Script Command Examples

Commands that have no arguments are not listed.

Part 1 – Module Motion

conveyor_pulse_decode(type, A, B)

Example command:

```
conveyor_pulse_decode(1, 2, 3)
```

Parameters in example

type = 1 - is quadrature encoder, input A and B must be square waves with 90 degree offset. Direction of the conveyor can be determined.

A = 2 – Encoder output A is connected to digital input 2

B = 3 – Encoder output B is connected to digital input 3

force_mode(task frame, selection vector, wrench, type, limits)

Example command:

```
force_mode(p[0.1,0,0,0,0.785],[1,0,0,0,0,0],[20,0,40,0,0,0],2,[.2,.1,.1,.785,.785,1.57])
```

Parameters in example

Task frame = p[0.1,0,0,0,0.785] – This frame is offset from the base frame 100 mm in the x direction and rotated 45 degrees in the rz direction

Selection Vector = [1,0,0,0,0,0] – The robot is compliant in the x direction of the Task frame above.

Wrench = [20,0,40,0,0,0] – The robot apples 20N in the x direction. It also accounts for a 40N external force in the z direction.

Type = 2 – The force frame is not transformed.

Limits = [.1,.1,.1,.785,.785,1.57] – max x velocity is 100 mm/s, max y deviation is 100 mm, max z deviation is 100 mm, max rx deviation is 45 deg, max ry deviation is 45 deg, max rz deviation is 90 deg.

movec(pose_via, pose_to, a=1.2, v=0.25, r=0)

Example command

movec(p[x,y,z,0,0,0], pose to, a=1.2, v=0.25, r=0.05)

Parameters in example

Note: first position on circle is previous waypoint.

pose_via = p[x,y,z,0,0,0] – second position on circle.
Note rotations are not used so they can be left as zeros.
Note: This position can also be represented as joint angles [j0,j1,j2,j3,j4,j5} then forward kinematics is used to calculate the
corresponding pose

pose_to = p[x,y,z,rx,ry,rz] – third position on circle.
Note: This position can also be represented as joint angles [j0,j1,j2,j3,j4,j5} then forward kinematics is used to calculate the corresponding pose

a = 1.2 – acceleration is 1.2 m/s/s

v = 0.25 – velocity is 250 mm/s

r = 0 – blend radius (at pose_to) is 50 mm.

movej(q, a=1.4, v=1.05, t=0, r=0)

Example command

movej([0,1.57,-1.57,3.14,-1.57,1.57], a=1.4, v=1.05, t=0, r=0)

Parameters in example

q = [0,1.57,-1.57,3.14,0,1.57] – base is at 0 deg rotation, shoulder is at 90 deg rotation, elbow is at -90 deg rotation, wrist 1 is at 180 deg rotation, wrist 2 is at -90 deg rotation, wrist 3 is at 90 deg rotation. Note: joint positions (q can also be specified as a pose, then inverse kinematics is used to calculate the corresponding joint positions)

a = 1.4 – acceleration is 1.4 rad/s/s

v = 1.05 – velocity is 1.05 rad/s

t = 0 – the time (seconds) to make move is not specified. If it were specified the command would ignore the a and v values.

r = 0 – the blend radius is zero meters.

movel(pose, a=1.2, v=0.25, t=0, r=0)

Example Command

movel(pose, a=1.2, v=0.25, t=0, r=0)

Example Parameters

pose = p[0.2,0.3,0.5,0,0,3.14] – position in base frame of x = 200 mm, y = 300 mm, z = 500 mm, rx = 0, ry = 0, rz = 180 deg.

a = 1.2 – acceleration of 1.2 m/s^2

v = 0.25 – velocity of 250 mm/s

t = 0 – the time (seconds) to make the move is not specified. If it were specified the command would ignore the a and v values.

r = 0 – the blend radius is zero meters.

movep(pose, a=1.2, v=0.25, t=0, r=0)

Example Command

movel(pose, a=1.2, v=0.25, t=0, r=0)

Example Parameters

pose = p[0.2,0.3,0.5,0,0,3.14] – position in base frame of x = 200 mm, y = 300 mm, z = 500 mm, rx = 0, ry = 0, rz = 180 deg.

a = 1.2 – acceleration of 1.2 m/s^2

v = 0.25 – velocity of 250 mm/s

r = 0 – the blend radius is zero meters.

position_deviation_warning(enabled, threshold)

Example Command

Position_deviation_warning(True,0.8)

Example Parameters

Enabled = True – Logging of warning is turned on

Threshhold = 0.8 – 80% of deviation that causes a protective stop causes a warning to be logged in the log history file.

