of the image in bytes (which is the size of the next packet).
	b	Packet 2: b = Current image. This will be a binary packet encoded in PNG image format. The image is 480x480 pixels.

Request the status of the SA

RD command	SA responses	Description
		Normal sequence
GetStatus>	**GetStatus(a,b,c,d)>**	Provides a mechanism to gather information about faults which have occurred and general status. This message can also be used to prove connectivity has not been lost by polling on a regular basis. It is recommended that this be used regularly as both a test for fault conditions and as a keep alive signal to check for proper connectivity. This command should not be sent during another command sequence. a = Free space available on the SA. This will be a value between 0 and 100 and represents the percentage of available free space. This can be used to tell if the "DeleteResults>" command should be issued to free space. b = Cartridge status. Possible codes are as follows: CART_OK CART_EMPTY CART_PURGE_NEEDED c = Performance check status. Possible codes are as follows: PCHECK_OK PCHECK_DUE d = Pump status PUMP_OK PUMP_TIMEOUT *Example response:* GetStatus(53,CART_OK,PCHECK_OK,PUMP_OK)>

Change the SA into measurement mode where it is ready to accept measurement commands

RD command	SA responses	Description
		Normal sequence when SA is already in measurement mode or at the main menu
GoToMeasurement>	**GoToMeasurement>**	The SA will exit any menus which the user has placed the SA into using the touchscreen. This command works the same as the green button on the SA. The command will return once the system is in measurement mode, but may not be fully up to pressure. If a measurement is taken before full pressure is reached, the measure command will respond properly with a pressure error. NOTE: This is not meant to be used as a cancel for other modes, but will act as one if the SA is in barcode scanning, performance check, or factory purge. When used as a cancel there will be a delay during the cancel operation before the GoToMeasurement> response is sent. This command will not exit from modal popup windows or menus. NOTE: If the login screen is showing when this command is issued, the default user will be logged in and the system we move to the measurement screen (functionally the same as setting the auto login feature)

Change active surface profile on the SA

RD command	SA responses	Description
		Normal sequence
LoadProfile(a)>	**LoadProfile>**	Changes the active profile to the profile specified by “a” in the RD command. a = Name of the profile to switch to. NOTE: The profile names are case sensitive and must match exactly.
		Failure response
LoadProfile(a)>	**LoadProfileNotFound>**	If the active profile cannot be changed to “a” then this response is given instead of the response above. The profile is not changed.
LoadProfile(a)>	**LoadProfileDynamicDetectionLocked>**	If the device does not have Dynamic Detection enabled, but the profile “a” has Dynamic Detection enabled, then the profile will not be loaded. The profile is not changed.

Get detailed information about the last performance check card readings

RD command	SA responses	Description
		Normal sequence
LogLastPCHK>	**LogLastPCHK(a)>**	Returns the values from the last performance check card sequence. This command is not required to accomplish a performance check sequence – the data provided is only provided for logging or further data display and analysis. This command can be sent during a performance check sequence to check the status of a sequence after a PCHK_ADJUSTED_CONTINUE> response is received. If this command is not issued before the completion of the follow-up performance check then the data will be over written by the next sequence. Each type of performance check result can result in different formats of returned data so the “a” placeholder in the response could follow the format of any of the response types below. NOTE: If only 4 measurements are used for the mean and standard deviation then the unused contact angle will be in parenthesis. *Example response (Successful completion):* LogLastPCHK(PCHK_PASSED_STOP,2018-05-02T15:59:44.878,Angles: 79.0, 80.0, 75.0, 81.0, 77.0, Mean: 78.4, StDev: 2.2)> *Example response (Adjustment made):* LogLastPCHK(PCHK_ADJUSTED_CONTINUE,2018-05-02T15:56:59.204,Angles: 91.0, 93.0, 91.0, 87.0, 86.0,meanVal: 89.6, diffAngle: 12.6, theorDegOT: 3.15, OTdelta: 1, actDegOT: 2.5, actPdeg: 10.1, Pdelta: 0.77, OTold: 30, OTnew: 31, pOld: 5.7, Pnew: 6.47, setValvePressure/valveOpenTime: 5.9/34, Mean: 89.6, StDev: 2.7)> *Example response (Standard deviation fault):* LogLastPCHK(PCHK_FAILED_STD_DEV_STOP,2018-05-02T15:54:29.319,Angles: 92.0, 81.0, 91.0, 91.0, 86.0, Mean: 88.2 StDev: 4.2)> *Example response (Under limits fault):* LogLastPCHK(PCHK_UNDER_LIMITS_STOP,2019-04-18T14:45:28.001,Angles: (85), 77, 74, 73, 76, meanVal: 75, diffAngle: -20, theorDegOT: -5, OTdelta: -2, actDegOT: -5, actPdeg: -15, Pdelta: -1.15, OTold: 33, OTnew: 31, pOld: 5.54, Pnew: 4.39, Mean: 75, StDev: 1.6)> *Example response (Over limits fault):* LogLastPCHK(PCHK_OVER_LIMITS_STOP,2019-04-18T15:13:19.443,Angles: (70), 62, 61, 61, 60, meanVal: 61, diffAngle: 20, theorDegOT: 5, OTdelta: 2, actDegOT: 5, actPdeg: 15, Pdelta: 1.15, OTold: 36, OTnew: 38, pOld: 6.5, Pnew: 7.65, Mean: 61, StDev: 0.7)>

