

## **CONTROL THROUGH THE RS232 PORT**

Upon power-up, the units on the chain may be numbered arbitrarily depending on the power up sequence. There may be multiple units with the same number and they will all execute the same commands. To solve this problem you must issue a renumber instruction after all the units in the chain are powered up and every time you add or remove a unit from the chain. The computer must not transmit any further data while the chain is renumbering or the renumbering routine may be corrupted.

When all units have renumbered, you may start issuing instructions over the RS232 connection.

Your communications settings must be: 9600 baud, no hand shaking, no parity, one stop bit. The amber LED is lit when there is activity on the RS232 lines. You can use this feature to try to determine which COM port you are connected to. When using a terminal program, make sure that it does not transmit carriage returns, line feed or other control characters that will interfere with command interpretation.

All instructions consist of a group of 6 bytes. They must be transmitted with less than 10 ms between each byte. If the unit has received less than 6 bytes and then a period of more than 10 ms passes, it ignores the bytes already received. We recommended that your software do something similar when receiving data from the actuators, especially in a noisy environment like a pulsed laser lab.

The following table shows the instruction format:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Device #	Command #	Data (least significant byte)	Data	Data	Data (most significant byte)

The first byte is the device number in the chain. Device number 1 is the unit closest to the computer, device number 2 is next and so forth. If the number 0 is used, all the units in the chain will execute the accompanying command simultaneously.

The second byte is the command number. Bytes 3,4,5,and 6 are data in long integer, 2's complement format with the least significant byte transmitted first. How the data bytes are interpreted depends on the accompanying command number. Complete details are given in the command reference on the following page.

Examples:        Move unit #5 to an absolute position of 257 micro-steps is: 5,20,1,1,0,0  
                  Move unit #2 to a relative position of -1 micro-step is: 2,21,255,255,255,255

1 micro-step=0.09921875  $\mu$ m. Make sure you carry at least 6 decimals in any calculations, otherwise the accuracy will be compromised.

Most compilers use 2's complement, least significant byte first, however some may not. If you experience problems with the actuator going to a strange position in place of what you requested, check how the particular compiler that you use sends the bytes on the serial port.

Some of the instructions cause the actuator to reply with a return code. It is also a group of 6 bytes. The first byte is the device #. Byte #2 is the instruction just completed or 255 if an error occurs. Bytes 3,4,5 and 6 are data bytes in the same format as the instruction data byte

---

## COMMAND REFERENCE

In June of 2001, firmware on all T-Series devices was upgraded to version 2.00. This upgrade offered many new commands and features in addition to those previously available, however the commands were rearranged for a more organized and expandable command set. Although every attempt was made to keep the new firmware backward compatible, we do not recommend continuing to use the old command set on new units. To determine which version of firmware you have simply issue the “return firmware version” command (51). If no reply is received you have a firmware version older than 2.00. To ensure your actuator is functioning well, you may first wish to issue command #10 which will result in a reply under all firmware versions. Most of the commands described in this manual will only function on units with firmware version 2.00 or higher. For older versions, you can download the command reference from the support section of our website.

Command	Description	Data Bytes	Reply Data
0	Reset. This sets the actuator to its powerup condition. It has the same effect as unplugging and restarting the unit. This is useful for resetting internal registers to their default values. The position stored in the actuator prior to this command will be lost, so you must save it and reload it if it is important.	Ignored.	None.
1	Home. The actuator retracts until it triggers its internal home switch. It then moves forward slightly to avoid accidentally triggering the home switch during use.	Ignored.	Absolute position (in this case 0)
2	Renumber. This command must always be issued with a 0 in Byte 1 (i.e. it must be issued to all units simultaneously). The unit closest to the computer becomes number 1. The next unit becomes 2 and so forth. Renumbering may take up to 1 second. The controlling computer must not send any commands during this time.	Ignored.	Each unit replies with its device ID after it finishes renumbering.
20	Move absolute. The actuator moves to the position given by the data bytes. The position must be within the range of the actuator.	Absolute position in micro-steps.	Absolute position. If the data is out of range the actuator will not move but will return 255 in byte 2 as well as the absolute position.
21	Move relative. The actuator moves to the position given by its position before the command plus the value in the data bytes. The final position must be within the range of the actuator.	Relative position (can be negative) in micro-steps.	Absolute position. If the data is out of range the actuator will not move but will return 255 in byte 2 as well as the absolute position.
22	Constant speed. The actuator moves at a constant speed selected by the data bytes until it is out of range or until a new instruction is received. If the requested step time is less than the minimum step time, the minimum step time is used. The relationship between the velocity and the data bytes is non-linear. Contact Zaber	Velocity. The velocity varies roughly from 0 to 250 um/s for data from 0 to 63 and from 750 to 4500 um/s for data from	Current speed. The unit may also be set to return its position continuously during constant velocity moves using the set mode command