reset_revolution_counter(qNear=[0.0, 0.0, 0.0, 0.0, 0.0, 0.0])

Example Command

```
reset_revolution_counter(qNear=[0.0, 0.0, 0.0, 0.0, 0.0, 0.0])
```

Example Parameters

qNear = [0.0, 0.0, 0.0, 0.0, 0.0, 0.0] – Optional parameter, resets the revolution counter of wrist 3 to zero on UR3 robots to the nearest zero location to joint rotations represented by qNear.

servoc(pose, a=1.2, v=0.25, r=0)

Example Command

```
Servoc(p[0.2,0.3,0.5,0,0,3.14],a=1.2,v=0.25,r=0)
```

pose = p[0.2,0.3,0.5,0,0,3.14] – position in base frame of x = 200 mm, y = 300 mm, z = 500 mm, rx = 0, ry = 0, rz = 180 deg.

a = 1.2 – acceleration of 1.2 m/s^2

v = 0.25 – velocity of 250 mm/s

r = 0 – the blend radius at the target position is zero meters.

servoj(q, a, v, t=0.008, lookahead_time=0.1, gain=300)

Example Commands

```
servoj([0.0,1.57,-1.57,0,0,3.14], t=0.1, lookahead_time=0.1, gain=300)
```

or

```
servoj([0.0,1.57,-1.57,0,0,3.14], 0, 0, 0.1, 0.1, 300)
```

q = [0.0,1.57,-1.57,0,0,3.14] – joint angles in radians representing rotations of base=0°, shoulder=90°, elbow=-90°, wrist1=0°, wrist2=0°, wrist3= 180°

a = 0 not used in current version

v = 0 not used in current version

t=.1 time where the command is controlling the robot. The function is blocking for time t [S]

lookahead time=.1	time [S], range [0.03,0.2] smoothens the trajectory with this lookahead time
gain=300	proportional gain for following target position, range [100,2000]

set conveyor tick count(tick count, absolute encoder resolution=0)

Example Command

Set_conveyor_tick_count(24543,0)

Tick count = 24543 which is a value read from a MODBUS register being updated by the absolute encoder

Absolute encoder resolution = 0. 0 is a 32 bit signed encoder, range [-2147483648 ; 2147483647] (default)

Set_pos(q)

Example Command

set_pos([0.0,1.57,-1.57,0,0,3.14])

q = [0.0,1.57,-1.57,0,0,3.14] –the position of the simulated robot with joint angles in radians representing rotations of base=0°, shoulder=90°, elbow=-90°, wrist1=0°, wrist2=0°, wrist3= 180°

speedj(qd, a, t)

This is the equivalent of a joint jog command. Robot moves as directed for time t.

Example Command

speedj([0.2,0.3,0.1,0.05,0,0], 0.5, 0.5)

qd -	Joint speeds of – base = 0.2 rad/s, shoulder = 0.3 rad/s, elbow=0.1 rad/s, wrist1=0.05 rad/s, wrist2 and wrist 3 = 0 rad/s
a	acceleration of 0.5 rad/s^2 of the leading axis (shoulder is this case)
t	time of 0.5 s – time before the function returns

speedl(xd, a, t, aRot=’a’)

This is the equivalent of a linear jog command. Robot moves as directed for time t.

Example Command

speedl([0.5,0.4,0.0,0.,1.57,0,0], 0.5, 0.5)

qd - Tool speeds of – x = 500 mm/s, y = 400 mm/s, rx=90 deg/s, ry and rz= 0 mm/s

a acceleration of 0.5 m/s^2 of the leading axis (shoulder is this case)

t time of 0.5 s – time before the function returns

stop conveyor tracking(a=15, aRot='a')

Example Command

stop conveyor tracking(a=15)

a tool acceleration of 15 m/sˆ2

stopj(a)

Example Command

Stopj(2)

a = 2 rad/s^2 rate of deceleration of the leading axis.

stopl(a, aRot='a')

Example Command

Stopj(20)

a = 20 m/s^2 - rate of deceleration of the tool

aRot: tool deceleration [rad/sˆ2] (optional), if not defined a, position acceleration, is used. i.e. it supersedes the "a" deceleration.

track conveyor circular(center, ticks per revolution, rotate tool)

Example Command

track conveyor circular(p[0.5,0.5,0,0,0,0],500.0, false)

center = p[0.5,0.5,0,0,0,0] – location of the center of the conveyor

ticks per revolution = 500.0 – the number of ticks the encoder sees when the conveyor moves one revolution.

rotate tool = false - the tool should not rotate with the conveyor, but stay in the orientation specified by the trajectory (movel() etc.).

track conveyor linear(direction, ticks per meter)

Example Command

track conveyor linear(p[1,0,0,0,0,0],1000.0)

direction = p[1,0,0,0,0,0] - Pose vector that determines the direction of the conveyor in the base coordinate system of the robot

ticks per meter = 1000. - How many ticks the encoder sees when the conveyor moves one meter.