Take a measurement and return the results

RD command	SA responses	Description
		Normal sequence
Measure> *or* **MeasureNP>**	**Measure(a,b,c,d,e,f,g,h,i)>**	*NOTE: MeasureNP> initiates a standard measurement without returning the image packet. All other responses are the same.* Successful measurement Packet 1: The results of the measurement which consists of the following: a = Measured contact angle (This will be set to 999 for measurement failures) b = Outlier points (number of outliers if turned on, if turned off this will be 0) c = Compactness value of the drop (decimal number <=1) d = Drop center distance (integer distance from cross hair location to drop found location) e = Timestamp (internal time of measurement) - “yyyy-mm-ddThh:mm:ss.nnn” f = Drop count (number of measurement drops used out of the cartridge) g = Detection flags (result code indicating if the drop was automatically found to be good or defective) The possible codes are as follows: BD_OUTLIERS BD_COMPACTNESS BD_OUT_OF_FOCUS BD_DROP_TOO_SMALL GD h = Pass Fail Flag (status of the inspection using the internal pass/fail limits. The code is a single character.) The possible codes are as follows: P (pass) F (fail) S (fail, surfactant detected) N (none, limits set to 0-180 which disables limits) i = Size of the image in bytes (which is the size of the next packet if not the NP version). *Example response (Failed detection due to outliers and failed measurement):* Measure(999,40,0.93,62,2018-05-03T15:32:05.327,251,BD_OUTLIERS,F,153815)> *Example response (Good detection but failed measurement):* Measure(58,0,0.94,9,2018-05-03T15:31:49.972,250,GD,F,160560)> *Example response (Good detection passing measurement):* Measure(52,6,0.96,9,2018-05-03T15:40:31.011,256,GD,P,161005)>

RD command	SA responses	Description
	a	Packet 2 (only returned for a “Measure>” command): a = Image from inspection. This will be a binary packet encoded in PNG image format. The image is 480x480 pixels. ***NOTE: This packet will not be sent if the “MeasureNP>” version of the command initiated the process.***
		Possible Failure responses
Measure> *or* **MeasureNP>**	**TM_ERROR_PUMP_RAMPING>**	Failed measurement: System has not reached pressure yet, measurement was taken too soon after entering measurement mode or system is taking a long time to reach pressure. This response happens if a measure is requested while the SA is displaying the black pressurizing screen.
Measure> *or* **MeasureNP>**	**TM_ERROR_PRESSURE:XXXX>**	Failed measurement: The system is not at the proper pressure. The “XXXX” is replaced with the current pressure value.
Measure> *or* **MeasureNP>**	**TM_ERROR_NOT_IN_PREVIEW>**	Failed measurement: The SA is not in the measurement mode.
Measure> *or* **MeasureNP>**	**TM_ERROR_OVER_DROP_COUNT>**	Failed measurement: The SA is unable to take a measurement because the cartridge is out of liquid. The cartridge must be replaced before measurements can be taken.
Measure> *or* **MeasureNP>**	**TM_ERROR_CART_PURGE_NEEDED>**	Failed measurement: The SA is unable to take a measurement because the system requires purging (typically after a cartridge change). Use ContinuousPurge> before attempting again.
Measure> *or* **MeasureNP>**	**TM_ERROR_DB_TRANSFER>**	Failed measurement: The SA is unable to take a measurement because the system is currently transferring the result database to the RD.