	for further information if accurate velocity control is important to your application.	64 to 255.	(#40) bit 4.
23	Stop. If it is currently executing a constant speed command, the actuator will stop.	Ignored.	Current absolute position.
40	Set mode. This command allows setting several options. Each option is controlled by a single bit of Byte 3. All bits default to 0 at startup. The options available are described in detail at the end of this table.	Byte 3 is the mode configuration byte. All other bytes are ignored.	Reply data is simply an echo of the command data.
41	Set the maximum step time. When the actuator moves, it starts with a step time given by maximum step time (slow), and accelerates at the acceleration rate up to a maximum speed given by minimum step time (fast).	Byte 3 is the maximum step time in increments of approximately 0.05 ms. The default is 96 (4.8ms). Other data bytes are ignored.	Reply data is simply an echo of the command data.
42	Set the minimum step time. When the actuator moves, it starts with a step time given by maximum step time (slow) and accelerates at the acceleration rate up to a maximum speed given by minimum step time (fast). Note that step times less than 1.5 ms are likely to result in missed steps.	Byte 3 is the minimum step time. in increments of approximately 0.05 ms. The default is 48 (2.4ms) Other data bytes are ignored.	Reply data is simply an echo of the command data.
43	Set the acceleration rate. When the motor moves, it starts with a step time given by maximum step time (slow speed) and accelerates at the acceleration rate up to a maximum speed given by minimum step time.	Byte 3 is the acceleration rate in increments of about 0.05 ms per step per step. The default is 1 (0.05 ms/step/step). Other data bytes are ignored.	Reply data is simply an echo of the command data.
44	Set maximum extension limit. Use this command to limit the maximum extension to a value other than the default (28mm or 60mm depending on the model). Exercise caution since using this command it is possible to set the limit to a value greater than the physical limits of the actuator. This command is only available in firmware versions 2.09 and up.	The desired new maximum limit. Note that the least significant byte is ignored. The true limit will be within 4 steps (25.4 um) of the specified value	Reply data is simply an echo of the command data.
45	Set position. The internal register that holds the position is set to the value given by the data bytes. The phase of the stepper motor is controlled by the least significant byte of the position, thus the actuator may move by +/- 2 full steps (12.7 um) unless the set position corresponds to the true current position of the actuator. This command is useful if you want to turn off the system without losing position. Simply save the position in the controlling computer before powering down After power up, set the position back to the saved value. In	Absolute position.	Absolute position (in this case the position that was just set).

	this way you can continue without having to home the actuator.		
48	Set alias. This creates an alternate unit number for a device (in addition to it's actual unit number). By setting several aliases to the same value, you can control groups of units with a single instruction. When you send an instruction using an alias unit number, all devices with that alias will execute the instruction and reply using their actual unit numbers. To remove an alias, simply set it to zero. This command is only available in firmware versions 2.52 and up.	Byte 3 is the alias unit number. Valid numbers are between 0 and 254. To avoid confusion, it is best to choose an alias greater than the number of units connected, however this is not required.	Reply data is simply an echo of the command data.
50	Return device ID. This causes the unit to return an identification code indicating the type of device connected.	Ignored.	Device ID.
51	Return firmware version. This causes the unit to return its firmware version number. This is a number between 100 and 999. A decimal is assumed after the first digit. (ie 102 indicates firmware version 1.02)	Ignored.	Firmware Version.
52	Return power supply voltage. This causes the unit to return the voltage level of its power source in increments of 0.1 Volts. (Ex. 127 indicates 12.7 V).	Ignored.	Power supply voltage.
53	Return setting. This causes the unit to return the current value of the setting number (40, 41, 42, 43, or 45) specified in Byte3. The unit will reply with the setting number in Byte2, as if a command to change the setting had just been issued, however the setting will not be changed.	Setting Number: 40 – mode 41 – max steptime 42 – min steptime 43 – acceleration 44 – max limit 45 – position	Setting number in Byte2. Current setting value in Byte3.
60	Return position. This causes the unit to return it's current micro-step position.	Ignored.	Current absolute position.

