On-Board IS Cryopump With Sublime Regeneration Command Set Reference

8040744 Revision B

For Technical Support:

Location	GUTS® Contact Number
North America	+1-800-FOR-GUTS (1-800-367-4887) +1-978-262-2900
Europe	+49-1804-CALL-GUTS (+49-1804-2255-4887)
Japan	+81-45-477-5980
China	+86-21-5131-7066
Taiwan	+886-3-5525225
Korea	+82-31-288-2500
Singapore	+65-6464-1481

Visit us online: www.brooks.com

January 11, 2013 Part Num 8040744 Revision B

Contents

This Page Intentionally Left Blank

Figures

This Page Intentionally Left Blank

Tables

This Page Intentionally Left Blank

Revisions

Overview

Changes may be made or additional documents or drawings added to the manual at any time. This section provides a brief description of each change.To identify the current revisions or to obtain a current set of drawings and documents, contact Brooks Automation Technical Support.

Rev. 0

Initial release.

Rev. 1

Added information to the section Message Checksum Character on page 1-5. Changed ASCII Check sum to b in first table under the paragraph Example of Host - Controller Communication on page 1-9. Added information to Regeneration Response (o Command) Description on page 2-13. Added information to Command B in Table 3-1 On-Board IS Cryopump Commands on page 2-2. Updated information in Command Y of Table 3-1 On-Board IS Controller Commands on page 3-2. Updated ASCII Response Character in j Command Example 1 on page 3-15. Added information to commands O and P in Table 4-1 On-Board IS 1000 Compressor Commands on page 4-2. Added Command Y to Table 4-1 On-Board IS 1000 Compressor Commands on page 4-2. Added Changes chapter to the manual.

This Page Intentionally Left Blank

1 Introduction to On-Board *IS* System Commands

Overview

You can integrate the On-Board *IS* System into your current system through specific commands and queries (messages). This section describes the structure of this document, and the format of the messages you send and the responses you receive from system components.

Chapter Contents

About the Format of this Reference

You can send messages from your computer device (host) to an On-Board *IS* System Component (slave or component), and receive responses from the component. The commands and responses appear in the following sections for each component:

- **On-Board *IS* Two Stage Cryopump with Sublime Regeneration**. See On-Board IS Cryopump Commands on page 2-1.
- **On-Board *IS* Controller**, see On-Board IS Controller Commands on page 3-1.
- **On-Board *IS* Compressor**, see On-Board IS Compressor Commands on page 4-1.

Your system may include one, some, or all of these components. If you are not sure which components you have, contact Customer Support (see Appendix A: Customer Brooks Automation Technical Support Information on page 5-2).

The remaining parts of this section explain the communication structure between the host and On-Board *IS* System Components.

Hardware Requirements for System Communication

An On-Board *IS* System Component transmits responses at a rate of 9600 baud. You can use a DB-9/DB-25 cable or a DB-9/DB-9 cable to connect the host and slave.

NOTE: *Use metal case type connectors on cables that are less than 40 feet long.*

WARNING

Equipment Damage

To avoid excessive EMI conditions, do not route the RS-232 cable with other power cables.

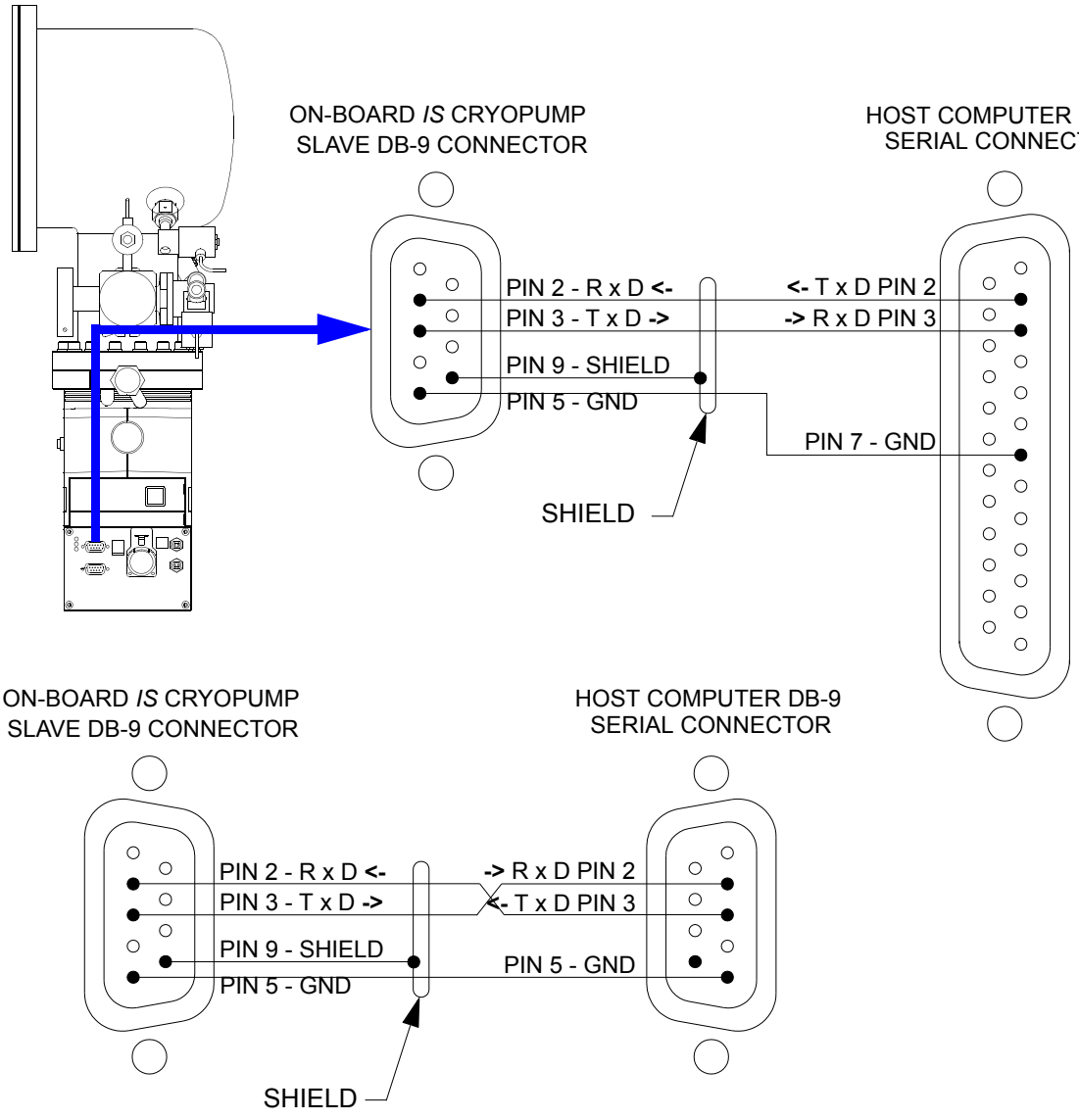

Figure 1-1: On-Board IS Cryopump RS-232 Cable Connector Pin Assignments

About Host and Component Communication

Communication originates at the host unit, as a message packet, containing a command or query. Commands cause specific actions to occur in the On-Board *IS* System Component (slave or component). Queries request that the On-Board *IS* System Component reply with status or other parametric information. The component responds to all such correctly received messages.

