

# Command:ZS

MS2000 or RM2000 syntax

<b>Shortcut</b>	ZS
<b>Format</b>	ZS [X=dZ] [Y=num_slices] [Z=mode] [F=stack_timeout] [M=stack_state] [T?]
<b>Remembered</b>	Using SS Z

Tiger syntax

<b>Shortcut</b>	ZS
<b>Format</b>	[addr#]ZS [X=dZ] [Y=num_slices] [Z=mode] [F=stack_timeout] [M=stack_state] [T?]
<b>Type</b>	Card-Addressed
<b>Remembered</b>	Using [addr#]SS Z

## Required Firmware Modules:

- IN0\_INTERRUPT (need to use [TTL X=4](#) for Z-stacks)

## Incompatible Firmware Modules:

- TTL\_REPORT\_INT

## Parameters:

**X [dZ]:** The amount to move each slice in 10ths of microns (ASi units). For example, the default value of the units multiplier is UM Z=10000, so ZS X=50 is 5 microns per step.

If the unit multiplier has been changed with the [UM](#) command you will have to adjust the value that you use for dZ. For example, if UM Z=1000000 then ZS X=5000 is 5 microns per step.

**Y [num\_slices]:** The number of slices in the Z-stack, the maximum value is 32767. Prior to MS2000 v9.50 and Tiger v3.43 the maximum number of slices is 127.

**Z [mode]:** Select between a sawtooth or triangle waveform for stage motion for repeated stacks. The default mode is the sawtooth waveform, i.e. repeating stacks always in the same direction.

ZStack Mode	
0	Sawtooth waveform
1	Triangle waveform

**F [stack\_timeout]:** The maximum amount of time in milliseconds between TTL input pulses. The stage will return to the starting position of the Z-stack after stack\_timeout ms has elapsed without a TTL input trigger.

The default stack\_timeout is 500, and the max value is 32767.

MS-2000 v9.55 or Tiger v3.53 required

**T [stack\_index]:** Read the current stack\_index with ZS T?, the index starts at 0 and goes up to num\_slices-1.

MS-2000 v9.55 or Tiger v3.54 required

**M [stack\_state]:** Read the current stack state as an integer. The Z-stack is active if the return code is not 0.

Sending ZS M=0 exits the Z-stack and returns to the initial position. This is the only valid state to send to ZS M=#.

ZStack State		
Code	State	Notes
0	START	Idle
1	UP	Active
2	DOWN	Active (Triangle waveform only)

### Command Description:

This command sets parameters for use with TTL triggered Z movement. User must also set [TTL X=4](#) for this trigger mode to be active. When a positive TTL edge is detected, the focus axis is moved by an amount dZ (expressed in 10ths of microns). Note that internally the amount dZ is actually stored as a multiple of the encoder unit, e.g. ~22 nanometers for a 4 TPI stage with rotary encoders, or 10 nanometers exactly for most linear encoded stages. This move distance is repeated for num\_slices TTL triggered moves. If mode=1, the stage will step in the opposite direction for n moves, then turn around again, repeating a triangular waveform cycle. If mode=0 the stage will return to the original position after num\_slices moves and repeat a saw-tooth waveform cycle.

The current position when the first TTL pulse is received becomes the center of the stack as well as the position that the stage returns to after the timeout duration.

The stack begins with a move to the negative extreme if dZ is positive and moves in increasing position. To reverse the polarity use a negative dZ.

Note that the total range traversed during the stack is  $dZ * (num\_slices - 1)$ .

### Changing The Axis Under Control:

The axis moved by the TTL is the designated "focus index" (also the axis used for CRISP among other things). Use [UNLOCK F \(UL F\)](#) to read or set the axis letter corresponding to "focus index". Note the setting has to be changed, settings saved, and the controller reset or power cycled for the new setting to take effect. If the controller has a piezo but no motorized focus drive then the piezo axis should be set as the "focus index". If both are present the "focus index" normally defaults to the piezo. Prior to Tiger v3.44 the ZS command would always move the z index rather than the focus index.

This is not the only way to perform a Z-stack using ASI hardware, the [ring buffer module](#) can be setup to do Z-stacks as well.



**Backlash:** The Z-stack routine also performs the backlash compensation move for each step. For step size smaller than 10 microns, this might result in issues like irregular step sizes. Consider disabling [BACKLASH \(B\)](#) for smaller step sizes.



**Stack Timeout:** If the TTL frequency is less than 2Hz, then the controller might consider it a stack time out condition. Consider increasing the stack timeout to accommodate the slower TTL frequency, to avoid any issues.

## Reply

If there are no errors, a positive reply of :A will be returned. Example

## Example

Setup to do twenty 1 micron slices with triangular pattern.