Take a measurement (broken into discrete steps) and return the results

RD command	SA responses	Description
		Normal sequence
MeasureDiscreteStart>	**SubstrateCaptured>**	Starts the sequence for a measurement which is broken up into individual steps. If the timing is held the same as a standard measurement then this command will achieve the same results as a standard “Measure>” or “MeasureNP>” command while allowing an automatic system to take extra actions in-between steps. Response indicates that the substrate image has been taken and system is ready for movement if needed once response is received. ***NOTE: The normal imaging and dispensing cycle includes specific delays and pauses between steps setup in the Archer software. When using this feature the timing and delays are no longer controlled by the Surface Analyst and must be properly controlled by the Remote Device.***
MeasureDropDispense>	**DropDispensed>**	Initiate a drop dispense using the appropriate settings for the current profile. Response once drop has been dispensed and the system is ready for movement if needed.
MeasureInspect> *or* **MeasureInspectNP>**	**DropCaptured>**	Trigger the inspection of the drop which was just dispensed. The response indicates the drop image has been taken and the system is ready for movement if needed. *MeasureInspect> results in 3 response packets while MeasureInspectNP> results in only 2 packets.* Packet 1: drop image has been captured and system is ready for movement if needed. ***NOTE: While using Surfactant detection mode this command will initiate two images being taken with a delay between them. The delay can be set in the Surfactant detection settings of the SA.*** ***NOTE: MeasureInspectNP> initiates a measurement without returning the image packet. All other responses are the same.***

RD command	SA responses	Description
	Measure(a,b,c,d,e,f,g,h,i)>	Successful measurement Packet 2: The results of the measurement which consists of the following: a = Measured contact angle (This will be set to 999 for measurement failures) b = Outlier points (number of outliers if turned on, if turned off this will be 0) c = Compactness value of the drop (decimal number <=1) d = Drop center distance (integer distance from cross hair location to drop found location) e = Timestamp (internal time of measurement) - “yyyy-mm-ddThh:mm:ss.nnn” f = Drop count (number of measurement drops used out of the cartridge) g = Detection flags (result code indicating if the drop was automatically found to be good or defective) The possible codes are as follows: BD_OUTLIERS BD_COMPACTNESS BD_OUT_OF_FOCUS BD_DROP_TOO_SMALL GD h = Pass Fail Flag (status of the inspection using the internal pass/fail limits. The code is a single character.) The possible codes are as follows: P (pass) F (fail) S (fail, surfactant detected) N (none, limits set to 0-180 which disables limits) i = Size of the image in bytes (which is the size of the next packet if not the NP version). *Example response (Failed detection due to outliers and failed measurement):* Measure(999,40,0.93,62,2018-05-03T15:32:05.123,251,BD_OUTLIERS,F,153815)> *Example response (Good detection but failed measurement):* Measure(58,0,0.94,9,2018-05-03T15:31:49.937,250,GD,F,160560)> *Example response (Good detection passing measurement):* Measure(52,6,0.96,9,2018-05-03T15:40:31.041,256,GD,P,161005)>

RD command	SA responses	Description
	a	Packet 3 (only returned for a "MeasureInspect>" command): a = Image from inspection. This will be a binary packet encoded in PNG image format. The image is 480x480 pixels. ***NOTE: This packet will not be sent if the "MeasureInspectNP>" version of the command initiated the process.***
		Possible Failure responses
MeasureInspect > *or* **MeasureInspectNP >**	**TM_ERROR_PUMP_RAMPING>**	Failed measurement: System has not reached pressure yet, measurement was taken too soon after entering measurement mode or system is taking a long time to reach pressure. This response happens if a measure is requested while the SA is displaying the black pressurizing screen.
MeasureInspect > *or* **MeasureInspectNP >**	**TM_ERROR_PRESSURE:XXXX>**	Failed measurement: The system is not at the proper pressure. The "XXXX" is replaced with the current pressure value.
MeasureInspect > *or* **MeasureInspectNP >**	**TM_ERROR_NOT_IN_PREVIEW>**	Failed measurement: The SA is not in the measurement mode.
MeasureInspect > *or* **MeasureInspectNP >**	**TM_ERROR_OVER_DROP_COUNT>**	Failed measurement: The SA is unable to take a measurement because the cartridge is out of liquid. The cartridge must be replaced before measurements can be taken.
MeasureInspect > *or* **MeasureInspectNP >**	**TM_ERROR_CART_PURGE_NEEDED>**	Failed measurement: The SA is unable to take a measurement because the system requires purging (typically after a cartridge change). Use ContinuousPurge> before attempting again.

Initiate a performance check sequence using printed performance check surface card. Because of the complexity of the performance check sequence a flow chart of possible operations and responses is included in Appendix A.