All communications between a host computer and the On-Board *IS* System Component occurs within this message packet framework. Software operating on the host computer must generate and interpret the command-response pairs to properly execute and control remote operation and data-logging of the component. Software within the component interprets valid message packets and returns appropriate replies. These replies appear in each specific On-Board *IS* System Component command code and response code table in Table 1-1 on page 1-8, and in the component chapters of this document.

The format of message packets is explained in the following sections of this chapter.

NOTE: *If a faulty message is received by the On-Board IS System Component (due to improper production in the host, or transmission media failure), the message is discarded and no response is sent back to the host as a reply.*

Message Packet Format

Each message packet consists of a series of ASCII characters transmitted through a standard asynchronous framing convention of:

- one (1) start bit
- seven (7) data bits
- a parity bit generated for even parity
- one (1) stop bit

A message packet for cryopumps and compressors is composed of:

- a starting flag character: **$** for ASCII (24 hex)
- message dependent Data Field Characters
- Message Checksum Character
- an ASCII carriage return (CR) code (0D hex)

The starting flag character serves the unique purpose of synchronizing the receiver to the transmitter, by signaling the start of the message packet. This **$** code is not contained in the set of characters used to construct the data field or the checksum character, and therefore establishes a fixed reference point to synchronize the data flow.

Whenever either receiver (host or slave) accepts a **$** character, all history and status of previous partial packet data (if any) is aborted and lost, and a new message packet-starts.

A message packet for the On-Board *IS* Controller is different from the other components. It is composed of:

- a starting flag character: **$** for ASCII (24 hex)
- **P** and a cryopump or compressor address, when communicating with a cryopump or compressor on the network

  (examples: **P00** = cryopump #0; **P20** = compressor #0)

  or **N** when communicating directly to the Controller
- message dependent Data Field Characters
- Message Checksum Character
- an ASCII carriage return (CR) code (**0D** hex)

See the appropriate *On-Board* IS *Component Installation* or *Operation* manual for setting component addresses.

Data Field Characters

The data field consists of one (1) to a maximum of fourteen (14) ASCII characters, the meaning of which is defined in the remaining section of this chapter and the command code and response code table (see About the Format of this Reference on page 1-2 for a list of where to find these codes). You can use all characters except **$** and 0D hex in the data field, if suitable.

Message Checksum Character

The message checksum character guards against garbled or incorrectly received messages that might cause undesirable or damaging results. Only messages conveyed accurately and intact from the host to slave (or vice-versa) are accepted and executed.

The checksum character that follows the data field for a message packet sent to a Cryopump or compressor is computed by a modified binary sum technique (see Checksum Algorithm Computation on page 1-6) over the characters in the data field.

When the checksum character is calculated for a message packet sent to the On-Board *I*S Controller, the 'N' character for a controller command, or the network address for a cryopump or compressor command, as well as the characters in the data field, are used to generate the proper checksum character.

The host or slave generates this sum based on the characters it used to produce the data field, and appends it after the field just prior to the CR code terminator. The receiving unit performs the same checksum algorithm on all characters which it receives between the **$** character and the character just prior to the CR terminator (non-inclusive).

If this sum matches the final character preceding the CR terminator, then the message is validated and processed by the receiver. If not, then the packet is discarded. No action or reply is given by the hardware. The checksum algorithm generates a character between ASCII **0** and **o** (30 hex to 6F hex), inclusive.

Checksum Algorithm Computation

The checksum algorithm shown in Figure 1-1 on page 1-7 is defined as follows:

Perform the 8 bit (modulo 256) sum of all the ASCII characters sent in the data field (with the most significant bit cleared to 0, ignore parity). This is performed for one (1) to fourteen (14) character code bytes. Fold the resulting eight bit sum into six bits by exclusive or-ing the two MSBs of the sum (**D7, D6**) with the two LSBs (**D1, D0**) of the sum, such that the new **D1** is the old **D1 XOR D7** and the new **D0** is the old **D0 XOR D6**. The resulting lower six bits (**D5..D0**) are then masked, producing a code range of 00 to 3F hex. This is then added to the ASCII code for **0** (30 hex), generating the final printable checksum character in the range of 30 hex to 6F hex (**0..o**).

```
////////////////////////////////////////////////////////////////
////////////////////////////////////////////
//FUNCTION:    CtiChkSum
//PURPOSE: Calculate the checksum for a given NULL terminated
//command
unsigned char CtiChkSum (unsigned char * pCommand)
{
     unsigned char checkSum = 0;
     unsigned char charNum = 0;
     do
     {
          checkSum = checkSum + (pCommand[charNum] & 0x7f) );
          charNum++;
     } while (pCommand[charNum] !=0 );
     checkSum = 0x3f & (checkSum ^ (checkSum >>6) );
     checkSum = (0x30 + checkSum) & 0x7f

     return checkSum;
}// End function CtiChkSum
```

Figure 1-1: C Language Checksum Calculation Example

Response Codes

The On-Board *IS* System Component returns response codes (see Table 1-1) whether it receives a valid message packet, or an incorrect and/or misinterpreted message packet (a software error, not a communications error). Invalid commands, improper parameter ranges, or requests to perform unexecutable operations all result in a response code reply.

Table 1-1: Response Codes

Code	Description
A	Command understood and resultant reply (if any) follows. No power failure or reset has occurred since last acknowledgement.
B	Command understood and resultant reply (if any) follows. A power failure or reset has occurred, which has not been acknowledged.
E	An invalid command was received. No power failure or reset has occurred since last acknowledgement.
F	An invalid command was received A power failure or reset has occurred, which has not been acknowledged.
G	A valid command was received, but cannot be acted upon. No power failure or reset has occurred since last acknowledgement.
H	A valid command was received, but cannot be acted upon. A power failure or reset has occurred, which has not been acknowledged.
I	*A valid command was received, but cannot be executed due to another serial port locking access to all other serial ports. No power failure or reset has occurred since last acknowledgement
J	*A valid command was received, but cannot be executed due to another serial port locking access to all other serial ports. A power failure or reset has occurred, which has not been acknowledged.
Z	*The On-Board *IS* Controller could not address a component on the network because of an incorrect address or powered down component.
*Indicates an On-Board *IS* Controller response code only. **NOTE:** *Z typically returns* `$ZBBCOMFAIL`.	

