Gemini Command Reference

Contents

Introduction

The Gemini controller hub is based upon the FocusLynx dual stepper control circuit featuring multiple communication connections and methods. Gemini can control two separate stepper motors for focuser and rotator movements through a single circuit board that allows serial RS-232, wired Ethernet, or wireless 802.11 Wi-Fi Ethernet connection for communication with a client PC. The command syntax is universal regardless of which connection and communication method is being used to send the command and receive responses. TCP/IP protocols are used for both wired Ethernet (port 9760) and the optional WiFi board (port 9761). A basic text string command structure is described herein.

Optec offer a freely available serial terminal application that can be used to connect with and control the Gemini controller hub using the command syntax described below. The command processor included with the Gemini controller is designed to use a Start-Command character and an End-Command character. This allows for commands of varying lengths and the ability to start a new command or terminate a command at any time. The Start-Command and End-Command characters are '<', and '>' respectively.

Following the Start-Command character the next character designates whether the command is intended for focuser 'F', rotator 'R', or the hub 'H' (controller) followed by a Device ID placeholder of '1'. The next two characters define a Transaction ID used as a response check for multi-threading processes. The following six characters define the Command ID as defined within the pages below. Any remaining characters before the end character '>' will provide Boolean or Payload values of varying length.

Syntax Conventions

The individual commands are listed below and should follow the format below:

    <DdiiCCCCCCyxxx…>   where,

    '**D**' is the **Target Device** – valid values are "F" for focuser, "R" for rotator, or "H" for hub,
    '**d**' is the **Device ID** – always "1" for the Gemini focusing rotator *(see note below)*,
    '**ii**' is the **Transaction ID** – a 2-digit numeric character string from 00 to 99,
    '**CCCCCC**' is the **Command ID**, always 6-characters, as detailed below,
    '**y**' is an optional **Boolean** character used in certain commands as a Boolean operator,
    '**zzz…**' is the **Payload**, a variable length character string used in certain commands requiring additional parameters. The length and contents of this Payload string are dependent upon the preceding Command ID. Additional details are given below.

In the following syntax reference, all upper-case letters are constant for the given command; all lower-case letters are variable parameters for the command. For example, valid **Target Device** identifiers 'D' are exclusively "F", "R", or "H".

The 'd' parameter is the **Device ID** placeholder and will always have a value of "1" for the Gemini Focusing Rotator. *(Note: The Device ID is residual from the FocusLynx Command Reference where multiple devices of the same type (i.e. focusers) could be controlled from the same hub.)*

The 'ii' parameter is the **Transaction ID** and can be any numeric text value from 00 to 99. The Transaction ID created and defined by the client PC and is always echoed by the Gemini controller preceded by the bang '!' character. Using a Transaction ID allows multi-threaded client level applications to maintain proper registration of commands and responses.

The 'CCCCCC' **Command ID**'s are individually details in this document. Command ID's fall into three classes, GET commands return configuration and device details. SET commands push configuration details to the device. DO action commands perform various device movements or other actions.

The 'y' parameter is optionally included as a **Boolean** operator where the Command ID requires details about a feature that may be enabled or disabled. Valid values for 'y' are "0" and "1", with "1" generally meaning Enabled, True, or Yes. Likewise, "0" represents Disabled, False, or No.

The 'zzz…' **Payload** parameter is a variable length string and will follow certain Command ID's requiring additional details. For example, the Payload parameter may represents a target position in steps, or a network IP address, or a Device NickName.

Gemini Responses

The Gemini controller will immediately respond to every command with a bang '!' character, followed by the Transaction ID 'ii', and a newline (ASCII 0x10) character to indicate that a command was successfully received. Often a multiline response will finish with an "END" string followed by the newline character. If a syntax error was detected or an unrecognized command was received, the controller will respond with a string of text indicating an error code and error message in place of the expected response.

See Appendix E for additional Error Code details.

Gemini Device Properties

There are several device properties for the Focuser, Rotator, and Gemini Hub reported by the various commands described below. Each property is reported using an 8-character description of the property. A brief explanation of each property is provided in this section. See Appendix C for default values.

Focuser Properties

Nickname: A 16-character ASCII text string describing the device name. This variable can be set by the client software and end user. Valid string lengths for "Nickname" are 1 to 16 ASCII characters.

MaxSteps: The maximum step position for the focuser. See Appendix C for default value.

Dev Type: Device Type code, always type "A" for Gemini Focuser. See Appendix A.

TComp On: Temperature Compensation On flag. Boolean value, 1 means temperature compensation is active, 0 indicates temperature compensation is off or inactive.

TCMode A: Current value for Temperature Compensation Mode A in steps/degree C.

TCMode B: Current value for Temperature Compensation Mode B in steps/degree C.

TCMode C: Current value for Temperature Compensation Mode C in steps/degree C.

TCMode D: Current value for Temperature Compensation Mode D in steps/degree C.

TCMode E: Current value for Temperature Compensation Mode E in steps/degree C.

CurrentTC: Indicates the currently selected TC Mode for active temperature compensation. Valid values are A, B, C, D, and E.

BLCompOn: BackLash Compensation flag, Boolean value of 1 means backlash compensation is enabled, 0 indicates backlash compensation is disabled.

BLCSteps: BackLash Compensation Steps indicates the number of steps the focuser will move beyond a target position, and the return to the target. Backlash Compensation is designed to take up any gear backlash in the motor gearhead or mechanical interface that may be inherent in the device. Valid values are 00 to 99 steps.   BLCSteps can be set by client software and user. See Appendix C for default value.

HOnStart: Home On Start, Boolean flag to indicate whether a homing action should be initiated upon power up. Value of 1 indicates Home On Start, 0 indicates do not home but pull the current position from EEPROM.

CurrStep: The current position of the Gemini focuser. Valid values are 0 to MaxSteps.

CurrTemp: The current temperature in degrees Celsius, measured from the active temperature probe attached to the focuser.

