

Command:BCUSTOM (BCA)

MS2000 or RM2000 syntax

| | |
|---------------------------|--|
| Shortcut | BCA |
| Format | BCA [X = @ Normal Press] [Y = @ Long Press] [Z = @ Extra Long Press] [F = Home Long Press] [T = Home Extra Long Press] [R = JS Normal Press] [M = JS Long Press] |
| Units | Integer code (see table below) |
| Firmware Version Required | 9.2g+ |
| Remembered | Automatically on power down. Not affected by SS Z or SS X. Before version 9.2m, SS Z used to save settings. |

Tiger syntax

| | |
|---------------------------|---|
| Shortcut | BCA |
| Format | [addr#]BCA [X = @ Normal Press] [Y = @ Long Press] [Z = @ Extra Long Press] [F = Home Long Press] [T = Home Extra Long Press] [R = JS Normal Press] [M = JS Long Press] |
| Units | Integer code (see table below) |
| Type | Card-Addressed |
| Firmware Version Required | 3+ |
| Remembered | Automatically on powerdown. Not affected by [addr]SS Z or [addr]SS X. Before version 3.3, [addr]SS Z used to save settings. |

In Tiger version 3.0+ and MS2000 version 9.2g+ the Button Function assignment has been rewritten to be more flexible. Every possible button function is now assigned a number. This function can be assigned to any button (@, Home, and Joystick Button) and any press duration (Normal, Long and Extra Long Press) through the BCA commands X, Y, Z, F, T, R, and M arguments.

The settings of BCUSTOM are automatically saved in non-volatile memory when changed, they will be available even on controller restart.

As of Tiger version 3.18 and MS2000 version 9.2l a button function can be initiated over serial using the BE F command ([BENABLE](#)). The function doesn't need to be assigned to a particular button for this to work. Note: this function does not interact with the button_flag_byte, use [EXTRA M=#](#) if you need that functionality.

Note: Behavior of this command is very different pre-Tiger version 3.0 and MS2000 version 9.2g

As of MS000 version 9.2m+ and Tiger v3.35+, it's possible to assign button functions to Home Normal Press and Joystick Extra Long Press using the [BENABLE](#) command. The parameters **R** and **T** are used, and follow the same behavior as BCUSTOM. The parameter **M** to sets the Zero/Halt Normal Press button function with [BENABLE M=#](#).

Note: The Zero/Halt button will always halt all axes on a button press unless you set the button function to 0 - No Function Performed, which disables the halting routine. The halting routine happens as soon as you press the button, not in the release handler.

Example: #BENABLE R=0 to disable the home button normal press, where # is the Tiger card address.

Button Press Duration:

The amount of time that a button is held down determines the type of button press.

- * 0 to 1 seconds: Normal Press
- * 1 to 3 seconds: Long Press
- * 3 or more secs: Extra Long Press

Button Function Table

The table below lists and describes all possible button functions

| No | Function |
|----|--|
| 0 | No function performed |
| 1 | Smart Move related This function no longer exists, removed in Tiger 2.5 and MS2000 9.2p |
| 2 | Toggles Knob between two axis, like Z and F |
| 3 | TRACKING, CRISP, LSRTK related, short press functions. Steps from IDLE to Calibration states to lock and unlock state. |
| 4 | CLOCKED DEVICE related, moves clocked devices like Turret and slider to next position. Note, if there are multiple clocked devices on the same card or controller, all of them are moved. For more specificity see 36 and 37 below |
| 5 | ARRAY MODULE related, same as Command:ARRAY (AR) . When TTL X=7 does move to next array position. |
| 6 | RING BUFFER related, move to next ring buffer position; On TGLED card, button press mimics TTL pulse version 3.24+ . |
| 7 | SCAN MODULE related, halts scan move |
| 8 | AUTOFOCUS related, performs autofocus routine. |
| 9 | CRIFF related, toggle CRIFF lock This function no longer exists, CRIFF was replaced by CRISP. |
| 10 | AT_XYZ_KNOB related, changes xyz knob state. Cycles knob control between x,y,z axis. |
| 11 | AT_XYZF_KNOB related, changes xyzf knob state. Cycles knob control between x,y,z and f axis. |
| 12 | ZLOADER related, Moves Z away from the sample, to its upper limit, and back. For loading samples. |
| 13 | CRISP and TRACKING related, performs CRISP very long press functions |
| 14 | ADEPT related, toggles between piezo external and internal input mode |
| 15 | CRISP and TRACKING related, performs CRISP long press functions. Stops current CRISP operation like stop dither, unlock etc. |
| 16 | ARRAY MODULE related, puts ARRAY Module in start state |
| 17 | CRIFF related, change CRIFF state. This function no longer exists, CRIFF was replaced by CRISP. |
| 18 | RING BUFFER related, loads current stage positions of all axes into ring buffer |
| 19 | AT RAMM LOAD related, moves Y and F axis to their upper and lower limits. |