If a power loss response code is received, reset the response code to **`A, E,`** or **`G`** by using the **`S1`** command. Use the **`?`** command to reset response codes for the Controller.

Between the time that the host sends a message and the slave returns a response message, further characters sent from the host to the slave are ignored. The slave responds to any verified message packet it receives within one second. The host can use a time-out period in excess of this to detect a failure and the need to re-try. After a response to a message has been received by the host, it can send new messages immediately, faster than the one-second turn-around time-out. In this way the message traffic is self synchronizing.

Examples of Typical Communication Exchange

Example of Host - Cryopump/Compressor Communication

The following is an example of typical host - cryopump/compressor communication, where a host sends a query directly to a cryopump or compressor, requesting the software version:

	Flag	Data Field	Checksum	Terminator
ASCII	$	@	1	Cr
hex	24	40	31	0D

Slave sends reply of ASX01.00, meaning no error, cryopump version *SX01.00*.

	Flag	Data Field	Checksum	Terminator
ASCII	$	ASX01.00	C	Cr
hex	24	41 53 47 30 2E 36 30	43	0D

Example of Host - Controller Communication

The following is an example of typical host - Controller communication, where a host sends a query to the controller, requesting the software version from Pump 01 on the network:

	Flag	Pump Address	Data Field	Checksum	Terminator
ASCII	$	P01	@	b	Cr
hex	24	50 30 31	40	31	0D

Slave sends reply of ASX01.00, meaning no error, cryopump version *SX01.00*.

	Flag	Data Field	Checksum	Terminator
ASCII	$	ASX01.00	C	Cr
hex	24	41 53 47 30 2E 36 30	43	0D

This Page Intentionally Left Blank

2 On-Board *IS* Cryopump Commands

Overview

This section describes the commands and responses for communication with the On-Board *IS* Cryopump.

NOTE: *The On-Board IS System acts on received commands. Ensure that issued commands are not a detriment to the goals for the cryopumping system. Examples are commands that open the rough valve while the tool is processing, or turn off the cryopump (or compressor) while maintaining high vacuum.*

For communication syntax, see About Host and Component Communication on page 1-4 and Examples of Typical Communication Exchange on page 1-9.

Chapter Contents

Host to On-Board *IS* Cryopump Commands

The following table explains each valid command that you can issue from the host to the On-Board *IS* Cryopump.

Table 2-1: On-Board IS *Cryopump Commands*

Command	Name	Parameter	Description
`@`	Read Software Version	None	Returns an identifier string indicating module type and software revision level.
`A`	Read/Write Cryopump Motor State	`[0..1] | ?`	Turns the cryopump motor ON/OFF. The commands are: `A0` = Turns the cryopump motor OFF. `A1` = Turns the cryopump motor ON. `A?` = Reads back `0` or `1` as the current status.
`B`	Read Pressure Gauge State	`?`	Returns TC gauge status. Response is always `1` , to indicate TC gauge is on.
`D`	Read/Write Rough Valve State	`[0..1] | ?`	Turns roughing valve ON/OFF. The commands are: `D0` = Closes the valve. `D1` = Opens the valve. `D?` = Reads back `0` or `1` as the current status.
`E`	Read Purge Valve State	`?`	Returns purge valve status. The responses are: `0` = closed `1` = open
`H`	Read/Write First Stage Temperature Control Setpoint	`[0, 65..120, 320] | ?`	Sets temperature control of first stage. If a `0` is specified, temperature control is turned off. A `?` reads back the current status of `0, 65..120, 320`.
`J`	Read First Stage Temperature	None	Returns the first stage temperature in Kelvin (K). Readings of 400K for 3 consecutive seconds indicates a Diode Short Fault. Readings of 0K for 3 consecutive seconds indicates a Diode Open Fault.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
K	Read Second Stage Temperature	None	Returns the second stage temperature in Kelvin (K). Readings of 400K for 3 consecutive seconds indicates a Diode Short Fault. Readings of 0K for 3 consecutive seconds indicates a Diode Open Fault.
L	Read Cryopump Pressure	None	Sends back the cryopump gauge pressure.
N	Abort/Start Regeneration Process	**[0..4]**	Starts or aborts a regeneration cycle. The commands are: **N0** = Aborts a currently running regeneration cycle. **N1** = Starts a Sublime Regeneration. **N2** = N/A. **N3** = Starts a Sublime Shutdown. **N4** = Starts a Startup Regeneration.
O	Read Regeneration State	None**|?**	Send back the current regeneration step. See tables in Regeneration Response (O Command) Description on page 2-13 for responses. If the **?** parameter is used, the command returns a numeric code rather than the letter code. The numeric code is equal to the letter code (ASCII numeric value), minus **0x41**. This numeric value is then converted to the ASCII printable representation of the number. As an example, if the **O** command returns the letter code **B**, the **O?** command would return the string **1**. The **1** string is derived from (**B – 0x41 = 1**); the binary **1** is then converted to the ASCII value for **1** and returned in the response string.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
P	Read/Write Regeneration Parameters	**[6, A, e] [3 digit number] | ?**	Reads/Sets a regeneration parameter. **P6** = Power Recovery temperature (after power failure), range = 110 - 260 **PA** = Clear/set rough valve interlock, range = 0 - 1 **Pe** = Sublimation temperature, range = 110 - 250
S1	Read System Status	None	Send back the various status indicators.The response is offset by **@** (decimal *64*) for ASCII printable results. The bit weights are: **01H**: **0** = cryopump motor off, **1** = cryopump motor on **02H**: **0** = rough valve closed, **1** = rough valve open **04H**: always **1** (purge valve open) **08H**: always **1** (TC gauge on) **10H**: always **0** **20H**: Power Reset, **0** after reset, **1** after this query **40H**: always **1** (character bias bit) **80H**: always **0** (character bias bit) Example: ASCII response to **S1** command is **I** Response is offset by **64** In decimal form: 73 - 64 = 9 Convert decimal value to binary: 001001 Bits **01H** and **08H** are set, indicating that the cryopump and the TC gauge are on. Bit **20H** is not set, indicating that there has been a power reset.
VA	Read Cryopump Serial Number	**?**	Returns the first 8 characters of the cryopump serial number
VQ	Read Cryopump Serial Number	**?**	Returns the last 8 characters of the cryopump serial number

