

Command:TTL

TTL Has different syntax and slightly different function if you have a Tiger(TG-1000) or MS2000 and RM2000 controller.

On Tiger

Format	[Addr#]TTL [X=IN0_mode] [Y=OUT0_mode] [Z=aux_IO_state] [F=OUT0_polarity] [R=aux_IO_mask] [T=aux_IO_mode] [Addr#]TTL (firmware v3.16+)
Type	Card-Addressed
Remembered	Using [Addr#]SS Z



Only partly implemented in TG-1000

The MS2000 controller has a buffered TTL input (IN0) and output (OUT0) port as well as several unbuffered I/O ports. The signals IN0 and OUT0 are found on the board connector SV1 pin1 and 2 respectively. On many controllers, these signals are connected to the IN and OUT BNC connectors on the back of the controller. The IN0_mode and OUT0_mode parameters set with this command determine the character of the I/O pins. It's a Card-Addressed command.

IN0_mode

0 - turns off TTL IN0 controlled functions; TTL interrupt DISABLED.

1 - TTL IN0 initiates a Move-to-Next-Position of the stored positions in the Ring Buffer pointed to by the buffer_pointer. When the buffer_pointer reaches a value equal to the number of saved positions, it resets to the first position, allowing cyclic repetitions to the saved locations. See RBMODE and LOAD command.

2 - TTL IN0 repeats most recent relative move (See MOVREL) For example, begin a session by issuing the command MOVREL X=0 Y=0 Z=0.5, and each subsequent move to Next Position will cause the Z axis to move 0.05 micron. This function can be used for repetitive relative moves of any axis or combination of axes. You may directly set the dZ value with the ZS command's X parameter.

3 - TTL IN0 initiates an autofocus operation on systems with autofocus installed.

4 - enables TTL IN0 controlled Z-stacks. (See [ZS command](#)).

5 - enables TTL IN0-started position reporting via the serial interface. Information is asynchronously sent out the serial interface every report_time interval, where report_time is set with the RT command. Data returned in the serial stream are the elapsed time in milliseconds since the TTL trigger, followed by the position of each axis enable by the axis_byte. On TRACKING systems, the PMT sum signal is also reported. Reporting is toggled on and off by the TTL input pulse.

6 - TTL interrupt ENABLED; use with TTL triggered position reporting.

7 - TTL commanded ARRAY move to next position.

9 - Used with CRISP focus lock. TTL IN0 HIGH engages lock if the system is in the READY state. TTL IN0 LOW will cause the system to UNLOCK is locked already.

12 - Behavior is exactly the same as mode 1 above except the moves are relative rather than absolute.

CAUTION: If you are using TTL mode 12, the values entered into the ring buffer using the [LOAD command](#) represent RELATIVE coordinates, not ABSOLUTE coordinates. You must drive the stage to the appropriate starting position before triggering a ring buffer sequence.

20 - Used with TGLED card, All LED channels remain off until a TTL pulse is received on the TTL IN connector. On the Rising edge of the TTL pulse, the LED channels turn ON and remain ON for a fixed duration set by the RT Y command.

21 - Used with TGLED card, TTL Pulses cycle thru channels of the TGLED card one at a time turning them ON for a set time. On the Rising edge of the TTL pulse, only ONE LED channel turns ON for a fixed time. Then on next TTL pulse, the next LED channel turns ON for a fixed time and so on. Starting v3.24, when [RB F=3](#) TGLED card cycles thru all channels without waiting for TTL pulses.

OUT0_mode

0 - TTL OUT0 unconditionally set LOW.

1 - TTL OUT0 unconditionally set HIGH.

2 - generates TTL pulse at end of a commanded move (MOVE, MOVREL, move via ring buffer, or via array module). The pulse duration is set with command RT Y=???.

3 - output TTL OUT0 gated HIGH during axis index 0 (X) constant speed move.

4 - output TTL OUT0 gated HIGH during axis index 1 (Y) constant speed move.

5 - output TTL OUT0 gated HIGH during axis index 2 (Z) constant speed move.

8 - TTL OUT0 timed arrival pre-pulse output. See RT command. Requires PREPULSE firmware module.