TargStep: The current target position the focuser is moving toward. Target P can be set be several commands. If Target P and Curr Pos are not equal, IsMoving will be true (1).

IsMoving: Boolean Moving flag, true (1) if the focuser is currently moving, false (0) if stationary.

IsHoming: Boolean Homing flag, true (1) if the focuser is currently executing a Home operation.

Is Homed: Current status, true (1) if the focuser has been homed, false (0) if not yet homed or last Home operation was unsuccessful. The IsHomed property will be set to false if the focuser is Halted.

TempProb: Boolean flag indicating presence of a temperature probe. True (1) if a temperature probe is connected, false (0) otherwise.

Rotator Properties

Nickname: A 16-character ASCII text string describing the device name. This variable can be set by the client software and end user. Valid string lengths for "NickName" are 1 to 16 ASCII characters.

MaxSteps: The maximum step position for the rotator. Normally, all positions and moves are defined as instrumental Position Angles only.   See Appendix C for the default MaxSteps value for the Gemini rotator.

Dev Type: Device Type code, the Gemini Rotator is defined as Dev Type "B". See Appendix A.

BLCompOn: BackLash Compensation flag, Boolean value of 1 means backlash compensation is enabled, 0 indicates backlash compensation is disabled.

BLCSteps: BackLash Compensation Steps indicates the number of steps the focuser will move beyond a target position, and the return to the target. Backlash Compensation is designed to take up any gear backlash in the motor gearhead or mechanical interface that may be inherent in the device. Valid values are 00 to 99 steps.   BLCSteps can be set by client software and user. See Appendix C for default value.

HOnStart: Home On Start, Boolean flag to indicate whether a homing action should be initiated upon power up. Value of 1 indicates Home On Start, 0 indicates do not home but pull the current position from EEPROM.

iReverse: The Reverse Boolean indicates whether the reverse flag is set to true (value = 1) or false (value = 0).   The default reverse property is false and represents rotator movements in the clockwise (CW) direction.   The default direction for the Gemini rotator is CW.

MaxSpeed: MaxSpeed is the maximum speed in steps per second for the rotator motor. *Note: This command is reserved for future implementation.*

CurrStep: The current step position for the rotator. This value is reported for debugging purposes only and cannot be altered directly by the client software. To avoid cable-wrap, CurrStep value can only be between 0 and MaxStep.

TargStep: The current target rotator step position the device is moving toward.

CurentPA: The current instrumental position angle of the Gemini rotator in 1/1000th of a degree. Valid position angle values are 000000 to 359999. Note that instrumental PA=0 is equivalent to PA=360.

TargetPA: The current target instrumental position angle the rotator is moving toward. TargetPA can be set be several commands. If TargetPA and CurrntPA are not equal, IsMoving will be true (1).

IsMoving: Boolean Moving flag, true (1) if the focuser is currently moving, false (0) if stationary.

IsHoming: Boolean Homing flag, true (1) if the focuser is currently executing a Home operation.

IsHomed: Current status, true (1) if the focuser has been homed, false (0) if not yet homed or last Home operation was unsuccessful. The IsHomed property will be set to false is the focuser is Halted.

PAOffset: The Position Angle Offset is a numeric string value that indicates the number of steps required to adjust the instrumental Position Angle (PA) of PA=0 to match Sky PA=0. This is a residual setting from early Pyxis rotator implementations. *IMPORTANT NOTE:* Current Best Practices should always set this instrumental position angle offset to 0, and leave the algorithms for matching instrument

or Rotator PA to the camera Sky PA for the higher level client software.

Gemini Hub Properties

Firmware: The firmware version currently running on the Gemini controller.

LEDBrite: Current setting for the LED Brightness, numeric text value of 00 to 99.

HandCtrl: Boolean flag indicates the status of an attached hand controller.   A value of 1 indicates a hand controller is attached, 0 means no hand controller is detected.

Wired IP: The Wired IP returned value is the string representation for the current IP address of the wired Ethernet connection. The IP address for IPV4 consists of four octets separated by periods "."   If the Gemini hub is connected to a network via wired Ethernet, the IP address assigned by the local network DHCP server will be reported.

WiFi Mod: Indicates whether the controller has detected an attached Wi-Fi module or not. A value of 1 indicates a WiFi module is present, 0 means no WiFi module was detected.

WiFiConn: Boolean value indicates where successful communication has taken place between the Wi-Fi module and the main Gemini controller board. Value of 1 indicates successful communication, 0 indicates a communication error.

WiFiFVOK: WiFi Firmware O.K. indicates that the firmware version installed in the Wi-Fi module is compatible with the main Gemini firmware. 1 means they are compatible, 0 means incompatible. If WiFiFVOK is false, no other WiFi commands will be functional.

WiFiFVer: Reports the firmware version of the Wi-Fi module.

WiFiSSID: Indicates the SSID (wireless network name) that the Wi-Fi module is currently configured to connect to. Valid string lengths up to 32 characters.

WiFiAddr: Reports the current IP address for the Wi-Fi module. The IP address for IPV4 consists of four octets separated by periods "."   If the Gemini WiFi module is connected to a wireless network the value of the IP address assigned by the DHCP server will be reported. If the WiFi module is configured for Ad-Hoc mode, the WiFi IP address will be self-assigned. The default Ad-Hoc WiFi IP address will be 169.254.1.1.   The IP address for IPV4 consists of four octets separated by periods "."   If the Gemini WiFi module is connected to a wireless network the value of the IP address assigned by the DHCP server will be reported. If the WiFi module is configured for Ad-Hoc mode, the WiFi IP address will be self-assigned. The default Ad-Hoc WiFi IP address will be 169.254.1.1.

WiFiSecM: The WiFi Security Mode will be reported.   Indicates the security mode that the Wi-Fi module is configured to use when connecting to a wireless network. See Appendix B – Wi-Fi Security Modes for available Security Modes and additional information.