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
Y	Read Elapsed Time	**?**	Read elapsed time. The units are in hours and expressed as an integer.
Z	Read Number of Completed Regeneration Cycles	**?**	Read number of regeneration cycles that have been completed.
a	Read Time Since Last Regeneration	None	Read how long it has been since the last Regeneration. A completed Startup, Sublime, or Shutdown cycle resets time to zero. The units are in hours.
e	Read Regeneration Abort Error Code	None	Read error code result of regeneration cycle operation. Returns one of the following responses: **@** = No error. **F** = User performed a manual abort to stop regeneration. **K** = Running a Sublime Regeneration before a power failure, but too warm to continue Sublime Regeneration after the power failure. **L** = Initial cooldown test failed due to pressure. **M** = Initial cooldown test failed due to temperature. **N** = Cooldown timeout error. **O** = Warmup timeout error. **W** = Loss of network communication during coordinated regeneration. **[** = Running a regeneration before power failure. **** = Internal communication error.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
g	TC Gauge Calibration	None	Initate calibration of TC gauge. Will return **G** or **H** error code if conditions are not valid to perform a calibration, such as the TC gauge is not within some minimum range of the 0 operating point. (Maximum pressure for performing calibration is 30 microns.)
i	Read/Write Power Fail Recovery Mode	**[0..2] | ?**	Set/Clear/Query the power fail recovery mode. The commands are: **i0** = Power fail recovery mode OFF. **i1** = Power fail recovery mode ON. Following a power cycle, cryopump turns on if motor was on, or initates a regeneration if a regeneration was in progress. **i2** = Power fail recovery mode COOL. Following a power cycle, cryopump turns on if motor was on, and temperature is below the power fail recovery setpoint. System initates a regeneration if a regeneration was in progress. **i?** = Query power fail recovery mode.
j	Read/Write Regeneration Start Delay	**[0..5994] | ?**	Program/Query the delay to start of regeneration parameter. The value is in minutes. A **?** returns the value, set in minutes.
k	Read Regeneration Count Down Timer	None	This value represents the time remaining in a delay start within a regeneration cycle. For times from 1 to 60 seconds, the response will be 1 minute. For times from 61 seconds to 120 seconds the response will be 2 minutes. When it reaches 0, the response will be 0 minutes.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
t	Read/ Acknowledge Power Fail Recovery Flag	**? | =**	Command to either poll **t?** or acknowledge and clear **t=** the power fail recovery flag of the cryopump. The responses are defined as: **0** = Power fail recovery ackowledgement received. **1** = Cryopump performing Regeneration cooldown before power failure; continue performing cooldown. **2** = Cryopump performing power fail recovery regeneration. **3** = Power fail recovery is in process. Motor is on and cryopump is recovering. System attemps cooldown to <18K. **4** = Power fail recovery is successful and cryopump is cold. **5** = Motor is on, but cryopump did not cool down to 18K within 20 minutes. **6** = Cryopump was off before power failure, no recovery necessary. **7** = Power fail recovery received, but cryopump is too warm to start Regeneration. **8** = Power fail recovery is in process.
u	Read Motor Over Temperature Flag	None	Indicates whether the cryopump motor was shut down due to an over temperature condition. **0** = motor has not exceeded setpoint temperature. **'** = motor is off due to over temperature condition.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
v	Read Regeneration Condition Flag	None	Polls cryopump for several flag conditions combined as bits in one response. The response is offset by **@** (decimal *64*) for ASCII printable results. The bit weights are: **01H** = True when cryopump is waiting for accress to the rough manifold. **02H** = always 0 **04H** = True when a heater failure is detected during a regeneration cycle. **08H** = always 0 **16H** = always 0 **32H** = always 0
z	Read/Write Lockout On-Board *IS* Remote	**[0..1] | ?**	Set/Clear/Query the user access lock state. When true, user can not access any On-Board *IS* Remote displayed menu areas, other than monitor and cryopump information. **z0** = Lockout is turned OFF. **z1** = Lockout is turned ON. **z?** = Query lockout status. Reads back **0** or **1** for status.

Table 2-1: On-Board IS *Cryopump Commands (Continued)*

Command	Name	Parameter	Description
#l (Lower-case "L")	Read/Write Serial Port Baud Rate	**[0..1]** **[?|0..3]**	Read/Set the baud rate of one of the serial ports. First parameter selects the port and the second parameter selects the baud rate. First Parameter: **0** = Host Port **1** = Service Port Second parameter: **0** = 2400 (baud rate) **1** = 9600 **2** = 19200 **3** = 38400 **?** = Query baud rate Example 1: Send **#101**. Host port is now set to a baud rate of 9600. Example 2: Send **#10?**. Query for baud rate of host port.
]C	Read Cryopump Pump Status Buffer	**?**	Command to query the compacted status of the cryopump. This command requests multiple pieces of information and status about the cryopump, available with separate queries, but with one command. See]C Command Description on page 2-10 for more information. The **j** command (see j Command Controller Codes and Definitions on page 3-15) for the On-Board *IS* Controller is the more commonly used command for multiple cryopump systems.

]C Command Description

The **]C** command queries for the buffer status of the cryopump (see]C on page 2-9). For examples of how to decode the responses from sending the **]C** command, see]C Command Example 1 on this page and]C Command Example 2 on page 2-12.

For command queries from the Controller, see j Command Controller Codes and Definitions on page 3-15.

]C Command Example 1

Send query: **]C?**
(Asks for the buffer status of cryopump.)

Receive the ASCII response: **Ai@@ExA`A**

To decode the response, use this Cryopump Code string (where **z** is the **A** or **B** response code):

```
z 01ABCDEF 01KLMNOP 01YZGHIJ 01STUVWX 01ghijQR 01abcdef 01pqrstu 010klmno
```

Define the ASCII response (**Ai@@ExA`A**) with the Hex and Binary equivalents:

Response Character	**A**	**i**	**@**
Hex	41	69	40
Binary	N/A	**01101001**	**01000000**
Cryopump Code	N/A	**01ABCDEF**	**01KLMNOP**

Response Character	**@**	**E**	**x**
Hex	40	45	78
Binary	**01000000**	**01011001**	**01111000**
Cryopump Code	**01YZGHIJ**	**01STUVWX**	**01ghijQR**

Response Character	**A**	**`**	**A**
Hex	41	60	41
Binary	**01000000**	**01100000**	**01000001**
Cryopump Code	**01abcdef**	**01pqrstu**	**010klmno**

Use the definitions of the Individual and Group Item Codes in the Cryopump Code to determine the status of the cryopump:

Table 2-2: Individual Item Codes

Item Code	0	1	Description
A	Off	On	Regeneration status
B	Off	N/A	(Not used)
C	Off	On	TC gauge status
D	Closed	Open	Purge valve status
E	Closed	Open	Rough valve status
F	Off	On	Cryopump motor status