RD command	SA responses	Description
		Normal sequence
PCHK(a)>	**PCHK>**	Performance check has been started. a = Timeout in seconds. Enter 0 to disable timeout. The timeout starts timing once the barcode scanner has started reading the image. The total time for this command to fully timeout is longer to account for the period of time when the camera is switching modes to prepare for barcode scanning. Minimum time recommended is 2 seconds to ensure a good read. Packet 1: Camera needs to be over the barcode for proper reading when this command is issued.
	ScanOK(a)>	Packet 2: Barcode has been read. a = Data encoded into the barcode. The format of the data encoded on normal performance check cards is: Angle,Tolerance,Deviation,NumberOfTests, degreesPerDroplet,Version,Lot,Expiration *Example response:* ScanOK(71,02,02.5,05,02.4,00.13,161202,1701)>
	PCHK_CAM_READY_1>	Packet 3: The SA is ready for the system to move to the first measurement location on the performance check surface card.
Measure> *Or* **MeasureNP>**	**1 or 2 standard measure response packets. See measure response above for details**	Response is identical to a normal measurement response. See measurement command above for details on response format. If the measure is unsuccessful this command will need to be repeated.
	PCHK_CAM_READY_2>	Indicates that the above measurement command was successful and the SA is ready for the next measurement.
Measure> *Or* **MeasureNP>**	**1 or 2 standard measure response packets. See measure response above for details**	Response is identical to a normal measurement response. See measurement command above for details on response format. If the measure is unsuccessful this command will need to be repeated.
	PCHK_CAM_READY_3>	Indicates that the above measurement command was successful and the SA is ready for the next measurement.
Measure> *Or* **MeasureNP>**	**1 or 2 standard measure response packets. See measure response above for details**	Response is identical to a normal measurement response. See measurement command above for details on response format. If the measure is unsuccessful this command will need to be repeated.
	PCHK_CAM_READY_4>	Indicates that the above measurement command was successful and the SA is ready for the next measurement.

RD command	SA responses	Description
Measure> *Or* **MeasureNP>**	**1 or 2 standard measure response packets. See measure response above for details**	Response is identical to a normal measurement response. See measurement command above for details on response format. If the measure is unsuccessful this command will need to be repeated.
	PCHK_CAM_READY_5>	Indicates that the above measurement command was successful and the SA is ready for the next measurement.
Measure> *Or* **MeasureNP>**	**1 or 2 standard measure response packets. See measure response above for details**	Response is identical to a normal measurement response. See measurement command above for details on response format. If the measure is unsuccessful this command will need to be repeated.
	PCHK_PASSED_STOP>	Passing result: The performance check sequence has completed successfully, any adjustments made were minor and do not require a re-test.
Individual measurement bad flag response sample sequence – repeat of measurement required.		
. . . . . .	**PCHK_CAM_READY_1>**	Example for measurement 1, this example can be substituted for any measurement 1-5. Standard sequence contains the SA response to take measurement #1 which causes the RD to initiate a measurement.
Measure> *Or* **MeasureNP>**	**Measure(?,?,?,?,?,?, BD_OUTLIERS,?,?)>** *Or* **Measure(?,?,?,?,?,?,BD_COMPACTNESS,?,?)>** *Or* **Measure(?,?,?,?,?,?,BD_OUT_OF_FOCUS,?,?)>** *Or* **Measure(?,?,?,?,?,?,BD_DROP_TOO_SMALL,?,?)>**	The SA responded with the measure response but the flags indicate that the drop did not meet the acceptance criteria and was rejected. See measurement command above for full explanation of the fields and their meanings for the measurement return message. ***NOTE: Any flag response other than "GD" will result in the sequence repeating.***
	PCHK_CAM_READY_1>	The same measurement needs to be repeated, so the SA will send the same response as above in this example causing the sequence to repeat until a good measurement is taken.
Individual measurement failure response sample sequence – repeat of measurement required		
. . . . . .	**PCHK_CAM_READY_1>**	Example for measurement 1, this example can be substituted for any measurement 1-5. Standard sequence contains the SA response to take measurement #1 which causes the RD to initiate a measurement.
Measure> *Or* **MeasureNP>**	**TM_ERROR_*?????*>**	The SA responded with a failure message instead of the normal measurement response packets. See measurement command above for possible failure responses.
	PCHK_CAM_READY_1>	The same measurement needs to be repeated, so the SA will send the same response as above in this example causing the sequence to repeat until a good measurement is taken.