### Tiger Example

The focus drive axis is at card address 2

```
2ZS X=10 Y=20 Z=1
:A
```

### MS2000 Example

```
ZS X=10 Y=20 Z=1
:A
```

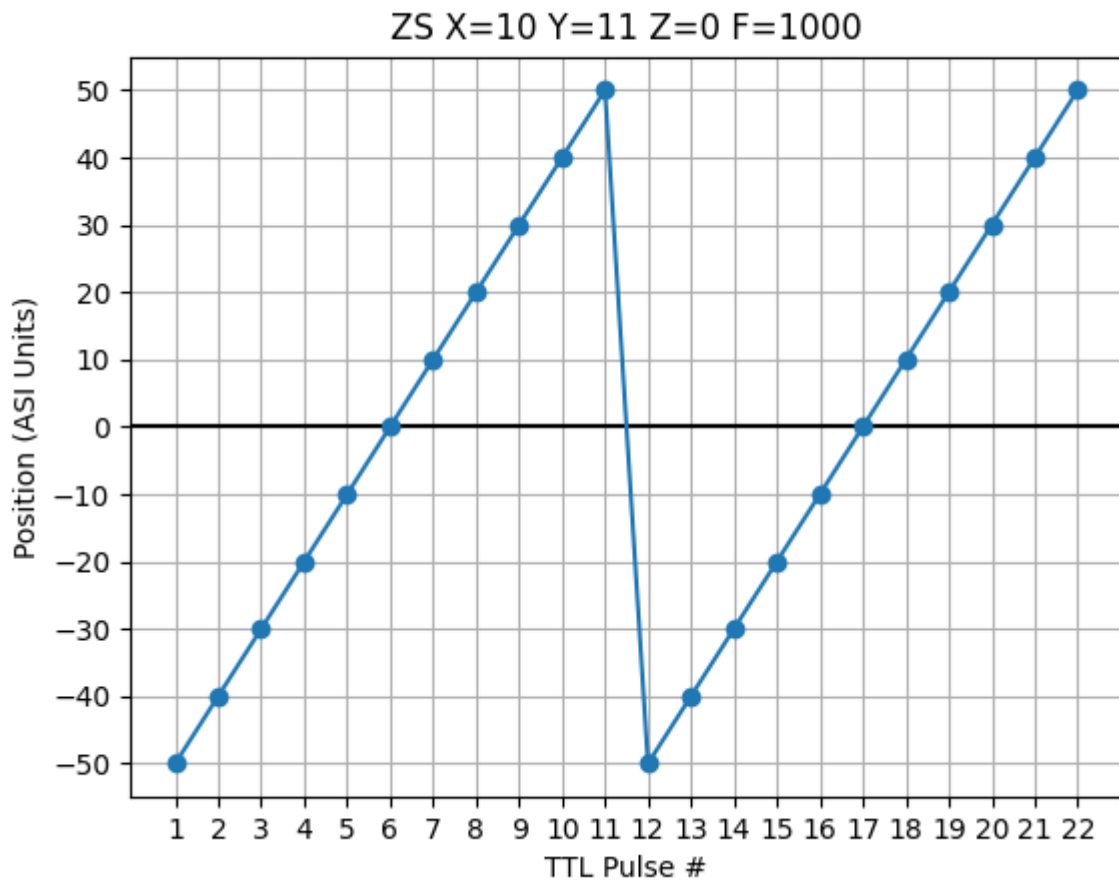
Examples of expected behavior:

Graphs of the Z position data

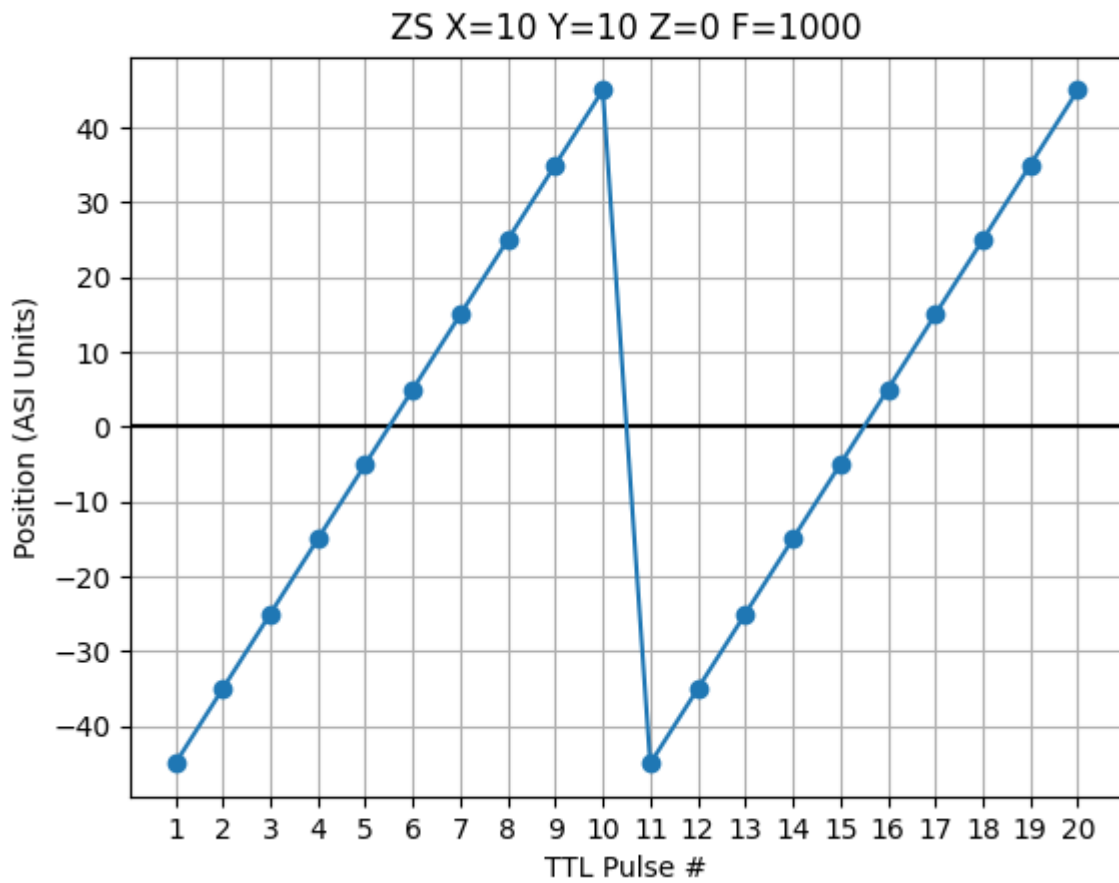
The center of the Z-stack is 0 in the following graphs.

## Sawtooth Waveform

Use an odd num\_slices if you want the center point to be on the starting position.