Table 2-3: Group Item Codes

Item Code	Decimal Numeric Range	Description
GHIJKLMNOP	0 - 1023	Regeneration state
QRSTUVWXYZ	0 - 999K	First Stage Temperature, in Kelvin
abcdefghij	0 - 999K	Second Stage Temperature, in Kelvin
klmnopqrstu	0 - 999 µ	TC Gauge Pressure, in microns

To de-code the Individual Item Codes (from Table 2-2), match the binary bit with the appropriate letter code:

Response Character	**i**
Hex	69
Binary	**01101001**
Controller Code	**01ABCDEF**

bit **A** = **1**: regeneration on

bit **B** = **0**: N/A (not used)

bit **C** = **1**: TC gauge on

bit **D** = **0**: purge valve closed

bit **E** = **0**: rough valve closed

bit **F** = **1**: cryopump motor on

To de-code the Group Item Codes (from Table 2-3 on page 2-11), match the binary bit with the appropriate letter code for temperature and pressure information:

Regeneration State	**G**	**H**		**I**	**J**	**K**	**L**		**M**	**N**	**O**	**P**				
Binary	**0**	**0**		**0**	**0**	**0**	**0**		**0**	**0**	**0**	**0**	= **0** x **0** = **0** decimal			

Regeneration state = **A** = Warm up
(See Table 2-4 on page 2-13 for all **O** command state codes.)

First Stage Temperature	**Q**	**R**		**S**	**T**	**U**	**V**		**W**	**X**	**Y**	**Z**				
Binary	**0**	**0**		**0**	**1**	**1**	**0**		**0**	**1**	**0**	**0**	= **0** x **64** = **100** decimal			

First stage temperature = 100K

Second Stage Temperature	**a**	**b**		**c**	**d**	**e**	**f**		**g**	**h**	**i**	**j**				
Binary	**0**	**0**		**0**	**0**	**0**	**0**		**1**	**1**	**1**	**0**	= **0** x **E** = **14** decimal			

Second stage temperature = 14K

TC Gauge Pressure	**k**	**l**	**m**	**n**	**o**	**p**	**q**		**r**	**s**	**t**	**u**				
Binary	**0**	**0**	**0**	**0**	**1**	**1**	**0**		**0**	**0**	**0**	**0**	= **0** x **60** = **96** decimal			

Pressure = 96 µ

]C Command Example 2

You want to know if the rough valve is open or closed for the cryopump.

Send query: **]C?**

Receive the ASCII response: **Ai@@ExA'A**

To decode the response, use this Controller Code string (where **z** is the **A** or **B** response code):

```
z 01ABCDEF 01KLMNOP 01YZGHIJ 01STUVWX 01ghijQR 01abcdef 01pqrstu 010klmno
```

From Table 2-2 on page 2-11, the Item Code for the rough valve status is **E**, with 0 = closed and 1 = open.

If:

Response Character	**A**	**i**	**@**
Hex	41	69	40
Binary	N/A	**0 1 1 0 1 0 0 1**	**01000000**
Controller Code	N/A	**0 1 A B C D E F**	**01KLMNOP**

Then the rough valve is closed.

Regeneration Response (O Command) Description

The tables in this section describe the On-Board *IS* Cryopump response codes to the **O** command. Each response indicates the cryopump actions during the current regeneration phase.

For more information about Sublime Regeneration, see the *On-Board* IS *Two Stage Cryopump with Sublime Regeneration Installation and Operation Instructions*, part number 8040737.

Responses to the **O** command are **A..z** (equal to numeric codes **0..57** and **58..N**). The numeric return information is preceded by the plus sign (**+**). Refer to page 2-3 for a description of how to convert from an alpha character response to a numeric character response.

NOTE: *Responses may appear to overlap within a regeneration cycle.*

Table 2-4: All O Responses, in Alpha-Numeric Order

Response Code	Regeneration Type: Regeneration Phase Action or Description
****	Startup: Start warming.
A, E	Startup: Continue warming.
H	Startup: Perform extended purge.
P	Sublime and Startup: Regeneration is finished and history is stored.
V	Sublime and Startup: Regeneration error or manual abort.
X, Y	Startup: Start as part of power fail recovery.
63..67	Startup: Perform cooldown.
68	Sublime: Start warming.
69..70	Sublime: Wait for pressure to recover.
71..72	Sublime: Start as part of power fail recovery.
73..76	Sublime: Cool down after frost sublimates.
78	Sublime: Wait for time delay to finish, and then start warming.
79	Sublime: Cooldown after frost sublimates and requests rough pump.
80	Sublime: Continue warming and requests rough pump.
81	Sublime: Wait for ice to sublimate and requests rough pump.

Table 2-4: All O Responses, in Alpha-Numeric Order

Response Code	Regeneration Type: Regeneration Phase Action or Description
`84..85`	Sublime: Continue warming.
`86..90`	Sublime: Wait for frost to sublimate.
`92`	Startup: Continue warming.
`119`	Shutdown: Continue warming.
`120`	Shutdown: Warm second stage as part of Shutdown Regeneration
`[`	Cooldown is complete and the cryopump is cold. Calibrate the TC gauge for 0 µ.

3 On-Board *IS* Controller Commands

Overview

This section describes the commands and responses for communication with the On-Board *IS* Controller.

NOTE: *The On-Board IS system acts on received commands. Ensure that issued commands are not a detriment to the goals for the cryopumping system. Examples are commands that open either rough or purge valves while the tool is processing, or turn off the cryopump (or compressor) while trying to achieve high vacuum.*

For communication syntax, see About Host and Component Communication on page 1-4 and Examples of Typical Communication Exchange on page 1-9.

Chapter Contents

Host to On-Board *IS* Controller Commands

The following table explains each valid command that you can issue from the host to the On-Board *IS* Controller.

NOTE: *When you send a Controller command, precede it with an* **N**.

For more information about command syntax, see Message Packet Format on page 1-4.