WiFiSecK: The WiFi Security Key will be reported. Depending upon the security mode selected, a passphrase or security key will be required for wireless communications with the WiFi router.

GET Configuration Commands

Command: Focuser - GET Device NickName

Syntax: `<FdiiGETDNN>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the focuser identification string or "Nickname". Valid string lengths for "Nickname" are 1 to 16 ASCII characters. The controller will respond with the focuser's user-defined nickname.

Note: GETDNN is generally used to ping the device to check communication status.

Example Command: `<F101GETDNN>`

Example Response:

```
!01
Nickname = Castor
END
```

Command: Rotator - GET Device NickName

Syntax: `<RdiiGETDNN>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the rotator identification string or "Nickname". Valid string lengths for "Nickname" are 1 to 16 ASCII characters. The controller will respond with the rotator's user-defined nickname.

Note: GETDNN is generally used to ping the device to check communication status.

Example Command: `<R102GETDNN>`

Example Response:

```
!02
Nickname = Pollux
END
```

Command: Focuser - GET STAtus

Syntax: `<FdiiGETSTA>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command will request the controller to report the status of the specified focuser. The first line of the status report is "!ii" and the last line is "END" to indicate that the output is complete. The status output contains the following fields:

CurrTemp: The current temperature in degrees Celsius.

CurrStep: The current step position of the specified focuser.

TargStep: The target step position the focuser is currently moving toward.

IsMoving: Boolean flag is set to '1' if the device is moving, '0' if the device is stationary.

IsHoming: Boolean flag is set to '1' while the device is homing and '0' otherwise.

Is Homed: Boolean flag is set to '1' if the focuser has been homed, '0' if not yet homed.

TempProb: Boolean flag indicates the status of an attached temperature probe. A value of '1' means a probe is attached, '0' means no probe is detected.

Note: *CurrTemp*, *CurrStep*, and *TargStep* are each reported as numerical values with no additional padding.   Additionally, the *CurrTemp* reponse includes a sign and decimal point in the output ascii string.

Example Command: `<F103GETSTA>`

Example Response:
```
!03
CurrTemp = +24.2
CurrStep = 57600
TargStep = 57600
IsMoving = 0
IsHoming = 0
Is Homed = 1
TempProb = 1
END
```

Command: Rotator - GET STAtus

Syntax: `<RdiiGETSTA>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command will request the controller to report the status of the specified rotator. The first line of the status report is "!ii" and the last line is "END" to indicate that the output is complete. The status output contains the following fields:

CurrStep: The current step position of the specified rotator.

TargStep: The target step position that the rotator is currently moving toward.

CurentPA: The current position angle (x1000) for the specified rotator.

TargetPA: The target position angle (x1000) the rotator is moving toward.

IsMoving: Boolean flag is set to '1' if the device is moving and '0' if the device is stationary.

IsHoming: Boolean flag is set '1' while the device is homing and '0' otherwise.

Is Homed: Boolean flag will be set to '1' after the focuser is home, '0' otherwise.

Note: *CurrStep*, *TargStep*, *CurentPA*, and *TargetPA* are each reported as ascii strings representing numerical values with no additional padding.

Example Command: `<R104GETSTA>`

Example Response:
```
!04
CurrStep = 45000
TargStep = 45000
CurentPA = 359999
TargetPA = 359999
IsMoving = 0
```

```
IsHoming = 0
Is Homed = 1
END
```

Command: Focuser - GET ConFiGuration settings

Syntax: `<FdiiGETCFG>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command will request the controller to report the configuration settings of the specified focuser. The first line of the configuration report is "!ii" and the last line is "END" to indicate that the output is complete. The configuration output contains the following fields:

Nickname:	The user-defined nickname of the specified focuser.
MaxSteps:	The maximum absolute position, MAXPOS, for the selected focuser.
Dev Type:	A one character designator for the currently set device type, see Appendix A.
TComp On:	Boolean value, '1' = true (temp comp on) and '0' = false (temp comp off).
TCMode A:	Temperature Coefficient (TC) value for each of five available settings, A to E.
(through *TCMode E*)	*Note*: All TC value units are negative steps per degree C.
CurrentTC:	Indicates the currently active temperature compensation mode, 'A' through 'E'.
BLCompOn:	BackLash Compensation On, '1' when true, '0' when compensation is disabled.
BLCSteps:	Number of steps set in EEPROM for backlash compensation.
	Note: the focuser will travel past the target position before returning to the target position in order to compensate for mechanical backlash in the focusing device. Backlash compensation applies to IN direction moves only.
TC Start:	Indicates Temperature Compensation is active, '1' means on, '0' means off.
HOnStart:	Home on Start- Boolean value, '1' is true, '0' means do not home on start.

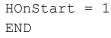**Example Command:** `<F105GETCFG>`

Example Response:
```
!05
Nickname = Castor
MaxSteps = 115200
Dev Type = A
TComp On = 0
TCMode A = 86
TCMode B = 86
TCMode C = 86
TCMode D = 86
TCMode E = 86
CurrenTC = A
BLCompOn = 0
BLCSteps = 40
TC Start = 0
```

```
HOnStart = 1
END
```

Command: Rotator - GET ConFiGuration settings

Syntax: `<RxiiGETCFG>`

Parameters: x =Device ID [1]; ii = Transaction ID.