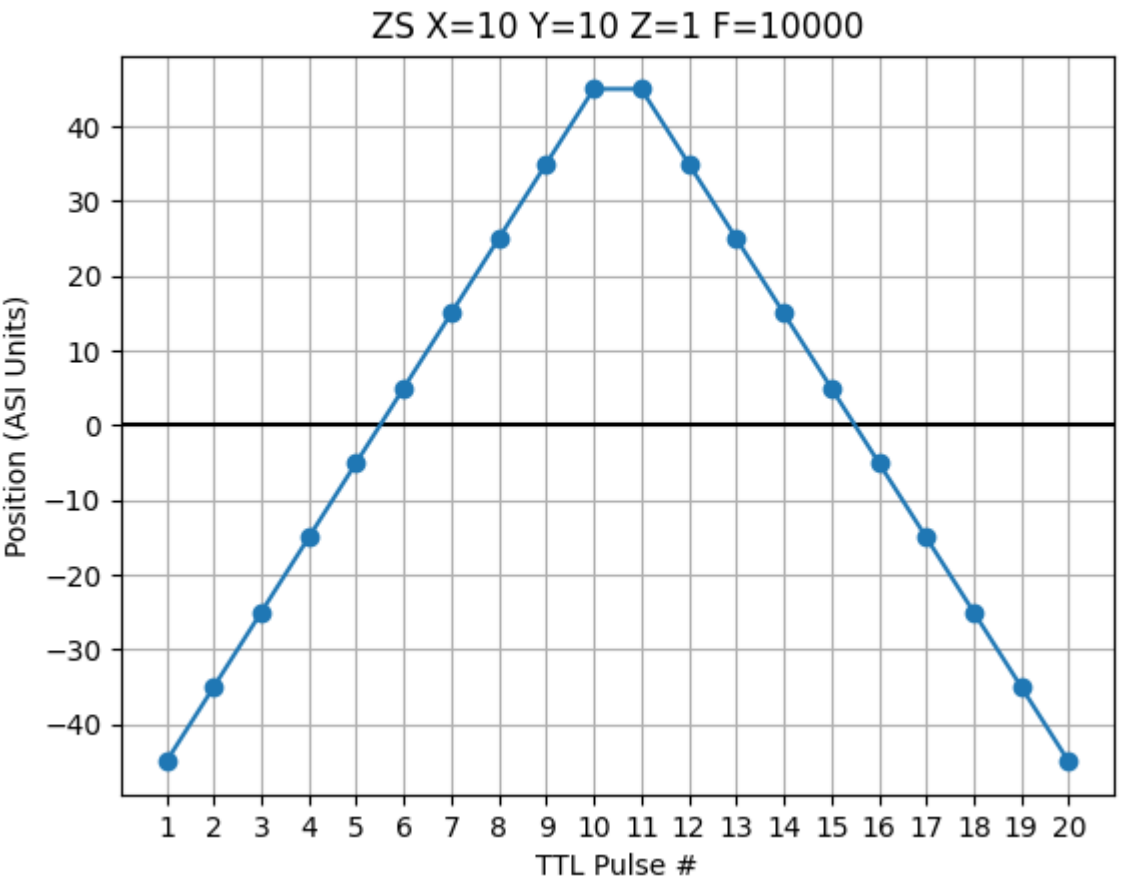
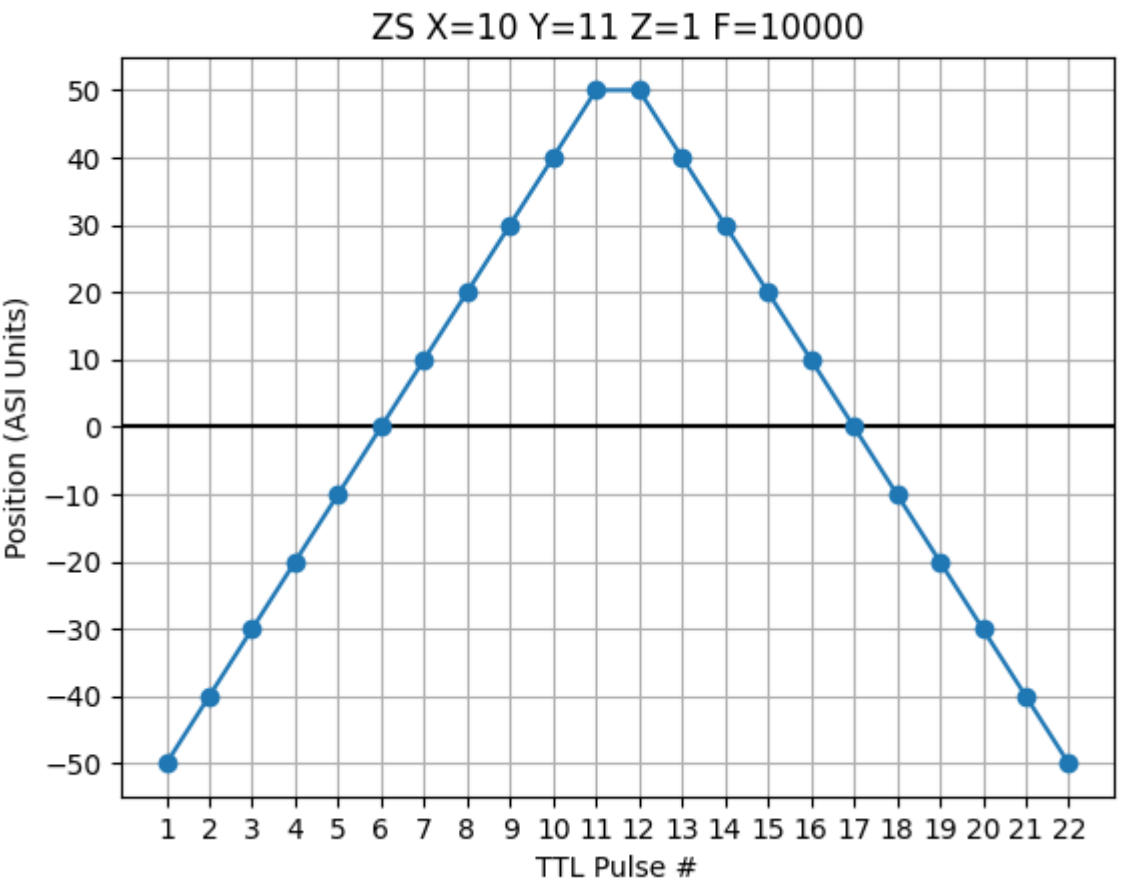


An even num\_slices is symmetrical around the starting position.



### Triangle Waveform

The triangle waveform swaps starting positions every Z-stack.



**Version History**

Version		
MS	TG	Description
9.55	3.53	add ZS T? to read the stack index
9.54	3.53	fixed triangle waveform behavior
9.54	3.52	odd num_slices regression from (9.51, 3.45) fixed
9.51	3.45	even num_slices are symmetric around start position
9.50	3.43	max num_slices increased from 127 to 32767
9.2j	3.44	move the focus_index instead of z_index

[commands](#), [tiger](#), [ms2000](#)

From:

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

Permanent link:

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

Last update: **2025/10/08 13:36**