9 - TTL OUT0 PWM and MicroServo Output. See the LED or the SECURE command. Requires LED_DIMMER or USERVO firmware module

10 - output TTL OUT0 gated HIGH upon completion of video AUTOFOCUS function. AUTOFOCUS hardware and firmware required.

11 - generates TTL OUT0 pulse at end of commanded move providing CRISP is in 'F' state. Waits for CRISP 'F' state after move completion to send a pulse. Also enables CRISP status busy polling that will not return 'N' until CRISP is in the In-Focus 'F' state.

12 - TTL OUT0 high when CRISP is 'F' state, low otherwise.

20 - TTL OUT0 set during SPIM state machine operation. Requires MM_SPIM firmware module.

Until v3.12 was code 10 (changed to avoid MS2000 conflict)

21 - TTL OUT0 set at the end of a ring buffer move or AIJ-initiated move (for laser trigger). Requires MM_TARGET firmware module. Until v3.12 was code 11 (changed to avoid MS2000 conflict).

22 - In this mode, TTL OUT0 is controlled by [Single-axis function](#) module. With the [SAP command](#) the user can generate a TTL pulse that is synchronized with [Single-axis function](#) actuator motion. Available only in firmware version 3.17 and above.

OUT0_polarity

1 - default polarity, -1 inverts the polarity of TTL OUT0.

aux_IO_state

Requires TTL_AUXILIARY firmware module; behavior depends on the firmware build and hardware present. Sets the state of the auxiliary TTL output according to the aux_IO_mask. Input and output as a decimal number representing the binary pattern desired. The following uses have been defined so far:

For MM_SPIM firmware with SPIM TTL card: Bit0 = Side0/Laser0 output, Bit1 = Side1/Laser1 output

aux_IO_mask

Requires TTL_AUXILIARY firmware module; behavior depends on the firmware build and hardware present. Controls how the aux_IO_state bits are used, or how the backplane is used when aux_IO_mode is set to 2. Input and output as a decimal number representing a binary mask. If the corresponding mask bit is set to 1 then the state bit will be reflected at the output, but if the mask bit is 0 then the state bit has no effect. The following uses have been defined so far:

For MM_SPIM firmware: Defaults to 3 = 0b00000011.

aux_IO_mode

Requires TTL_AUXILIARY firmware module; behavior depends on the firmware build and hardware present. The SPIM state machine overrides these setting during its operation.

0 - TTL outputs determined by aux_IO_state/mask.

1 - TTL outputs determined by the LED command (requires MM_LASER_TTL module). The default setting for MM_SPIM firmware.

2 - Simulates a TTL input from the backplane. The backplane value is masked by aux_IO_mask and the binary value is considered. If a 0-1 transition occurs then a TTL input pulse is simulated and action will be taken depending on the setting of IN0_mode. The default setting for TTL_AUXILIARY on piezo firmware.

Without any Argument

In firmware version 3.16 and above , When TTL command is issued without any arguments , like [Card Addr#]TTL , the controller report the state of TTL IN. The controller replies with :A 0 when

signal is low. and :A 1 when TTL IN sees a high signal.

On MS2000 and RM2000

Format	TTL [X=IN0_mode] [Y=OUT0_mode] [Z=aux_IO_mode] [F=OUT0_polarity] TTL (firmware v9.2k+)
Remembered	Using SS Z
Required Version	8.5+

The MS2000 controller has a buffered TTL input (IN0) and output (OUT0) port as well as several unbuffered I/O ports. The signals IN0 and OUT0 are found on the board connector SV1 pin1 and 2 respectively. On many controllers, these signals are connected to the IN and OUT BNC connectors on the back of the controller. The IN0_mode and OUT0_mode parameters set with this command determine the character of the I/O pins.

IN0_mode

0 - turns off TTL IN0 controlled functions; TTL interrupt DISABLED.

1 - TTL IN0 initiates a Move-to-Next-Position of the stored positions in the Ring Buffer pointed to by the buffer_pointer. When the buffer_pointer reaches a value equal to the number of saved positions, it resets to the first position, allowing cyclic repetitions to the saved locations. See [RBMODE](#) and [LOAD](#) commands.