RD command	SA responses	Description
Adjustment and re-measure required – on last measurement request the response is changed and a new sequence is started		
Measure> *Or* **MeasureNP>**	**PCHK_ADJUSTED_CONTINUE>**	Passing result with adjustment: Adjustments have been made and another set of measurements need to be taken. The performance check sequence is started over, and the SA issues the response saying it is ready for the first measurement
	PCHK_CAM_READY_1>	Repeat the measurement sequence above on a new performance check surface.
Failure response due to lack of fluid. The sequence is terminated.		
PCHK(a)>	**PCHK_ERROR_CART_EMPTY>**	The fluid cartridge is empty and needs to be replaced. The performance check sequence is terminated.
Failure response due to scanning timeout. The sequence is terminated.		
PCHK(a)>	**PCHK>**	Immediate response to the performance check with a timeout on the barcode scan set to a value that is not long enough for the read.
	ScanTimeout>	After the timeout has elapsed without reading a barcode this response will be sent.
Failure sequence when the barcode is not from a performance check surface		
PCHK(a)>	**PCHK>**	Example request for performance check with a timeout of “a” second(s) with the immediate response indicating acceptance of the command.
	ScanCardInvalid(b)>	The barcode was able to be read but is not of the proper format for a performance check surface. b = Data encoded into the barcode. *Sample response:* ScanCardInvalid(https://btglabs.com)>
Failure sequence when the barcode indicates that the card has expired		
PCHK(a)>	**PCHK>**	Example request for performance check with a timeout of “a” second(s) with the immediate response indicating acceptance of the command.
	ScanCardExpired(b)>	The barcode was able to be read and is of the proper format for a performance check surface but the card has expired. b = Data encoded into the barcode. The format of the data encoded on a performance check card is: Angle,Tolerance,Deviation,NumberOfTests, degreesPerDroplet,Version,Lot,Expiration *Sample response:* ScanCardExpired(77,02,02.8,05,02.4,01.00,170601,1806)>
Failure sequence – on last measurement request the response is changed to one of the below. The sequence is terminated.		
Measure> *Or* **MeasureNP>**	**PCHK _FAILED_STD_DEV_STOP>**	Failed result: This will occur after the 5 inspections have completed and there was too much variation in the drops. A purge is recommended before restarting the performance check.

RD command	SA responses	Description
Measure> *Or* **MeasureNP>**	**PCHK _FAILED_OVER_LIMITS_STOP>**	Failed result: This will occur after the 5 inspections have completed and the adjustments required push the settings over the upper limits. A purge is recommended before restarting the performance check.
Measure> *Or* **MeasureNP>**	**PCHK _FAILED_UNDER_LIMITS_STOP>**	Failed result: This will occur after the 5 inspections have completed and the adjustments required push the settings below the lower limits. A purge is recommended before restarting the performance check.
		Cancel request from the RD during the above sequences. This terminates the sequence
CancelPCHK>	**CancelPCHK>**	Sequence is terminated and no further adjustments are made.

Start a prime shot (1 drop) used before a dispense sequence to fill the nozzle with liquid

RD command	SA responses	Description
		Normal sequence
PrimeShot>	**PrimeShot>**	This purge is extremely short so will return almost immediately – response is sent once purge is complete.

Get current maintenance drop count

RD command	SA responses	Description
		Normal sequence
PurgeDropCount>	**PurgeDropCount(a,b)>**	Number of drops used for maintenance out of the cartridge. Maintenance drops are used by purge, performance check, and prime. Once all maintenance drops are used then maintenance operations start using measurement drop counts. Also included are the total number of maintenance drops the cartridge holds. a = Maintenance drops used (integer) b = Maintenance drops available (integer) *Example response (123 maintenance drops used out of a total of 1000):* PurgeDropCount(123,1000)>

Set the drop note to a new value

RD command	SA responses	Description
		Normal sequence
SetDropNote(a)>	**SetDropNote>**	This sets the drop note to the value specified. a = Value of the new drop note. This is a text field.

Set status of an output bit (0-3)

RD command	SA responses	Description
		Normal sequence
SetOutputPin(a,b)>	**SetOutputPin(c,d)>**	Set the state of a physical output on the system. a = Pin number 0-3 of the output to be set. ***NOTE: When DIO result mode has been enabled the first three pins (0-2) are used for the ready, pass, and fail outputs and cannot be used for general outputs (0=Ready, 1=Pass, 2=Fail).*** b = Desired state for the output. Valid states are HIGH and LOW The SA response contains the results of the operation. c = Pin number 0-3 of the output which was set d = Status of the output. Valid status responses are: HIGH = output was written and is in a high state LOW = output was written and is in a low state ERROR_PIN = output pin number is not valid (must be 0-3) ERROR_IO = I/O board not found

Shutdown the SA

RD command	SA responses	Description
		Normal sequence
ShutDown>	**ShutDown>**	This powers the SA down.