| No | Function |
|----|--|
| 20 | SCAN MODULE related, puts SCAN MODULE is start state |
| 21 | AUTOFOCUS related, performs autofocus calibration |
| 22 | ZOOM related, sets up zoom profile |
| 23 | PLANAR CORRECTION related, sets planar correction points |
| 24 | RING BUFFER related, clears ring buffer |
| 25 | TOGGLE ALL AXES related, swaps knob control between two axes. |
| 26 | TRACKING related, short press function |
| 27 | JS PULSE related, sets TTL in ready state |
| 28 | JS_FASTSLOW related, toggles joystick speed between fast and slow |
| 29 | PLANAR CORRECTION related, resets planar correction |
| 30 | CLOCKED DEVICE related, moves clocked devices like Turret and slider to previous position. Note, if there are multiple clocked devices on the same card or controller, all of them are moved. For more specificity see 38 and 39 below |
| 31 | JS_FASTSLOW related, sets joystick speed to fast |
| 32 | JS_FASTSLOW related, sets joystick speed to slow |
| 33 | LED related, increase LED intensity |
| 34 | LED related, decrease LED intensity |
| 35 | LED related, toggle LED ON and OFF |
| 36 | CLOCKED DEVICE related, moves the FIRST clocked devices like Turret and slider to next position. Similar to option 4 |
| 37 | CLOCKED DEVICE related, moves the SECOND clocked devices like Turret and slider to next position. Similar to option 4 |
| 38 | CLOCKED DEVICE related, moves the FIRST clocked devices like Turret and slider to previous position. Similar to option 30 |
| 39 | CLOCKED DEVICE related, moves the SECOND clocked devices like Turret and slider to previous position. Similar to option 30 |
| 40 | HOME related, issues a Command:HOME on all axis motors MS2000 9.2n and Tiger v3.36 firmware |
| 41 | ZERO related, issues a Command:ZERO on all axes. MS2000 9.2n and Tiger v3.36 firmware |
| 42 | Repeats the last issued relative move Command:MOVREL MS2000 9.2o firmware |

Example:

Take a firmware build like STD_XY. This firmware is packed with modules like,

- Ring Buffer module
- JS_FASTSLOW, joystick button press toggles joystick speed between fast slow.

In a built like this there are a lot of modules fighting for control of the buttons. There is a priority list that sets up the defaults button function . Lets Query the controller on what the default assignments ended up being.

Tiger Example

```
1BCA X? Y? Z? F? T? R? M?
X=0 Y=0 Z=0 F=0 T=0 R=28 M=18
X: @ Normal
Y: @ Long
Z: @ Ext Long
```

```
F: Home Long
T: Home Ext Long
R: Js btn Normal
M: Js btn Long
<LF>
```

MS2000 Example

```
BCA X? Y? Z? F? T? R? M?
X=0 Y=0 Z=0 F=0 T=0 R=28 M=18
X: @ Normal
Y: @ Long
Z: @ Ext Long
F: Home Long
T: Home Ext Long
R: Js btn Normal
M: Js btn Long
<LF>
```

So @ normal press, does nothing @ long press, does nothing @ extra long press, does nothing Home long press, does nothing Home extra long press, does nothing Joystick button normal press, toggles joystick speed Joystick button long press, loads current position into ring buffer.

If user wants the Ring Buffer to be more prominent, then following command can be issued.

Tiger Example

```
1BCA X=6 F=24 R=18 M=28
:A
<LF>
```

MS2000 Example

```
BCA X=6 F=24 R=18 M=28
:A
<LF>
```

Now, @ normal press, moves to next ring buffer position Home long press, clear the ring buffer Joystick button normal press, loads current stage position into ring buffer Joystick button long press, and toggles joystick speed.

Tiger example scenario

```
30: Comm v3.26 TIGER_COMM Apr 09 2019:15:52:41
31: X:XYMotor,Y:XYMotor v3.25 STD_XY_LED Mar 18 2019:15:56:27
32: Z:ZMotor,S:Slider v3.26 STDZ_SLDR8 Apr 30 2019:08:40:35
```

```
33: 0:Tur v3.26 TURRET_SINGLE Apr 30 2019:08:41:10
```

A user with a Tiger system in the above configuration has two clocked position devices. A filter slider on card #2 and turret on card#3. They would like to move these devices with button presses. Lets set it up so that on short '@' button press (less than 1-sec press) Filter slider moves, then on long '@' button press (1 to 3-sec press) the turret moves. The button function that does this is '# 4' (see the table above). '@' short press assigned with 'X' variable, and '@' long press is assigned with 'Y' variable.

2BCA X=4 Y=0 , this tells card#2 which drives the filter slider to respond to short @ press with slider move, and ignore any long @ button presses.

3BCA X=0 Y=4 , this tells card#3 which drives the turret to respond to long @press with turret move and ignore any short @ button presses.

1BCA X=0 Y=0 , this tells card#1 which drives XY stage to ignore any short or long @ button presses.

Try it out. If you like the behavior, commit it to systems onboard memory with serial commands

```
1SS Z  
2SS Z  
3SS Z
```

More info on SS Z command is here, http://www.asiimaging.com/docs/commands/saveset_commands_tiger_ms2000

From:

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

Permanent link:

<https://www.asiimaging.com/docs/commands/bcustom>

Last update: **2023/08/31 02:11**

