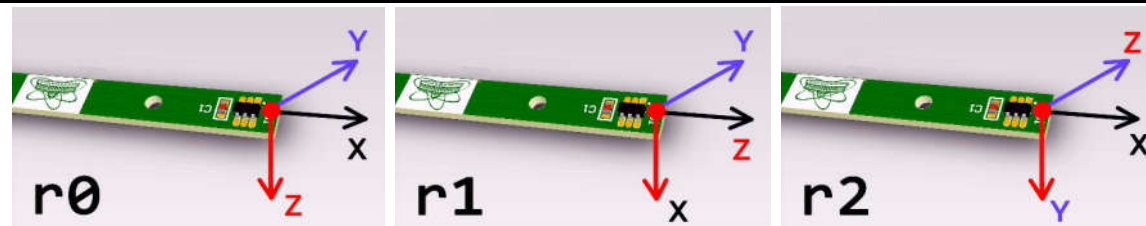


Command type	char	ASCII	Parameter	Device answer	Example of use
Set "zero"	'z'	0x7A	no	<u>success:</u> <i>ok\n\r</i> <u>error:</u> <i>cmd error\n\r</i>	Printf('z');
Get sensor info	's'	0x73	no	Device sends to host coordinate system and ID of the plugged sensor: "30" - 8.1 Gs sensor, "37" - 1.3 kGs sensor. Example: <i>r=0,id=37\n\r</i>	Printf('s');
Get sensor coefficients	'k'	0x6B	no	<u>success:</u> <i>a1000b1000c1000\n\r</i> coefficient a is equal to 1.000 on X, Y and Z. Measurements are multiplied on this coefficients, numbers after 'a' for X axis, after 'b' for Y and 'c' for Z data <u>error:</u> <i>cmd error\n\r</i>	Printf("k");
Set sensor coefficients	'a' 'b' 'c'	0x61 0x62 0x63	the coefficient: a1571 = 1.571 for X axis, etc. for Y and Z	<u>success:</u> <i>a=1000\n\r</i> <u>error:</u> <i>cmd error\n\r</i> or <i>err p1/2/3 \n\r</i>	Printf("a1000b1023c1000")
Set sensor coordinate system	'r'	0x72	'0' : r0 pic '1' : r1 pic '2' : r2 pic	<u>success:</u> <i>r=0\n\r</i> <u>error:</u> <i>cmd error\n\r</i> or <i>err p1/2/3 \n\r</i>	Printf("r0");



- * By default the device continuously sends measured data in format: **x-0626t\n y02006t\n z01018t\n**
- * For 8.1 Gs need to divide measured data to 100
- * If measurements has positive sign, the first char after axis word is '0', if negative = '-'