Start a 10 shot purge (Approximately 10 drops – short amount of time)

RD command	SA responses	Description
		Normal sequence
TenShotPurge>	**TenShotPurge>**	Response is sent once purge is complete. There will be a short time delay between sending the command and receiving the response which indicates completion. The timeout for this command should be appropriately long to not time out during the purge.

8. Commands for troubleshooting and debugging use

These commands are more unusual and typically not needed to implement a standard interface.
NOTE: All SA responses will be terminated with CrLf except for binary image data.

Dispense a custom drop

RD command	SA responses	Description
		Normal sequence
DSP(a,b,c)>	**DSP>**	Dispense a drop with custom parameters (Echo response is immediate): a = Number of droplets - small droplets that make up the larger drop. Standard dispense uses 19 droplets b = Valve open time (in processor ticks which are 30 microseconds long) – how long the valve will be open for each droplet. Normally in the 32 to 35 tick range. c = Valve Period (in processor ticks which are 30 microseconds long) – time between each valve opening (this must be longer than the valve open time). Normal value is around 300 ticks.
	DSP_Complete>	Response once the dispense has completed.

Get configuration information contained in the SA About Screen

RD command	SA responses	Description
		Normal sequence
GetInfo>	**GetInfo(a)>**	This returns the current configuration values in the SA as displayed in the about screen. The information provided through this call includes information about the currently loaded profile, keys describing the profile can be different between different profiles so it should not be assumed that all keys will always be present. Note: Special characters are stripped out of the data stream. a = returned information from the about screen formatted with descriptors and separators. *Example response (from a testing SA, information will be different depending on configuration):* GetInfo(Serial Number: A3340,Software version: Archer 8.12 Beta 79,Firmware version: 1.30,Transducer setting: 0001,OS build: 20161213.115117,Device IP: 192.168.100.2,Available Memory: 11.3 GB,Total Memory: 28.9 GB,Available Heap: 314 MB,Total Heap: 384 MB,Battery: 8.28 V,User: BTGLabs,Surface Profile: defaukt,User Drop remaining: 760,Purge Drop used: 88,Autologin: Disabled,Drop Note: ,Min Pass Angle: 65,Max Pass Angle: 180,Warn Limit: 90,Detection Accept/Reject: Auto,SmartDrop Limit: 0.4,Number of Outliers Pass Limit: 99,Continuous Outliers Pass Limit: 96,Time: 09-17-2020 15:17:51,Time zone: Eastern Standard Time,Cartridge Serial #: test - August 21 2020,Calibration Due: September 17 2021,Days Since Performance Check: 0,*Drop Dispense Parameters*,Drop settle time: 0.35 seconds,Pressure: 4.64,Pressure Tolerance: 5 %,Droplets per Drop: 19,Valve spike time: 17,Valve open time: 32,Valve period: 300,Drop Mass: 1.5,Extended Purge #: 0,Continuous Purge Time: 0,Quick Purge Shots: 10,*Analysis Parameters*,Dynamic Detection: Enabled,Drop center: Auto,Crosshair position: [50%|50%],Outlier Rejection: Enabled,Outlier % Diff. Threshold: 6,Ellipse Mode: Disabled,Image Alignment: Disabled,Image Alignment Eps: 1.0E-5,Image Alignment Max Count: 25,Invert Finding: Disabled,Edge Width Filter: 15,Edge Length Filter: 9,Center Min Diameter: 15,Center Max Diameter: 370,Center Merge All: Disabled,Center Merge Distance: 10,Multiplier: Pass 1 Near: 0.25,Multiplier: Pass 1 Far: 2,Multiplier: Pass 2 Near: 0.5,Multiplier: Pass 2 Far: 2,Multiplier: Pass 3 Near: 0.8,Multiplier: Pass 3 Far: 1.72,Dyne Mode: Disabled,Dyne Parameters: 0.0|0.0138|-1.9036|96.897,Surfactant Detection: Disabled,Surfactant Overall Time: 3 s,Surfactant Image Interval: 5,Surfactant Delta: 0,*Optical Parameters*,Illumination: 25,Exposure: 0,Optical Cal: 24305 pixels,DiamCorrectB: 1.01,*Purchasable Options*,Unlock All: Enabled)>