Table 3-1: On-Board IS *Controller Commands*

Command	Name	Parameter	Description
?	Power Reset Acknowledge	None	Acknowledges that host computer is aware of a power reset. Sending this command results in normal response code changing from **B** to **A**.
@	Software Revision	None	Returns an identifier string indicating Controller type and software revision level.
A?	Read Controller Serial Number	**?**	Returns the eleven character serial number for the On-Board *IS* Controller. Any blank positions are padded at the end with spaces.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
B	Scan Network	None	Returns a numeric code that identifies which modules are currently active on the network. The cryopumps active on the network are then configured according to the rough valve cooperation groupings already defined in the On-Board *IS* Controller. The On-Board *IS* Controller password (if set) is sent, and interlock turned on for all current cryopumps. The returned code is a number from **0** to **1073741823** decimal (0x0 - 0x3fffffff), comprised of the sum of binary bit weights for each of the possible cryopumps and compressors. The number of characters returned will be constant, regardless of the value returned. Empty digits are represented by a 'space' character. See Appendix B: Controller Command Bit Weights on page 5-4. A response of **0** indicates that no cryopumps or compressors are responding to the network scan. Example with a two cryopump and one compressor system: Pump #1 address 02 Pump #2 address 03 Comp #1 address 20 Return **A 1048588** **1048588** (decimal) **10000C** (hexadecimal) **10000000000000001100** (binary)

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
C	Get Rough Map	**[1..5]**	Returns a numeric code that identifies which cryopump modules are grouped in the selected rough valve map. The parameter 1 through 5 selects the map to be returned. (These are also Maps A through E on the Remote.) The response is the sum of the bit weights for each cryopump in the rough map. See command B on page 3-3 and Table 5-2 on page 5-4. Example to get cryopumps found for rough valve map A: Send: **NC1** Response: **A O** This response indicates that there are no cryopumps in the map.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
D	Set Rough Map	**[1..5] [0.. 1048575]** (StatusBit-Field)	Accepts a numeric code that identifies which cryopump modules are grouped in the selected rough valve map. The parameter 1 through 5 selects the map. (These are also Maps A through E on the Remote.) The bit weights for the set notation are the same as for the **B** command. In this way the rough valve map can be remotely programmed. A logical limitation is imposed such that a rough map set must have at least two members, and the members so defined must not be present in any of the other four (non-selected) rough maps. To move a cryopump from one map to another it must first be removed from the old grouping before being set into a new grouping. It must also reject the command if the map number is currently in a regeneration cycle. Example: **ND10** - Remove all cryopumps from roughing map A. **ND1129** - Set cryopumps 0 and 7 to roughing map A. (0x81)(0000 1000 0001) **ND2258** - Set cryopumps 1 and 8 to roughing map B (0x102)(0001 0000 0010).
E	Get All Rough Maps	None	Returns a numeric code that identifies which cryopump modules are assigned to one of the five roughing maps. This membership is logically based on the programming of the five maps, and does not rely on whether the cryopumps are currently communicating or not. Cryopump modules not in this set are not required to cooperate when accessing a rough valve during regeneration cycles. Example: After entering the **D** example commands, get the following for **E**: Send: **NE** Response: **A 387**

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
`F`	Get Exclusive Grant	None	Returns a numeric code that identifies which cryopump modules have been granted the exclusive use of the rough valve for their set. Membership in this set indicates an active regeneration is in process for the corresponding member cryopump. Other cryopumps which belong to the same rough map as those currently granted rough usage will be held off from regeneration usage of the rough valve until the rough valve is relinquished.
`G`	Read/Write On-Board *IS* Controller Password	`[0..32767] | ?`	Command to set or query the On-Board *IS* Controller Network password value. If a `?` is the argument, then the current password value is returned. This value is used to open up menu access to the remote cryopumps using one common password. If a number `0..32767` is the argument, then this value becomes the new password value, with a value of `0` being no password.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
L	Query Locked Out Rough Map Set	None	Query which of rough map sets **A..E** are locked out by external intelligence. The default at power up is none. The **M** command may acquire or grab sets which are not already in cooperative use, or latch a seizure desire to take over when maps become free to over-ride. The **N** command is used to free or release remote sets, or to remove seizure desire for rough map sets not yet locked out. They are also automatically released if On-Board *IS* Controller parser is not given **L** query at least each 5 seconds when supervision is active. **O** command sets/clears/queries supervisor mode. Returned value is sum of the following bit weights: Rough Map Weight A 1 B 2 C 4 D 8 E 16
M	Acquire Remote Rough Map Sets	**[0..31]**	Command to acquire rough map sets, which are not already in cooperative use. Zero or more sets can be identified to be disabled from the rough grant cycle in the On-Board *IS* Controller. Sets are identified by using the bit weights shown in the **L** command.
N	Free Locked Out Rough Maps	**[0..31]**	Command to release rough maps. Maps, which are desired to be released, are entered as elements of a set code. They are then released. No error is caused by releasing sets not currently held. Use bit weights shown in **L** command. Rough maps are also automatically released if On-Board *IS* Controller is not given **L** query at least each 5 seconds when supervision active.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
O	Read/Write Supervisor Mode	**? | =[0..1]**	Command to enable/disable/query the supervisor mode. **NO = 1**: Enables supervisor mode A periodic **L** command to poll the state of the remote lockout rough sets is required to maintain these sets locked out. If 5 seconds elapses with no such poll then the sets are released to On-Board *IS* Controller operation. **NO = 0**: Disables supervisor mode **NO?**: Queries status of supervisor mode
P	Get Group 1 Regeneration Cryopumps	None	Returns a numeric code that identifies which cryopump modules are assigned to regeneration group 1 for multi-pump regeneration start/stops. The response is the sum of the bit weights for each cryopump in the regeneration group. See command B on page 3-3 and Table 5-2 on page 5-4. **NOTE:** *This command is provided for backward compatibility with On-Board IS Controller versions that do not support 5 separate regeneration groups. It is essentially the same as the command* **X1**.
Q	Set Group 1 Regeneration Cryopumps	**[0..1048575]**	Accepts a numeric code that identifies which cryopump modules are to be grouped together for common regeneration start/stops. The entered code has the same meaning as in the **B** command (see 3-3). **NOTE:** *This command is provided for backward compatibility with On-Board IS Controller versions that do not support 5 separate regeneration groups. It is essentially the same as the command* **W1n**.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
W	Set Regeneration Group	**[1..5] [0..1048575]**	Accepts a numeric code that identifies which cryopump modules belong in the selected regeneration group. The parameter 1 through 5 selects the group number and the value represents which cryopumps. The bit weights for the set notation are the same as for the **B** command (see 3-3). In this way the group start grouping can be remotely programmed. Cryopumps in any such group will be commanded to start either a Full Regeneration or a Fast Regeneration (if Fast Regeneration cryopumps available) by a manual regeneration start or remote computer interface command to start. See command Y on page 3-10.
X	Get Regeneration Group	**[1..5]**	Returns a numeric code that identifies which cryopump modules are in the selected regeneration group. The parameter 1 through 5 selects the group's number. The response is the sum of the bit weights for each cryopump in the regeneration group. See command B on page 3-3 and Table 5-2 on page 5-4. Any cryopump can be present in any one of the groups 1-5.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
Y	Start/Abort/ Query Regeneration Group	**[1..5]** *(group number = n, in this description)* **[0..4]**	**NYn0**: Abort a currently running Group Sublime Regeneration in group n. This command results in aborting the regeneration of every cryopump in the group. **NYn1**: Clear the error condition in group n. *If an error condition exists, it must be cleared prior to initiating another group regeneration.* **NYn2**: Initiate a Group Sublime Regeneration in group n. This command results in initiating a Sublime Regeneration to every cryopump in the group. **NYn4**: Request information indicating whether the group regeneration was started successfully. Responses to the **Yn4** command are as follows: **0** – new status unavailable, command pending **1** – starting Group Regeneration abort **2** – starting Group Sublime Reneration **3** – Not applicable **4** – Group Regeneration inhibited. Problem with group. Clear this response prior to initiating another Group Regeneration. **5** – successful Group Regeneration has been initiated
Y (Continued)	Start/Abort/ Query Regeneration Group	**[1..5]** *(group number = n, in this description)* **[0..4]**	The following scenarios inhibit the start of a Group Regeneration: - One or more of the cryopumps in the group is already running a regeneration. - The group is set up to include a cryopump that is not on the network.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
f	Read/Write Serial Port Baud Rate	**[0..2]** **[? | 0..3]**	Read/Set the baud rate of one of the three serial ports. First parameter selects the port and the second parameter selects the baud rate. First Parameter: **0** = Host Port **1** = Service Port **2** = Rear Aux Second parameter: **0** = 2400 (baud rate) **1** = 9600 **2** = 19200 **3** = 38400 (available for "**0** = Host Port" only) **?** = Query baud rate Example 1: Send **N#101**. Host port is now set to a baud rate of 9600. Example 2: Send **N#11?**. Query for baud rate of service port.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
h	Get Buffered Cryopump List	None	Command to poll for a list (as in command B on page 3-3) of cryopumps, which are registered to be buffered for quick access to cryopump state. This means one or more cryopumps are periodically polled by the On-Board *IS* Controller for status, with this status being kept in updated form in the On-Board *IS* Controller, available for immediate reply to a host query without needing to wait for a cryopump to respond. A return of **0** indicates that no cryopumps are being buffered, a return of **1** indicates cryopump **0** alone is registered for the buffering, etc. For maximum performance, do not attempt to poll cryopumps that are not available to the network.
i	Set Buffered Cryopump List	Status Bit Field **[0.. 1048575]**	Command to define which of the 20 cryopumps available to an On-Board *IS* Controller are registered for data/status buffering. Registration of cryopumps not attached to the network will degrade operation, as the On-Board *IS* Controller will poll for data on those stations and time out on the communication attempts. Status of stations not present are signaled with a status bit in the buffer area. Use of this command enables/selects/disables the buffering option on a cryopump by cryopump basis.
j	Get Buffered Cryopump Data	**[0..19]**	Command to query for the buffer status of a specific cryopump. See j Command Controller Codes and Definitions on page 3-15 in this section.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
n	Read/Write Full Rough Coordination	**? | =[0..1]**	**Nn?**: query full rough coordination status. **Nn=0**: turn off full rough coordination. **Nn=1**: turn on full rough coordination. The full rough coordination feature allows all the On-Board *IS* Cryopumps in a common rough map to rough simultaneously during the initial roughing phase of a full regeneration. Subsequent requests for the rough manifold will be granted on an individual basis. The result is shorter Full Regeneration times. To use this feature, full rough coordination must be on, and the cryopumps must be started from within a regeneration group.
o	Read/Write Power Fail Coordination	**? | =[0..1]**	**No?**: query power fail coordination status. **No=0**: turn off power fail coordination. **No=1**: turn on power fail coordination. Command to query, turn ON or turn OFF the Power Fail Coordination Feature. Upon recovery from a power failure, all cryopumps sharing a common manifold will rough out to base pressure at the same time, assuming that both Full Rough Coordination and Power Fail Coordination were turned ON.