**Operating Mode Options:**

Command #40, "Set mode" allows setting several options that affect the operation of the actuator. The effects of these options are described in the following table. All these settings default to 0 on power up or reset.

Byte 3 bit	Option Name	Effect
0 (1s)	Disable Auto-reply	A value of 1 disables ALL replies except those to "return" commands (commands 50 and higher).
1 (2s)	Anti-backlash	A value of 1 enables the anti-backlash routine. On negative moves (retracting), the actuator will overshoot the desired position by 60 $\mu\text{m}$ , reverse direction and approach the requested position from below. On positive moves (extending), the actuator behaves normally. Care must be taken not to crash the moving payload into a fixed object due to the 60 $\mu\text{m}$ overshoot on negative moves.
2 (4s)	Anti-sticktion	A value of 1 enables the anti-sticktion routine. On moves less than 60 $\mu\text{m}$ , the actuator will first retract to a position 60 $\mu\text{m}$ less than the requested position and approach the requested position from below. Care must be taken not to crash the moving payload

		into a fixed object due to the 60 $\mu\text{m}$ negative move.
3 (8s)	Disable Potentiometer	A value of 1 disables the potentiometer preventing manual adjustment of the device.
4 (16s)	Enable Constant Speed Position Tracking	A value of 1 enables position tracking during constant speed commands. The unit will return its position periodically when a constant speed command is executed. The Disable Auto-Reply option above takes precedence over this option. This feature is available in firmware versions 2.04 and higher.
5 (32s)	Disable Manual Position Tracking	A value of 1 disables automatic position replies during manual moves. The Disable Auto-Reply option above takes precedence over this option. This feature is only available in firmware versions 2.09 and higher.
6 (64s)	Enable Logical Channels Communications Mode	A value of 1 enables the Logical Channels Communications Mode. In this mode of communication, only bytes 3 thru 5 are used for data. Byte 6 is used as an IDbyte that the user can set to any value they wish. It will be returned unchanged in the reply. This feature is only available in firmware versions 2.09 and higher.

## **REPLY-ONLY COMMAND REFERENCE**

In general, an actuator will reply to an instruction using the same command number as the instruction itself. However, there are occasions (such as when the user turns the potentiometer) when the actuator may transmit data without first receiving a request from the controlling computer. This type of reply may be considered to be a triggered reply as opposed to a requested reply. In this case the actuator uses a “reply-only” command number to distinguish the reply from those requested by the controlling computer. The meanings of these numbers are shown in the table below.

Command #	Description	Data Bytes	Reply Data
8 (reply-only)	Constant speed position tracking. This reply is triggered periodically during a constant speed move, it occurs only if constant speed position tracking has been enabled (by setting bit 4 with the “Set Mode” command.)	N/A. Reply-only	Position in micro-steps.
10 (reply-only)	Manual position change. This reply is triggered when the potentiometer is turned causing the actuator position to change.	N/A. Reply-only	Position in micro-steps.
14 (reply-only)	Power supply voltage out of range. This reply is triggered when the power supply voltage moves out of range. If your power supply is not capable of enough current to drive the devices connected to it, you may see this reply.	N/A. Reply-only	Measured power supply voltage in increments of 0.1V (ie 105 = 10.5V)
255 (reply-only)	Command position out of range. This reply is triggered by a command containing data that is outside the acceptable range for the actuator.	N/A. Reply-only	Current absolute position in micro-steps.