Read pump pressure setpoint and actual value (values are in PSI)

RD command	SA responses	Description
		Normal sequence
GetPRS>	**GetPRS(a,b)>**	Returns both the setpoint for the pressure and the current pressure as read by the SA. a = The set pressure in PSI as a decimal number with up to 2 significant digits. b = The actual pressure in PSI as a a decimal number with up to 2 significant digits. *Example response (pressure is set to 3 PSI):* GetPRS(3,2.94)>

Verify SA is alive and responsive

RD command	SA responses	Description
		Normal sequence
Ping>	**Ping>**	Provides a method to prove connectivity has not been lost without commanding an action. This should normally be replaced by the GetStatus> command which also returns the status of the device giving information on important faults and status which should be handled by the RD. Use Ping> for troubleshooting connectivity.

Turn the pump off

RD command	SA responses	Description
		Normal sequence
PumpOff>	**PumpOff>**	Turns the pump off which will depressurize the cartridge.

Turn the pump on

RD command	SA responses	Description
		Normal sequence
PumpOn>	**PumpOn>**	Turns the pump on which will pressurize the cartridge.

Enable debug message mode

RD command	SA responses	Description
		Normal sequence
RobotDebug>	**RobotDebug>**	Enables debugging mode where the SA will display some messages sent and received as toast messages. There is no command to turn this mode off – it will automatically turn off when the device is powered off. ***NOTE: This does not echo all commands and responses as toast messages but does major ones such as Ping>, Align>, Measure>, GetProfiles>, PurgeDropCount>, DropCount>.***

Disable robot mode functionality

RD command	SA responses	Description
		Normal sequence
RobotModeOff>	**RobotModeOff>**	Changes the status of the RobotMode flag in the SA so the system will allow user interaction and popup messages from the touchscreen. This feature should not normally be turned off when the SA is operated remotely – this is provided for maintenance operations where a user needs to interact with the SA touchscreen in a more “hand held” manner. ***NOTE: Full normal functionality can only be achieved after a reboot of the SA. Certain features will work immediately, but others may not function properly until after the reboot.***

Enable robot mode functionality

RD command	SA responses	Description
		Normal sequence
RobotModeOn>	**RobotModeOn>**	Changes the status of the RobotMode flag in the SA so the system will disable popup messages and other behaviors that interfere with remote operations. This feature should normally always be on when a SA is operated remotely.

Scan a performance check barcode without doing a performance check

RD command	SA responses	Description
		Normal sequence
ScanBarcode(a)>	**ScanBarcode>**	Starts the barcode scanner reading the barcode. When the barcode has been successfully read the results will be returned (unless a cancel command is issued first) a = Timeout in seconds. Enter 0 to disable timeout. The timeout starts timing once the barcode scanner has started reading the image. The total time for this command to fully timeout is longer to account for the period of time when the camera is switching modes to prepare for barcode scanning. Minimum time recommended is 2 seconds to ensure a good read.
	ScanOK(b)>	Results of the barcode scan. b = Data encoded into the barcode. The format of the data encoded on a performance check card is: Angle,Tolerance,Deviation,NumberOfTests, degreesPerDroplet,Version,Lot,Expiration *Sample response:* ScanOK(71,02,02.5,05,02.4,00.13,161202,1701)>
		Failure sequence when the scan times out
ScanBarcode(a)	**ScanBarcode>**	Example request for barcode scanning with a timeout of “a” second(s) with the immediate response indicating acceptance of the command.
	ScanTimeout>	After the timeout period has expired the timeout message is sent.
		Failure sequence when the barcode is not from a performance check surface
ScanBarcode(a)	**ScanBarcode>**	Example request for barcode scanning with a timeout of “a” second(s) with the immediate response indicating acceptance of the command.
	ScanCardInvalid(b)>	The barcode was able to be read but is not of the proper format for a performance check surface. b = Data encoded into the barcode. *Sample response:* ScanCardInvalid(https://btglabs.com)>

RD command	SA responses	Description
		Failure sequence when the barcode indicates that the card has expired
ScanBarcode(a)	**ScanBarcode>**	Example request for barcode scanning with a timeout of "a" second(s) with the immediate response indicating acceptance of the command.
	ScanCardExpired(b)>	The barcode was able to be read and is of the proper format for a performance check surface but the card has expired. b = Data encoded into the barcode. The format of the data encoded on a performance check card is: Angle,Tolerance,Deviation,NumberOfTests, degreesPerDroplet,Version,Lot,Expiration *Sample response:* ScanCardExpired(77,02,02.8,05,02.4,01.00,170601,1806)>
		Cancel scanning before it completes
CancelBarcode>	**CancelBarcode>**	Allows the RD to cancel a barcode scanning session. An echo packet is sent immediately.
	ScanCancelled>	Once the SA has exited scanning mode the ScanCancelled packet is sent indicating the barcode reading has been successfully cancelled. Cancelling can take several seconds.