Table 3-1: On-Board IS *Controller Commands (Continued)*

Command	Name	Parameter	Description
`[A1J`, where **`1`** is the helium map in this case, but could be **`[1..5]`**	Get/Set Helium Map	**`?|[0..1073741823]`**, where the bit weights are the crompressors and cryopumps assigned to the helium map	Assigns the cryopumps and compressors attached to each helium manifold. (This is similar to the configuring of the rough map, except it is a helium map.) See Table 5-2 on page 5-4. For example: **`$N[A2J1048591`** assigns a compressor (Channel C, Address 0) and 4 cryopumps (Channel A, Address 0..3) to helium map 2. If only one helium map is used, then you can use the **`$NB`** command to scan the network, and the StatusBitField response to set the helium map. To disable this feature, set the map to **`0`**.

j Command Controller Codes and Definitions

The **j** command queries for the buffer status of a specific cryopump. For query syntax, see the Controller Command for j on page 3-12. For examples of how to decode the responses from sending the j command, see j Command Example 1 on this page and j Command Example 2 on page 3-18.

j Command Example 1

Send query: **Nj2**
(Asks for the buffer status of cryopump 2.)

Receive the ASCII response: **AiKdV'A@A**

To decode the response, use this Controller Code string (where **z** is the **A** or **B** response code):

```
z 01HGFCDE 0100ABIJ 01QRSTUV 01efghij 01opqrst 0100MNOP 0100abcd 0100klmn
```

Define the ASCII response (**AiKTV'A@A**) with the Hex and Binary equivalents:

Response Character	**A**	**i**	**K**
Hex	41	69	4B
Binary	N/A	**01101001**	**01001011**
Controller Code	N/A	**01HGFCDE**	**0100ABIJ**

Response Character	**d**	**V**	**'**
Hex	64	56	60
Binary	**01100100**	**01010110**	**01100000**
Controller Code	**01QRSTUV**	**01efghij**	**01opqrst**

Response Character	**A**	**@**	**A**
Hex	41	40	41
Binary	**01000001**	**01000000**	**01000001**
Controller Code	**0100MNOP**	**0100abcd**	**0100klmn**

Use the definitions of the Individual and Group Item Codes in the Controller Code to determine the status of the cryopump:

Table 3-2: Individual Item Codes

Item Code	0	1	Description
A	Old data	New data	Data was refreshed since last
B	Off	On	Regeneration status
C	Closed	Open	Purge valve status
D	Closed	Open	Rough valve status
E	Off	On	Cryopump motor status
F	Off	On	TC gauge status
G	Off	N/A	(Not used)
H	Reset	Acknowledged	Power reset status
I	No	Yes	*Registered for buffering
J	No	Yes	Currently on network
*Reads Yes when registered and responding to bitbus traffic. Reads No if Not responding, or if not registered, or both.			

