

Advanced sercom protocol

The *advanced sercom* is a serial protocol used to get sensors values from the robot and set its actuators states. There are two modes of communication that are ASCII mode and BINARY mode; the command value identifies the current mode, if it is a positive value then the request is ASCII, otherwise it is BINARY. They differ in that the BINARY mode uses the minimal number of bytes to exchange data, whereas the ASCII mode presents the data in a readable way. Moreover with the BINARY mode many requests can be sent to the robot at the same time, for example by issuing [- 'A'][- 'N'][0] we can get both the accelerometer and proximity values (sequence terminated by a null character). Beware that the commands are case sensitive.

The available commands in ASCII mode are listed below:

Command	Description	Return value
A	get accelerometer axes values	a,x,y,z
B,#	set body led state: 0=off, 1=on, 2=inverse	b
b	get battery state: 1=ok, 0=low (e-puck HWRev < 1.3) get battery raw and percentage (e-puck HWRev 1.3 and 2)	b,# (e-puck HWRev < 1.3) b,# (%) (e-puck HWRev 1.3 and 2)
C	get selector position	c,#
D,#,#	set motors speed (left=[-1000...1000], right=[-1000...1000])	d
E	get motors speed	e,left,right
F,#	set front led state: 0=Off, 1=on, 2=inverse	f
G	get IR receiver data	g IR check : 0x#, address : 0x#, data : 0x#
g	get gyro 3 axes values (only available with e-puck HWRev 1.3 and 2)	g,x,y,z
H	print help	
I	get camera parameters	i,mode,width,height,zoom,x,y
J,#,#,#,#,#,#	set camera parameters: mode(0=grayscale, 1=rgb565), width=[1...640], height=[1...480], zoom=[1,2,4,8], x=[1...640], y=[1...480]	j

	The total image size must not exceed 4056 bytes.	
K	calibrate proximity sensors	k
L,#,#	set led state: led number, state (0=off, 1=on, 2=inverse)	l
M	get floor sensors values (only available with floor sensors extension (http://www.gctronic.com/doc/index.php/Others_Extension#Ground_sensors) and cliff extension (http://www.gctronic.com/doc/index.php/Others_Extension#Cliff_sensor))	m,#,#,#,#, the first 3 values are from floor sensors extension (left, center, right), the last two values are from cliff extension (right, left)
N	get proximity sensors values	n,#,#,#,#,#,#,#,#, the last two values are returned only if using the gumstix extension (http://www.gctronic.com/doc/index.php/Overo_Extension) and refer to the two long range proximity sensors (http://www.gctronic.com/doc/index.php/Overo_Extension#Long_range_proximity_sensors) mounted on the extension
O	get ambient light sensors values	o,#,#,#,#,#,#,#,#, the last two values are returned only if using the gumstix extension (http://www.gctronic.com/doc/index.php/Overo_Extension) and refer to the two long range proximity sensors (http://www.gctronic.com/doc/index.php/Overo_Extension#Long_range_proximity_sensors) mounted on the extension
P,#,#	set motors steps values (left, right)	p
Q	get motors steps values	q,left,right
R	reset robot	
S	stop the robot and turn off its leds	s
T,#	play sound: 1-5 (sound id), others values (stop sound)	t
t	get temperature (only available with e-puck HWRev 1.3 and 2)	t,#
U	get microphones amplitude values	u,#,#,#
V	get protocol version	v,Version 1.2.2 August 2008 GCtronic HW version: #
W,#,#,#	write I2C message (mod, register, value)	w

Y, #, #	read I2C register (mod, regsiter)	y, value
---------	-----------------------------------	----------

The available commands in BINARY mode are listed below:

Command	Description	Return value
[-a]	get accelerometer axes values	[X_LSB][X_MSB][Y_LSB][Y_MSB][Z_LSB][Z_MSB]
[-A]	get acceleration values	[A_LSB][A_1][A_2][A_MSB][O_LSB][O_1][O_2][O_MSB][I_LSB][I_1][I_2][I_MSB] A=acceleration, O=orientation, I=inclination
[-b]	get battery state: 1=ok, 0=low (e-puck HWRev < 1.3) get battery percentage (e-puck HWRev 1.3 and 2)	[STATE][0] (e-puck HWRev < 1.3) [PERCENT_LSB][PERCENT_MSB]
[-D][LEFT_LSB][LEFT_MSB][RIGHT_LSB][RIGHT_MSB]	set motors speed (left=[-1000...1000], right=[-1000...1000])	
[-E]	get motors speed	[LEFT_LSB][LEFT_MSB][RIGHT_LSB][RIGHT_MSB]
[-g]	get gyro 3 axes values (only available with e-puck HWRev 1.3 and 2)	[X_LSB][X_MSB][Y_LSB][Y_MSB][Z_LSB][Z_MSB]
[-I]	get camera image	[mode][width][height][...cam data...] header: mode, width, height are 1 byte each. in color mode cam data are in RGB565 format.
[-L][LED_NUM][STATE]	set led state: led number (0-7=leds, 8=body, 9=front), state (0=0ff, 1=on, 2=inverse)	
[-M]	get floor sensors values (only available with floor sensors extension (http://www.gctronic.com/doc/index.php/Others_Extensions#Ground_sensors) and cliff extension (http://www.gctronic.com/doc/index.php/Others_Extensions#Cliff_sensor))	[F0_LSB][F0_MSB][F1_LSB][F1_MSB][F2_LSB][F2_MSB][C0_LSB][C0_MSB][C1_LSB][C1_MSB] the first 3 values are from floor sensors extension, the last two values are from cliff extension

[-N]	get proximity sensors values	[P0_LSB][P0_MSB][P1_LSB][P1_MSB] [P2_LSB][P2_MSB][P3_LSB][P3_MSB] [P4_LSB][P4_MSB][P5_LSB][P5_MSB] [P6_LSB][P6_MSB][P7_LSB][P7_MSB] [P8_LSB][P8_MSB][P9_LSB][P9_MSB] the last two values (P8, P9) are returned only if using the gumstix extension (http://www.gctronic.com/doc/index.php/Overo_Extension) and refer to the two long range proximity sensors (http://www.gctronic.com/doc/index.php/Overo_Extension#Long_range_proximity_sensors) mounted on the extension
[-O]	get ambient light sensors values	[P0_LSB][P0_MSB][P1_LSB][P1_MSB] [P2_LSB][P2_MSB][P3_LSB][P3_MSB] [P4_LSB][P4_MSB][P5_LSB][P5_MSB] [P6_LSB][P6_MSB][P7_LSB][P7_MSB] [P8_LSB][P8_MSB][P9_LSB][P9_MSB] the last two values are returned only if using the gumstix extension (http://www.gctronic.com/doc/index.php/Overo_Extension) and refer to the two long range proximity sensors (http://www.gctronic.com/doc/index.php/Overo_Extension#Long_range_proximity_sensors) mounted on the extension
[-P][LEFT_LSB] [LEFT_MSB][RIGHT_LSB] [RIGHT_MSB]	set motors steps values (left, right)	
[-Q]	get motors steps values	[LEFT_LSB][LEFT_MSB][RIGHT_LSB] [RIGHT_MSB]
[-t]	get temperature (only available with e-puck HWRev 1.3 and 2)	[TEMP]
[-u]	get microphones amplitude values	[MIC0_LSB][MIC0_MSB][MIC1_LSB] [MIC1_MSB][MIC2_LSB][MIC2_MSB]
[-U]	get microphones sampling buffer	[...buffer of 600 bytes...][SCAN_ID]
[-W][MOD][REG][VALUE]	write I2C message (mod, register, value)	
[-w][R0][R1][R2][R3] [R4][R5][R6][R7][B0] [B1][B2][B3][B4][B5] [B6][B7][G0][G1][G2] [G3][G4][G5][G6][G7] [R8][B8][G8][IR0][IR1] [IR2][IR3][IR4][IR5] [IR6][IR7]	set LEDs and IRs state of the RGB extension (http://www.gctronic.com/doc/index.php/RGB_Panel); Rx=Bx=Gx=[0...128]; IRx=[0=off, 1=on]	

8/19/23, 9:56 PM

Advanced sercom protocol - GCtronic wiki

[-Y][MOD][REG]	read I2C register (mod, regsiter)	[value]
----------------	-----------------------------------	---------

Retrieved from "http://www.gctronic.com/doc/index.php?title=Advanced_sercom_protocol&oldid=2339"

-
- This page was last edited on 3 May 2023, at 05:59.