Description: This command will request the controller to report the configuration settings of the specified rotator. The first line of the configuration report is "!ii" and the last line is "END" to indicate that the output is complete. The configuration output contains the following fields:

Nickname:	The user-defined nickname of the specified rotator.
MaxSteps:	The maximum step position the selected rotator is capable of moving to.
Dev Type:	A one character designator for the currently set device type, see [Appendix A](#).
BLCompOn:	Boolean flag indicates whether backlash compensation is active. A value of '1' indicates rotator will perform backlash compensation, '0' indicates the feature is turned off.
BLCSteps:	This numeric string value indicates the number of steps that the rotator will travel past the target position before returning to the target position in order to compensate for mechanical backlash in the rotating device.
PAOffset:	The Position Angle Offset is a numeric string value that indicates the number of steps required to adjust the instrumental Position Angle (PA) of PA=0 to match Sky PA=0. This is a residual setting from early Pyxis rotator implementations.
HonStart:	The Home on Start Boolean indicates if the rotator will home upon initial power-up or start. A value of '1' indicates the rotator will perform a home at start, '0' indicates the focuser will not home at start.
iReverse:	Reverse Boolean indicates whether the reverse flag is set to true (value = '1') or false (value = '0').
MaxSpeed:	MaxSpeed is the maximum speed in steps per second for the rotator motor. *Note*: This command is reserved for possible future implementation allowing client software setting of the rotator speed.

Example Command: `<R106GETCFG>`

Example Response:

```
!06
Nickname = Pollux
MaxSteps = 215999
Dev Type = B
BLCompOn = 0
BLCSteps = 40
PAOffset = 0
```

```
HonStart = 1
iReverse = 0
MaxSpeed = 800
END
```

Command: Gemini Hub- GET ConFiGuration settings

Syntax: `<HxiiGETCFG>`

Parameters: x =Device ID [1]; ii = Transaction ID.

Description: The Hub Get Config command requests the hub to report the current Gemini Hub configuration information. The first line of the configuration report is "!ii" and the last line is "END" to indicate that the output is complete. The Hub Configuration report contains the following fields. See Gemini Hub Properties for full descriptions.

Firmware: The firmware version currently running on the Gemini controller.

LEDBrite: Current setting for the LED Brightness, numeric text value of 00 to 99.

HandCtrl: Boolean flag indicates the status of an attached hand controller.
              A value of '1' indicates a hand controller is attached, '0' means not detected.

Wired IP: The Wired IP address of the Gemini Hub is reported.

WiFi Mod: Indicates the presence of the optional Wi-Fi module A value of '1' indicates a WiFi module is present, '0' means no WiFi module was detected.

WiFiConn: Boolean value indicates communication status of the Wi-Fi module. Value of '1' indicates successful communication, '0' indicates a communication error.

WiFiFVOk: WiFi Firmware check indicates compatibility between hub and Wi-Fi module. '1' means WiFi firmware version is compatible with the Gemini firmware, '0' incompatible.

WiFiFVer: Reports the firmware version of the Wi-Fi module.

WiFiSSID: Indicates the SSID (wireless network name) that the Wi-Fi module is currently configured to connect to. Valid string lengths up to 32 characters.

WiFiAddr: Reports the current IP address for the Wi-Fi module.

WiFiSecM: WiFi Security Mode is indicated.   See Appendix D – WiFi Security Modes.

WiFiSecK: The WiFi Security Key will be reported. See Appendix D – WiFi Security Modes.

Example Command: `<H107GETCFG>`

Example Response:

```
!07
Firmware = 1.0.0
LEDBrite = 75
HandCtrl = 0
Wired IP = 169.254.1.1
WiFi Mod = 0
WiFiConn = 0
```

```
WiFiFVOK = 0
WiFiFirm = 0.0.0
WiFiSSID =
WiFiAddr = 0.0.0.0
WiFiSecM = A
WiFiSecK =
END
```

DO Action Commands

Command: Focuser- HALT

Syntax: `<FdiiDOHALT>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command immediately halts the current focuser movement. The controller will respond immediately with the "!ii" and "END" strings to acknowledge the command was received.

Note: If Temperature Compensation was active at the time the Halt command was received, temperature compensation will be automatically disabled.

Example Command: `<F110DOHALT>`

Example Response:

```
!10
END
```

Command: Rotator- HALT

Syntax: `<RdiiDOHALT>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command immediately halts the current rotator movement. The controller will respond with the "!ii" and "END" strings to acknowledge the command was received.

Note: If the rotator HALT is performed while the magnet is directly over the hall sensor, the IsHomed property will be set to false and a HOME procedure will be required.

Example Command: `<R111DOHALT>`

Example Response:

```
!11
END
```

Command: Focuser- HOME

Syntax: `<FdiiDOHOME>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the focuser to begin its homing routine. The controller will respond with "!ii" and "END" to indicate the focuser has started homing.

Note: The focuser IsMoving property will be set to true (1) and the IsHomed property will be set to false (0) until completion of the Home operation. FxiiDOHALT will immediate stop the homing operation.

Example Command: `<F112DOHOME>`

Example Response:

`!12`

`END`

Command: Rotator- HOME

Syntax: `<RdiiDOHOME>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the specified rotator to begin its homing routine. The controller will respond with "!ii" followed by "END" to indicate that is has started homing.

Note: The rotator IsMoving property will be set to True (1) and the IsHomed property will be set to False (0) until completion of the Home operation. RxiiDOHALT will immediate stop the homing operation.

Example Command: `<R113DOHOME>`

Example Response:

`!13`

`END`

Command: Focuser- CENTER

Syntax: `<FdiiCENTER>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the specified focuser to move to the center of its travel. The target position will be set to half of the focuser's maximum position value (Max Posi+1)/2. The IsMoving property for the focuser will be set to true until the focuser reaches the center position. The Gemini controller will respond with "!ii" and "END" to indicate that is has begun moving to center.

Note: The IsMoving flag will be set to True (1) until the completion of the action.

Example Command: `<F114CENTER>`

Example Response:

`!14`

`END`

Command: Focuser- Begin Relative Move

Syntax: `<FdiiDOMOVEy>`

Parameters: d =Device ID [1]; ii = Transaction ID, y = direction of movement.

Description: This command requests the specified focuser to begin moving in the y direction, y='0' moves IN, y='1' moves OUT. The move will initially travel at low speed, then shift into high speed after a few seconds.

Note: This command is designed for internal use with the momentary push buttons of the hand control. The target position is set to the extents of travel (0 for y='0', and Max Posi for y='1') and IsMoving is set to true ('1'). Issue the DOSTOP command to stop the current movement. The device will automatically stop at either extent of travel.

