16. Lista komend obsługiwanych przez kartę

Command	Device mode	Description
setMode	master/slave	[USB] setMode=<master/slave> - set master or slave mode
setLogging	master/slave	[USB] setLogging=<true or 1/false or 0> - turn logging on serial on or off
setEth	master/slave	[USB] setEth=<true or 1/false or 0> - set if should use Ethernet
setMac	master/slave	[USB] setMac=DEADBEEFFEED - set MAC address of the ethernet shield
setIp	master/slave	[USB] setIp=123.123.123.123 - set IP for no DHCP mode
setDhcp	master/slave	[USB] setDhcp=<true or 1/false or 0> - set if should use DHCP
setDomeDelay	master/slave	[USB] setDomeDelay=<delay in millis> set dome delay (for changing movement direction)
reset	master/slave	hard reset using reset pin (works only with compatible shield)
resetSoft	master/slave	soft reset
getFirmwareVersion	master/slave	get firmware version (from #define FIRMWARE_VERSION)
getDeviceName	master/slave	get device name (from #define DEVICE_RESPONSE)
getMac	master/slave	get MAC address of the Ethernet shield
emergencyStop	master/slave	in case of emergency turns off all the relays
getLocalStatus	master/slave	get local status in format: <inputs>;<encoders>;<buttons>;<relays>, e.g.: 1:0:0:0;100:0;0:0:0:0;1:0:1:0
setEncoderADebounce	master/slave	setEncoderADebounce=<debounce time in millis> set debounce time for encoder A
setEncoderBDebounce	master/slave	setEncoderBDebounce=<debounce time in millis> set debounce time for encoder B
setEncoderATimeout	master/slave	setEncoderATimeout=<timeout in millis> set timeout for encoder A (moveDome has to change encoder in that timeout)
setEncoderARange	master/slave	setEncoderARange=<range> set range for encoder A
setHomeSignalLow	master/slave	setHomeSignalLow=<true or 1/false or 0> set to 1 if home signal is low, 0 if home signal is high
setHttpPassword	master/slave	setHttpPassword=<password> set password for HTTP authentication
getRelaysNames	master/slave	get relays names in "name1:name2:name3:..." format
getInputsNames	master/slave	get inputs names in "name1:name2:name3:..." format
getButtonsNames	master/slave	get buttons names in "name1:name2:name3:..." format
getRelays	master/slave	get relays status in "1:0:0:1" format
getButtons	master/slave	get buttons status in "1:1:0:..." format
getInputs	master/slave	get inputs status in "1:1:1:0:..." format
getInputsAndEncoder	master/slave	get input and counter statuses "1:1:1:0:...:323"
switchOnFreeRelay	master/slave	switchOnFreeRelay=<relay> switch ON free relay; <relay> index from 1
switchOffFreeRelay	master/slave	switchOffFreeRelay=<relay> switch OFF free relay; <relay> index from 1
switchOnRelay	master/slave	*only for firmware or special use* switchOnRelay=<relay> - switch on relay nr <relay>
switchOffRelay	master/slave	*only for firmware or special use* switchOffRelay=<relay> - switch off relay nr <relay>
switchOnRelayTillInput	master/slave	*only for firmware or special use* switchOnRelayTillInput=<relay>:<input> - switch on relay till input is on
switchOnRelayTillTimeout	master/slave	*only for firmware or special use* switchOnRelayTillTimeout=<relay>:<timeout> - switch on relay till timeout
switchOnRelayTillInputOrTimeout	master/slave	*only for firmware or special use* switchOnRelayTillInputOrTimeout=<relay>:<input>:<timeout> - switch on relay till input is on or timeout passed
switchOnRelayTillEncoder	master/slave	*only for firmware or special use* switchOnRelayTillEncoder=<relay>:<count> - switch on relay till encoder counter counted to <count>
setEncoderA	master/slave	setEncoderA=<count> - set encoder A counter to <count>
setEncoderB	master/slave	setEncoderB=<count> - set encoder B counter to <count>
getStatus	master	get master+slave status in format: <master inputs>;<master encoders>;<master buttons>;<master relays>#<slave inputs>;<slave encoders>;<slave buttons>;<slave relays>#<bluetooth connected>#<fresh flag>
moveDome	master	moveDome=<direction>:<encoder> dome movement; <direction> is 'CW' or 'CCW', <encoder> is optional
stopDome	master	stop dome movement
setFresh	master	setFresh=<value> set "fresh" flag to value 1 or 0
findHome	master	find home in CCW direction
calibrate	master	calibrate (count encoder during rotation)
getCalibratedRotation	master	get counted rotation span
slave	master	slave=<command> send command to the slave device
moveShutter	slave	moveShutter=<direction>:<timeout> dome movement; <direction> is 'OPEN' or 'CLOSE', <timeout> in milliseconds is optional
stopShutter	slave	stop shutter movement
setSlaveOpenOnlyOnHome	slave	setSlaveOpenOnlyOnHome=<true or 1/false or 0> - set if slave should open only if it's on home