Table 3-3: Group Item Codes

Item Code	Decimal Numeric Range	Description
MNOPQRSTUV	0 - 999K	First Stage Temperature, in Kelvin
abcdefghij	0 - 999K	Second Stage Temperature, in Kelvin
klmnopqrst	0 - 999 u	TC Gauge Pressure, in microns

To de-code the individual item codes (from Table 3-2), match the binary bit with the appropriate letter code:

Response Character	**i**
Hex	69
Binary	**01101001**
Controller Code	**01HGFCDE**

bit **H** = **1**: power reset acknowledged

bit **G** = **0**: N/A (not used)

bit **F** = **1**: TC gauge on

bit **C** = **0**: purge valve closed

bit **D** = **0**: rough valve closed

bit **E** = **1**: cryopump motor on

Response Character	**K**
Hex	4B
Binary	**01001011**
Controller Code	**0100ABIJ**

bit **A** = **1**: new data

bit **B** = **0**: regeneration off

bit **I** = **1**: registered for buffering

bit **J** = **1**: currently on the network

To de-code the group item codes (from Table 3-3), match the binary bit with the appropriate letter code for temperature and pressure information:

First Stage Temperature	**M N**	**O P Q R S T U V**	
Binary	**0 0**	**0 1 1 0 0 1 0 0**	= **0** x **64** = **100** decimal

First stage temperature = 100K

Second Stage Temperature	**a b**	**c d e f g h i j**	
Binary	**0 0**	**0 0 0 1 0 1 1 0**	= **0** x **16** = **22** decimal

Second stage temperature = 22K

TC Gauge Pressure	**k l**	**m n o p q r s t**	
Binary	**0 0**	**0 1 1 0 0 0 0 0**	= **0** x **60** = **96** decimal

Pressure = 96 microns

j Command Example 2

You want to know if the motor for cryopump 5 is on or off.

Send query: **Nj5**
Receive the ASCII response: **AiKTV'A@A**

To decode the response, use this Controller Code string (where **z** is the **A** or **B** response code):

```
z 01HGFCDE 0100ABIJ 01QRSTUV 01efghij 01opqrst 0100MNOP 0100abcd 0100klmn
```

From Table 3-2 on page 3-16, the Individual Item Code for the cryopump motor status is **E**, with 0 = off and 1 = on.

If:

Response Character	**A**	**i**	**K**
Hex	41	69	4B
Binary	N/A	**01101001**	**01001011**
Controller Code	N/A	**01HGFCD**	**0100ABIJ**

Then the motor for cryopump 5 is on.

4 On-Board *IS* Compressor Commands

Overview

This section describes the commands and responses for communication with the On-Board *IS* Compressor.

NOTE: *The On-Board IS system acts on received commands. Ensure that issued commands are not a detriment to the goals for the cryopumping system. Examples are commands that open either rough or purge valves while the tool is processing, or turn off the cryopump (or compressor) while trying to achieve high vacuum.*

For communication syntax, see About Host and Component Communication on page 1-4 and Examples of Typical Communication Exchange on page 1-9.

Chapter Contents

Compressor Commands

The On-Board *IS* 1000 Compressor commands follow the cryopump command structure and are also available through the On-Board *IS* Controller.

NOTE: *When you communicate with a cryopump or compressor on the network, use* **P** *and a cryopump or compressor address:*
P00 = *cryopump #0*
P20 = *compressor #0*

The following table lists the On-Board IS 1000 Compressor commands.

Table 4-1: On-Board IS 1000 Compressor Commands

Command	Name	Parameter	Description
@	Read Software Version	None	Returns an identification string indicating module type and software version.
A	Read/Write Compressor Motor State	**[0..1] | ?**	Turns the compressor motor ON/OFF. The commands are: **A0** = Turns the motor OFF. **A1** = Turns the motor ON. **A?** = Reads back **0** or **1** as the current status.
F	Get Water Flow Rate	**?**	Returns the calculated flow rate of cooling water through the compressor in gallons/min.
O	Get Supply Pressure	**?**	Returns the measured helium supply pressure in psig. This command will return an **H** response code during the first 15 seconds after starting the cryopump motor to allow pressures to stabilize. Values will be returned as a integer number ranging from 0 to 500.
P	Get Return Pressure	**?**	Returns the measured return helium pressure in psig. This command will return an **H** response code during the first 15 seconds after starting the cryopump motor to allow pressures to stabilize. Values will be returned as a integer number ranging from 0 to 500
Y	Read Elapsed Time	**?**	Reads the elapsed time. The units are in hours and expressed as an integer.

5 Appendices

Overview

The following appendices are included to provide the user with a single location for specific information related to the Brooks Automation Product.

Contents

Appendix A: Customer Brooks Automation Technical Support Information

When contacting Brooks Automation for Technical Support, please have the following information available.

1. Record the part number and serial number from the equipment.
2. Provide the installed location of the equipment.
3. Provide name, e-mail address, and telephone number of the person to contact.
4. List any error codes received during the failure.
5. Prepare a detailed description of the events relating to the error.
   - Time that the equipment has been in operation
   - Work that was done on the equipment prior to the error
   - Functions that the equipment was performing when the error occurred
   - Actions taken after the error and the results of those actions
   - Other information that may assist the Specialist
6. Contact Brooks Automation Technical Support at these numbers:

Brooks Location	GUTS® Contact Number
North America	1-800-FOR-GUTS (1-800-367-4887) US/Canada +1-978-262-2900
Europe	+49 1804 CALL GUTS (+49 1804 2255 4887)
Japan	+81-45-477-5980
China	+86-21-5131-7066
Taiwan	+886-3-552-5225
Korea	+82-31-288-2500
Singapore	+65-6464-1481

For additional contact information, please go to the Brooks Automation web site at www.brooks.com or send an E-mail to techsupport@brooks.com.

Add additional contacts here:

Table 5-1: Additional Contacts

NAME	PHONE	E-MAIL

Appendix B: Controller Command Bit Weights

The table in this section shows the bit weights used by several On-Board *IS* Controller commands.

See the command B on page 3-3 for bit weight use.

Table 5-2: Bit Weights

ID#	Decimal	Hex	ID#	Decimal	Hex
0	1	0x00001	15	32768	0x08000
1	2	0x00002	16	65536	0x10000
2	4	0x00004	17	131072	0x20000
3	8	0x00008	18	262144	0x40000
4	16	0x00010	19	524288	0x80000
5	32	0x00020	20	1048576	0x100000
6	64	0x00040	21	2097152	0x200000
7	128	0x00080	22	4194304	0x400000
8	256	0x00100	23	8388608	0x800000
9	512	0x00200	24	16777216	0x1000000
10	1024	0x00400	25	33554432	0x2000000
11	2048	0x00800	26	67108864	0x4000000
12	4096	0x01000	27	134217728	0x8000000
13	8192	0x02000	28	268435456	0x10000000
14	16384	0x04000	29	536870912	0x20000000

Index

M

O

P

Q

R

S

T

Y