Gemini Command Reference, Revision 2.2 – January 2019   (eff. Firmware v1.2.0)

Page 15 of 34

Example Command: `<F115DOMOVE0>`

Example Response:

`!15`
`END`

Command: Rotator- Begin Relative Move

Syntax: `<RdiiDOMOVEy>`

Parameters: d =Device ID [1]; ii = Transaction ID, y = direction of movement.

Description: This command requests the specified rotator to begin moving in the y direction, y='0' moves CCW, y='1' moves CW direction. The move will initially travel at low speed, then shift into high speed after a few seconds.

Note: This command is designed for internal use with the momentary push buttons of the hand control. The target position is set to the extents of travel (0 for y='0', and Max Posi for y='1') and IsMoving is set to true ('1'). Issue the DOSTOP command to stop the current movement. The device will automatically stop at either extent of travel.

Example Command: `<R116DOMOVE0>`

Example Response:

`!16`
`END`

Command: Focuser- Stop Relative Move

Syntax: `<FdiiDOSTOP>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the specified focuser to stop the current movement. IsMoving will be set to False ('1') and target position will be set to the current position.

Note: This command is designed for internal use with a release of the momentary push buttons in the hand control.

Example Command: `<F117DOSTOP>`

Example Response:

`!17`
`END`

Command: Rotator- Stop Relative Move

Syntax: `<RdiiDOSTOP>`

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command requests the specified rotator to stop the current movement. IsMoving will be set to False ('1') and target position will be set to the current position.

Note: This command is designed for internal use with a release of the momentary push buttons in the hand control.

Example Command: `<R118DOSTOP>`

Example Response:

```
!18
END
```

Command: Focuser- Move Absolute Position

Syntax: `<FdiiMOVABSz>`

Parameters: d =Device ID [1]; ii = Transaction ID, z = target position.

Description: This command requests the specified focuser to begin moving to the absolute position specified. The target position must be between 0 and the focuser's maximum position. The Target Position will be set to z and the IsMoving flag will be set to True ('1') until the completion of the action.

Note: The 'z' payload parameter can be padded to six characters if preferred, but padding is not required. (i.e. 'z' = 100 is equivalent to 'z' = 000100.)

Example Command: `<F119MOVABS55000>`   *(Request move to absolute position= 55000)*

Example Response:

```
!19
END
```

Command: Rotator- Move Absolute Position Angle

Syntax: `<RdiiMOVEPAzzzzzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z =target position angle (x1000).

Description: This command requests the specified rotator to begin moving to the absolute position angle specified. The target position must be between 0 and the 359999.   The TargetPA will be set to 'z' and IsMoving will be set to true ('1').

Note: All position angles are instrumental position angles and multiplied by 1000 with no decimal point. (i.e. instrumental PA=15 is represented by 15000).

Example Command: `<R120MOVEPA90000>`   *(Request move to position angle 90000 or PA=90°)*

Example Response:

```
!20
END
```

Command *(Reserved)*: Rotator- Move Absolute Step Position

Syntax: `<RdiiMOVABSzzzzzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z =target step position.

Description: This command requests the specified rotator to begin moving to the absolute step position specified. The target position must be between 0 and MaxStep.   The Target Position will be set to 'z' and IsMoving will be set to true ('1').

Note: This command should not be used at the client level. All client level requests should use the MOVEPA command above for instrumental Position Angles rather than request individual step movements. This command is reserved for de-rotation applications only.

Example Command: `<R121MOVABS90000>`   *(Request move to step position 90000)*

Example Response:

```
!21
END
```

SET Configuration Commands

The following commands are used to set device configuration settings. Unlike the Get commands which tend to be all-inclusive in the properties they report, the Set commands are separated to apply to only a single device property. In the syntax definitions lower-case letters still indicate a parameter. See the Parameters section under each command for more information.

Command: Focuser- Set Device Nickname

Syntax: `<FdiiSETDNNzzz…>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = Device NickName (max of 16 characters).

Description: This command is used to set a custom nickname for the specified focuser. The nickname is the string that is returned by the controller when the GETDNN command is executed and is also reported in the GETCFG command response. The Gemini controller will respond with "END" to indicate that the setting was successfully updated.

The following example is used to set the focuser nickname to "Castor".

Example Command: `<F130SETDNNCastor>`

Example Response:

```
!30
END
```

Command: Rotator- Set Device Nickname

Syntax : `<RdiiSETDNNzzz…>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = Device NickName (max of 16 characters).

Description: This command is used to set a custom nickname for the specified rotator. The nickname is the string that is returned by the controller when the GETDNN command is executed and is also reported in the GETCFG response. The Gemini controller will respond with "END" to indicate that the setting was successfully updated.

The following example is used to set the rotator nickname to "Pollux"

Example Command: `<R131SETDNNPollux>`

Example Response:

```
!31
END
```

Command: Focuser- Set Device Type

Syntax: `<FdiiSETDEVz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = Device Type identifier.

Description: The Set Device Type command declares the type identification for the specified device. For the Gemini Focuser the Device Type will always be "A". See Appendix A - Device Types for additional information.

Example Command: `<F132SETDEVA>`

Example Response:

```
!32
```

```
END
```

Command: Rotator- Set Device Type

Syntax: `<FdiiSETDEVz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = Device Type identifier.

Description: The Set Device Type command declares the type identification for the specified device. For the Gemini Rotator the Device Type will always be "B". See [Appendix A - Device Types](#) for additional information.

Example Command: `<R132SETDEVB>`

Example Response:
```
!32
END
```

Command: Focuser- Set Home on Start

Syntax: `<FdiiSETHOSy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: The Set Home on Start command sets whether or not the device will home upon initial start-up or power on. A payload parameter of y = 0 will turn off the home on start feature; y='1' sets the device to home on start.