2 - TTL IN0 repeats most recent relative move (See MOVREL) For example, begin a session by issuing the command MOVREL X=0 Y=0 Z=0.5, and each subsequent move to Next Position will cause the Z axis to move 0.05 micron. This function can be used for repetitive relative moves of any axis or combination of axes. You may directly set the dZ value with the ZS command's X parameter.

3 - TTL IN0 initiates an autofocus operation on systems with autofocus installed.

4 - enables TTL IN0 controlled Z-stacks. (See ZS command).

5 - enables TTL IN0-started position reporting via the serial interface. Information is asynchronously sent out the serial interface every report_time interval, where report_time is set with the RT command. Data returned in the serial stream are the elapsed time in milliseconds since the TTL trigger, followed by the position of each axis enable by the axis_byte. On TRACKING systems, the PMT sum signal is also reported. Reporting is toggled on and off by the TTL input pulse.

6 - TTL interrupt ENABLED; use with TTL triggered position reporting.

7 - TTL commanded ARRAY move to next position.

9 - Used with CRISP focus lock. TTL IN0 HIGH engages lock if the system is in the READY state. TTL IN0 LOW will cause the system to UNLOCK if locked already.

10 - Toggle TTL OUT0. If TTL OUT0 is set either LOW or HIGH, an input pulse on the TTL IN0 will cause the output to toggle to the other state.

20 - On TTL IN0 trigger (or on [RM command](#)) TTL OUT0 outputs a pulse for a fixed duration set by the [RT Y command](#). TTL Y must be in 0 mode (TTL Y=0) or it might cause an issue. Can be

used with a LED illumination, acts like a FLASH when taking images. (added in firmware version 9.2I)

In case of a Dual LED(DLED), All LED channels remain off until a TTL pulse is received on the TTL IN connector. On the Rising edge of the TTL pulse, the LED channels turn ON and remain ON for a fixed duration set by the [RT Y command](#).

21 - Used with Dual LED(DLED), TTL Pulses cycle thru channels of the TGLED card one at a time turning them ON for a set time. On the Rising edge of the TTL pulse, only ONE LED channel turns ON for a fixed time. Then on next TTL pulse, the next LED channel turns ON for a fixed time and so on. Duration LEDs stay on set by the [RT Y command](#)

OUT0_mode

0 - TTL OUT0 unconditionally set LOW.

1 - TTL OUT0 unconditionally set HIGH.

2 - generates TTL pulse at end of a commanded move (MOVE, MOVREL, move via ring buffer, or via array module). The pulse duration is set with command RT Y=???.

3 - output TTL OUT0 gated HIGH during axis index 0 (X) constant speed move.

4 - output TTL OUT0 gated HIGH during axis index 1 (Y) constant speed move.

5 - output TTL OUT0 gated HIGH during axis index 2 (Z) constant speed move.

8 - TTL OUT0 timed arrival pre-pulse output. See RT command. Requires PREPULSE firmware module

9 - TTL OUT0 PWM and MicroServo/Solenoid Output. See the LED or the SECURE command. Requires LED_DIMMER or USERVO or SOL_LK firmware module

10 - output TTL OUT0 gated HIGH upon completion of video AUTOFOCUS function. AUTOFOCUS hardware and firmware required.

11 - generates TTL OUT0 pulse at end of commanded move providing CRISP is in 'F' state. Waits for CRISP 'F' state after move completion to send a pulse.

12 - TTL OUT0 high when CRISP is 'F' state, low otherwise

OUT0_polarity

1 - default polarity, -1 inverts the polarity of TTL OUT0.

Without any Argument

In firmware version 9.2k and above , When TTL command is issued without any arguments , like TTL , the controller report the state of TTL IN. The controller replies with :A 0 when signal is low. and :A 1 when TTL IN sees a high signal.

aux_IO_mode

Not used

[commands](#), [led](#), [tiger](#), [ms2000](#), [tgled](#), [crisp](#), [t1](#), [spim](#), [array](#), [sequencer](#)

From:

<http://www.asiimaging.com/docs/> - **Applied Scientific Instrumentation**

Permanent link:

<http://www.asiimaging.com/docs/commands/t1>

Last update: **2019/04/18 23:33**