Set fan temperature – Change or inquire about fan setpoint

RD command	SA responses	Description
		Normal sequence
SetFan(a)>	**SetFan(b)>**	Sets the integrated fan temperature setpoint (degrees Fahrenheit). a = The new target temperature in degree Fahrenheit as an integer. Use -1 to query the current fan setpoint. b = The value the fan is set to after the command executes. Will return -1 for a system without fan control. *Example command to set the fan temperature to 100 degrees F:* SetFan(100)> ***NOTE: Sending -1 into the command will query the current value / state of the fan.*** ***NOTE: If there is no fan present on the system then the command will return -1.*** ***NOTE: A value of 0 will turn the fan off and a value of 999 keeps the fan on all the time.***

Set pump pressure - Pressures are in PSI

RD command	SA responses	Description
		Normal sequence
SetPRS(a)>	**SetPRS>**	Sets the target pressure (PSI) to the value contained in the command. a = The new target pressure in PSI as a decimal number with up to 2 significant digits *Example command to set the pump pressure to 4.5 PSI:* SetPRS(4.5)> ***NOTE: This command does not wait until the target pressure is reached before returning.*** ***NOTE: This command does not change the pressure stored in the surface profile, switching surface profiles will return the pressure to the stored value for the surface profile.*** ***NOTE: Prime and Purge commands can change the pressure if the set pressure is too low to perform the function. If the pressure was changed, then the pressure will return (after the purge) to the value stored in the active surface profile.***

Ethernet message logging - Disable

RD command	SA responses	Description
		Normal sequence
TCPLoggingOff>	**TCPLoggingOff>**	Disables logging of the Ethernet messages to the “log_tcp.db” database file.

Ethernet message logging - Enable

RD command	SA responses	Description
		Normal sequence
TCPLoggingOn>	**TCPLoggingOn>**	Enables logging of the Ethernet messages to the “log_tcp.db” database file. ***NOTE: Logging will be automatically disabled on a power cycle of the Surface Analyst to reduce the risk of accidentally leaving this feature on and filling the storage space with the log database.***

Transfer database:

Connecting to the SA over TCP port 2223 will automatically stream the contents of the database. This port can be changed if needed with the MC command “tcpr:x”, contact BTG Labs for assistance.
The packets need to be assembled and stored as a database file to be useable. There is no RD command to initiate transfer, the connection will initiate the transfer and when the transfer is complete the connection should be closed.
This connection will transfer all databases available on the device.

NOTE: The database can also be remotely backed up by using the ADB command:
“C:\>adb pull /sdcard/results.db [optional destination path here]”

NOTE: If there are multiple databases on the SA then this sequence will repeat without delays between them until all databases have been sent. A timeout can be implemented on the RD to determine when all the databases have been received.

RD command	SA responses	Description
		Normal sequence
	*****	A single packet containing: 4 byte int - length of filename followed by the filename (will be of format "UnitSerialNumber_yyyy_mm_ddTHH_mm_ss_results_1.db" where the timestamp values are the current SA3001 clock time and the UnitSerialNumber is similar to 'A3332') data of filename is UTF8 string. Multiple database files have the trailing number incremented.
	*****	Stream of packets that contain the following information: 4 byte int which will == -2 to signify start of streaming 8 byte int length Data of length (results.db on device) 8 byte Adler32 checksum ***NOTE: Data is sent little-endian*** ***If the person implementing this doesn't have an Adler32 library for their platform they can just ignore that last 8 bytes.*** ***Sample VB code to demonstrate functionality is available from BTG Labs (AR-239).***
		Failure sequence – Connection does not result in database transfer.
	ERROR_MEASUREMENTS_SAVING	This error is returned if the system is currently still saving measurement results. The connection can be retried after a delay to allow measurements to be saved to the database.

9. Appendix A

Flow chart for the possible paths and outcomes of a performance check sequence when a Surface Analyst is in robot mode and commanded to execute a performance check. See PCHK> for commands associated.