Example Command: `<F133SETHOS0>`   (sets Home on Start to false)

Example Response:
```
!33
END
```

Command: Rotator- Set Home on Start

Syntax: `<RdiiSETHOSy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: The Set Home on Start command sets whether or not the device will home upon initial start-up or power on. A payload parameter of y = '0' will turn off the home on start feature; y='1' sets the device to home on start.

Example Command: `<R134SETHOS0>`   (sets Home on Start to false)

Example Response:
```
!34
END
```

Command: Focuser- Set Temperature Compensation Enabled

Syntax: `<FdiiSETTCEy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: The Set Temperature Compensation Enabled command allows the client to enable or disable the temperature compensation feature built into the Gemini focuser. A parameter of y = '0' will turn off the Temperature Compensation feature; y='1' will turn on the TC feature.

Example Command: `<F135SETTCE1>`   (set TC Enabled to true)

Example Response:
```
!35
END
```

Command: Focuser- Set active TC Mode

Syntax: `<FdiiSETTCMm>`

Parameters: d =Device ID [1]; ii = Transaction ID; m = TC mode to set active (A through E).

Description: The Set Temperature Compensation Mode command sets the active TC mode to be used when temperature compensation is enabled. Each of the five available TC Modes may have unique temperature coefficients (TC) for use with various optical configurations. The payload parameter 'z' can be any of the five available modes represented by 'A' through 'E'. The example below sets TC mode 'B' as the active temperature compensation mode.

Example Command: `<F136SETTCMB>`

Example Response:
```
!36
END
```

Command: Focuser- Set Temperature Compensation Coefficient

Syntax: `<FdiiSETTCCmszzzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; m= Mode to which the coefficient applies; s = sign (+/−) of the coefficient; zzzz= Temperature Coefficient (in steps per degree) expressed as a four digit number left-padded with zeros.

Description: This command allows the user to set the value of a specific temperature coefficient for a specific focuser. The value must include a sign (+ or -) and four numeric digits. The following example sets the temperature coefficient for mode 'D' to positive 192 steps per degree C.

Example Command: `<F137SETTCCD+0192>`

Example Response:
```
!37
END
```

Command: Focuser- Set Temperature at Start Enabled

Syntax: `<FdiiSETTCSy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: This command allows the user to enable (y='1') or disable (y='0') the Temperature Compensation at Start feature built into the Gemini controller. When the feature is enabled the Gemini controller will automatically perform a temperature compensation move when the device is powered on. The device will perform this move based on the recorded temperature and position of the last time temperature compensation was enabled.

Example Command: `<F138SETTCS1>`   (turns on TempComp at Start)

Example Response:
```
!38
SET
```

Command: Focuser- Set Backlash Compensation Enabled

Syntax: `<FdiiSETBCEy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: This command allows the client to enable or disable the backlash compensation feature built into the Gemini focuser. The payload parameter of y='1' turns on this feature; y='0' turns backlash compensation off.

Example Command: `<F139SETBCE1>`   (turns on Backlash Compensation)

Example Response:

```
!39
SET
```

Command: Focuser- Set Backlash Compensation Steps

Syntax: `<FdiiSETBCSzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = number of steps (0 to 99).

Description: This command allows the user to specify the number of steps that the focuser will move beyond the target position before returning to the target position. The following example sets the number of backlash compensation steps for the Gemini focuser to 45 steps.

Example Command: `<F140SETBCS45>`

Example Response:

```
!40
SET
```

Command: Rotator- Set Backlash Compensation Enabled

Syntax: `<FdiiSETBCEy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: This command allows the client to enable or disable the backlash compensation feature built into the Gemini rotator. The payload parameter of y='1' turns on this feature; y='0' turns backlash compensation off.

Example Command: `<R141SETBCE0>`   (turns off Backlash Compensation)

Example Response:

```
!41
SET
```

Command: Rotator- Set Backlash Compensation Steps

Syntax: `<RdiiSETBCSzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = number of steps (0 to 99).

Description: This command allows the user to specify the number of steps that the rotator will move beyond the target position before returning to the target position. The following example sets the number of backlash compensation steps for the Gemini rotator to 99 steps.

Example Command: `<R142SETBCS99>`

Example Response:
```
!42
SET
```

Command: Rotator- Set Reverse Property

Syntax: `<RdiiSETREVy>`

Parameters: d =Device ID [1]; ii = Transaction ID; y = Boolean value.

Description: The Set Reverse Property command allows the client to set the rotator reverse property to true or false.   A payload parameter y='1' sets Reverse to true; y='0' sets Reverse to false meaning the direction of movement will be in the default direction. The default direction for increasing Position Angle values is always clockwise. Setting the Reverse property to true will not affect the underlying step position of the rotator. However, the Gemini hub responses for position angles will be reversed when this property is true. Instrumental PA=0 and PA=180 will be unaffected.

Note: The Gemini rotator will always home in the anti-clockwise direction until it locates the home magnet, then move in a clockwise direction to instrumental PA = 0. From that position

Example Command: `<R143SETREV1>`   (set Reverse property to True)

Example Response:
```
!43
SET
```

Command: Gemini Hub- Set LED Brightness

Syntax: `<HdiiSETLEDzz>`

Parameters: d =Device ID [1]; ii = Transaction ID; z = brightness value (0 to 99).

Description: This command can be used to set the brightness of the power-indication LED on the Gemini controller enclosure. Changing the values adjusts the total current through the LED thus increasing or decreasing its brightness. A value of '0' will turn the LED off.

Example Command: `<H197SETLED75>`   (set LED brightness to 75%)

Example Response:
```
!97
SET
```

Command: Gemini Hub- Reset To Factory Defaults

Syntax: <HdiiRESETH>,

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command will reset all configuration and status information for the focuser to factory defaults.

See Appendix C for Factory Default settings.

Example Command: <H198RESETH>

Example Response:

!98

SET

Command: Gemini Hub- Reboot

Syntax: <HdiiREBOOT>

Parameters: d =Device ID [1]; ii = Transaction ID.

Description: This command will perform a soft reboot of the Gemini hub for remote operations.

Example Command: <H199REBOOT>

Example Response:

!99

SET

Wi-Fi Configuration Commands

The Gemini controller has an optional Wi-Fi plug-in module that enables PC's, tablets, mobile devices, or any device that supports Wi-Fi connections to communicate with it. The following commands apply to only those devices which have the Wi-Fi module installed.

In the previous section all of the commands began with the characters "SC" which stands for *Set Configuration.* The Wi-Fi commands all begin with the characters "SW" which stands for *Set Wi-Fi* configuration.

<u>IMPORTANT</u>: In order to commit changes to the Wi-Fi module you must issue the *Push Wi-Fi Settings* command which instructs the Gemini controller to save the new settings that you have entered. Before you can call the Push Wi-Fi Settings command you MUST first issue the commands to set the SSID, security mode, security-key (if applicable) and security-key-index (if applicable).

It is advised that you do not attempt to modify Wi-Fi settings while connected to the controller via the Wi-Fi.

Command: Set Wi-Fi SSID

Syntax: <FHSWSSnzzz…>

Parameters: n – SSID name section index, zzz… – SSID name

Description: This command is used to set a new SSID (wireless network name) which will be used by the Wi-Fi module when connecting. The command processor on the Gemini Controller can only handle parameters which are 16 characters long or less. According to the Wi-Fi protocol an SSID can be up to 32 characters long. Thus multiple commands must be issued in order to set the SSID to a value greater than 16 characters long. The n parameter of this command tells the controller which section of the name you are setting according to the following:

> *n=0: characters 1 through 16*
>
> *n=1: characters 17 through 32*

The following example sets the wireless SSID to "My Private Network"

Example Command:

<FHSWSS0My Private Netwo>

```
<FHSWSS1rk>
```
Example Response:
```
!
SET
!
SET
```

Command: Set Wi-Fi Security Mode

Syntax: `<FHSWSMz>`

Parameters: z – Security mode code (See Appendix B – Wi-Fi Security Modes for more information.)

Description: This command is used to set the security mode that the Wi-Fi module will use when connection to a wireless network. Appendix B – Wi-Fi Security Modes contains a table of all of the available security modes and their respective codes.

The following example sets the Wi-Fi security mode to WEP-40 (mode D).

Example Command: `<FHSWSMD>`

Example Response:
```
!
SET
```

Command: Set Wi-Fi Security Key

Syntax: `<FHSWSKnzzz…>`

Parameters: n – Key section index, zzz… - New security key

Description: This command is used to set the Wi-Fi security key and functions the same way as the *Set Wi-Fi SSID* command. The security key can be up to 64 characters in length thus four sections of 16 characters are needed. The n parameter indicates to the section of the key being set according to the following:

> *n=0: characters 1 through 16*
>
> *n=1: characters 17 through 32*
>
> *n=1: characters 33 through 48*
>
> *n=1: characters 49 through 64*

The number of characters required for the security key depends on the selected security mode and the network settings of the network you are trying to connect to. See Appendix B – Wi-Fi Security Modes for more information.

The following example sets the security key to "My Private Security Key 123456 7654321"

Example Command:
```
<FHSWSK0My Private Secur>
<FHSWSK1ity Key 123456 7>
<FHSWSK2654321>
```
Example Response:
```
!
SET
```

```
!
SET
!
SET
```

Command: Set Wi-Fi Security WEP Key Index

Syntax: `<FHSWWIz>`

Parameters: *z* – WEP key index (1 through 4);

Description: This command is used to set the WEP key index which is used by the Wi-Fi module when connecting to a wireless network which uses WEP authentication. The key index can be any value 1 through 4. The key index is setup and determined by the network administrator. The following example sets the WEP security key index to 2.

Example Command: `<FHSWWI2>`

Example Response:

```
!
SET
```

Command: Push Wi-Fi Settings

Syntax: `<FHSWPS>`

Parameters: None

Description: All of the Wi-Fi settings described above do not take effect until this command is executed. It is important that you set all other necessary Wi-Fi settings before executing this command. Once this command is executed the Wi-Fi module will attempt to reconnect to a wireless network using the received parameters.

Example Command: `<FHSWWI2>`

Example Response:

```
!
SET
```

Command: Wi-Fi Module Reset

Syntax: `<FHWIFIRESET>`

Parameters: None

Description: This command forces the Wi-Fi module to perform a software reset.

Example Command: `<FHWIFIRESET>`

Example Response:

```
!
SET
```

Command: Restore Wi-Fi Defaults

Syntax: `<FHWIFIDEFAULTS>`

Parameters: None

Description: This command will force the Wi-Fi module to restore its default settings. By default the Wi-Fi module is configured to create an Ad-hoc network which devices can connect directly to without using a router or switch. The Ad-hoc network can be used to assist you in configuring the controller to connect up to your existing network. It is recommended that you do not operate the device long term in Ad-hoc mode as the Ad-hoc network is slower and has more overhead.

Example Command: `<FHWIFIDEFAULTS>`

Example Response:

```
!
SET
```

Appendix A – Device Types

A: Gemini Focuser
B: Gemini Rotator
C: *reserved*
D: *reserved*

Appendix B – Factory Defaults

Listed below are the factory default settings for the Gemini Focuser and Rotator configuration and status responses.

Gemini Focuser Defaults:
Focuser Configuration
```
Nickname = Focuser
MaxSteps = 115200
Dev Type = A
TComp On = 0
TCMode A = 86
TCMode B = 86
TCMode C = 86
TCMode D = 86
TCMode E = 86
CurrenTC = A
BLCompOn = 0
BLCSteps = 40
TC Start = 0
HOnStart = 1
END
```
Focuser Status
```
CurrTemp = +20.0
CurrStep = 57600
TargStep = 57600
IsMoving = 0
IsHoming = 0
Is Homed = 1
TempProb = 1
END
```

Gemini Rotator Defaults:
Rotator Configuration
```
Nickname = Rotator
Max Posi = 215999
Dev Type = B
BLCompOn = 0
BLCSteps = 40
PAOffset = 0
HonStart = 1
iReverse = 0
MaxSpeed = 800
```

```
END
```

Rotator Status

```
CurrStep = 45000
TargStep = 45000
CurentPA = 359999
TargetPA = 359999
IsMoving = 0
IsHoming = 0
Is Homed = 1
END
```

Gemini Hub Defaults:

Hub Status

```
Firmware = 1.0.0
LEDBrite = 75
HandCtrl = 0
Wired IP = 169.254.1.1
WiFi Mod = 0
WiFiConn = 0
WiFiFVOk = 0
WiFiFVer = 0.0.0
WiFiSSID =
WiFiAddr = 0.0.0.0
WiFiSecM = A
WiFiSecK =
END
```

Appendix C – Full Command List

Presented in this appendix is an abbreviated reference list for all valid commands. Full detailed descriptions are provided in the main document above.

Command Syntax	Command Description
`<FdiiGETDNN>`	Get Focuser NickName (ping focuser)
`<RdiiGETDNN>`	Get Rotator NickName (ping rotator)
`<FdiiGETSTA>`	Get Focuser Status
`<RdiiGETSTA>`	Get Rotator Status
`<FdiiGETCFG>`	Get Focuser Configuration
`<RdiiGETCFG>`	Get Rotator Configuration
`<HdiiGETCFG>`	Get Gemini Hub Configuration
`<FdiiDOHALT>`	Halt Focuser immediately
`<RdiiDOHALT>`	Halt Rotator immediately
`<FdiiDOHOME>`	Home Focuser
`<RdiiDOHOME>`	Home Rotator
`<FdiiCENTER>`	Move Focuser at center of travel
`<FdiiDOMOVEy>`	Begin Focuser Relative Move *(internal use: Hand Control command)*
`<RdiiDOMOVEy>`	Begin Rotator Relative Move *(internal use: Hand Control command)*
`<FdiiDOSTOP>`	Stop Focuser Relative Move
`<RdiiDOSTOP>`	Stop Rotator Relative Move
`<FdiiMOVABSzzz…>`	Begin Absolute Focuser Position Move
`<RdiiMOVEPAzzz…>`	Begin Absolute Rotator Position Angle Move
`<RdiiMOVABSzzz…>`	*(Reserved Command: Begin Absolute Rotator Step Position Move)*
`<FdiiSETDNNzzz…>`	Set Focuser NickName
`<RdiiSETDNNzzz…>`	Set Rotator NickName
`<FdiiSETDEVz>`	Set Focuser Device Type
`<RdiiSETDEVz>`	Set Rotator Device Type
`<FdiiSETHOSy>`	Set Focuser Home on Start
`<RdiiSETHOSy>`	Set Rotator Home on Start
`<FdiiSETTCEy>`	Set Temperature Compensation Enabled

`<FdiiSETTCMm>`	Set active Temperature Compensation Mode
`<FdiiSETTCCmszzzz>`	Set Temperature Compensation Coefficient
`<FdiiSETTCSy>`	Set Temperature Compensation at Start Enabled
`<FdiiSETBCEy>`	Set Focuser Backlash Compensation Enabled
`<FdiiSETBCSzz>`	Set Focuser Backlash Compensation Steps
`<RdiiSETBCEy>`	Set Rotator Backlash Compensation Enabled
`<RdiiSETBCSzz>`	Set Rotator Backlash Compensation Steps
`<RdiiSETREVy>`	Set Rotator Position Angle Reverse Property
`<HdiiSETLEDzz>`	Set Hub LED Brightness
`<HdiiRESETH>`	Reset Hub to Factory Defaults
`<HdiiREBOOT>`	Reboot Hub power
`<HdiiWFSSIDzzz…>`	Set WiFi SSID
`<HdiiWFSECMz>`	Set WiFi Security Mode
`<HdiiWFSECKnzzz…>`	Set WiFi Security Key
`<HdiiWFWEPKz>`	Set WiFi Security WEP Key Index
`<HdiiPUSHWF>`	Push WiFi Settings to Hub
`<HdiiWRESET`	Reset WiFi Module
`<HdiiWIFIRD>`	Restore WiFi Factory Defaults

Appendix D – Wi-Fi Security Modes

A: Open Network (no security)

B: WPA with Passphrase (8 to 63 ASCII characters)

C: WPA with Key (64 Hex characters)

D: WEP-40 (5 ASCII characters or 10 Hex characters)

E: WEP-104 (13 ASCII characters or 26 Hex characters)

Appendix E – Error Codes

The Gemini firmware performs various error checks on input commands. When an incorrectly formatted command is received the hub will respond with an error code.   A brief explanation of the various error codes is provided below.

Error 0 – Invalid Command Received

Example Command: `<xian;f>`

Example Response:
```
ERROR ID = 0
ERROR TEXT = The received command is formatted incorrectly
END
```

Error 1 – Invalid

Example Command: `<>`

Example Response:
```
ERROR ID = 0
END
```

Error 2 – Invalid Parameter Received

Example Command: `<R192MOVABS216000>`

Example Response:
```
!92
ERROR ID = 2
ERROR TEXT = The received command contained invalid parameters
END
```

Error 3 – Identifier Not Recognized

Example Command: `<>`

Example Response:
```
ERROR ID = 3
END
```

Error 4 – Invalid Target Device

Example Command: `<G123GETCFG>`

Example Response:
```
ERROR ID = 4
ERROR TEXT = The command received was for an invalid target device
END
```

Error 5 – Device is Homing

Example Command: <F123DOMOVE1>

Example Response:

```
!23
!
ERROR ID = 5
ERROR TEXT = The command is invalid because the device is homing
END
```