Part 2 – Module Internals

get_inverse_kin(x, qnear, maxPositionError=1e-10, maxOrientationError=1e-10)

Example Command

get_inverse_kin(p[.1,.2,.2,0,3.14,0], [0.,3.14,1.57,.785,0,0])

x pose with position of x=100mm, y=200mm, z=200mm and rotation vector of rx=0 deg., ry=360 deg, rz=0 deg.

qnear solution should be near to joint angles of j0=0 deg, j1=360 deg, j2=180 deg, j3=90 deg, j4=0 deg, j5=90 deg.

maxPositionError is 1e-10 m

maxOrientationError is 1e-10 rad

is_within_safety_limits(pose)

Example Command

is_within_safety_limits(p[.1,.2,.2,0,3.14,0])

pose target pose with position of x=100mm, y=200mm, z=200mm and rotation vector of rx=0 deg., ry=360 deg, rz=0 deg.

popup(s, title='Popup', warning=False, error=False, blocking=False)

Example Command

popup("here I am", title="Popup #1",blocking=True)

s popup text is "here I am"

title popup title is "Popup #1"

blocking is True so the popup must be cleared before other actions will be performed.

set_gravity(d)

Example Command

```
set_gravity(0,9.82,0)
```

d is vector with a direction of y (direction of the robot cable) and a magnitude of 9.82 m/s^2 (1g).

set_payload(m,CoG)

Example Command

```
set_payload(3., [0,0,.3])
```

m mass is set to 3 kg payload

CoG Center of Gravity is set to x=0 mm, y=0 mm, z=300mm from the center of the tool mount in tool coordinates

set_payload_cog(CoG)

Example Command

```
set_payload([0,0,.3])
```

CoG is set to x=0 mm, y=0 mm, z=300mm from the center of the tool mount in tool coordinates

set_payload_mass(m)

Example Command

```
set_payload(3.)
```

m mass is set to 3 kg payload

set_tcp(pose)

Example Command

```
set_tcp(p[0.,.2,.3,0.,3.14,0.])
```

pose tool center point is set to x= 0mm, y= 200mm, z=300mm, rotation vector is rx=0 deg, ry=180 deg, rz=0 deg. In tool coordinates

sleep(t)

Example Command

```
sleep(3.)
```

textmsg(s1, s2=’ ’)

Example Command

```
textmsg(“value = “, 3)
```

s1 set first part of message to “value = “
s2 set second part of message to 3

message in the log is “value = 3”

Part 3 – Module urmath

acos(f)

Example Command

acos(0.707)

f is the cos of 45 deg. (.785 rad)

Returns .785

asin(f)

Example Command

asin(0.707)

f is the sin of 45 deg. (.785 rad)

Returns .785

atan(f)

Example Command

atan(1.)

f is the tan of 45 deg. (.785 rad)

Returns .785

atan2(x,y)

Example Command

atan2(.5,.5)

x is the one side of the triangle
y is the second side of a triangle

Returns atan(.5/.5) = .785

binary_list_to_integer(l)

Example Command

binary_list_to_integer([True,False,False,True]) l	represents the binary values 1001

Returns 9

ceil(f)

Example Command

ceil(1.43)

Returns 2

cos(f)

Example Command

cos(1.57)

f	is angle 1.57 rad (90 deg)

Returns 0.0

d2r(d)

Example Command

d2r(90)

d	angle in degrees

returns 1.57 angle in radians

floor(f)

Example Command

floor(1.53)

returns 1

get_list_length(v)

Example Command

get_list_length([1,3,3,6,2])

v	is the list 1,3,3,6,2

returns 5

integer_to_binary_list(i)

Example Command

integer_to_binary_list(57)

I integer 57

Returns binary list

interpolate_pose(p_from,p_to, alpha)

Example Command

interpolate_pose(p[.2,.2,.4,0,0,0],p[.2,.2,.6,0,0,0],.5)

p_from	p[.2,.2,.3,0,0,0]
p_to	p[.2,.2,.5,0,0,0]
alpha	.5 is 50% of the way in between p_from and p_to

Returns p[.2,.2,.4,0,0,0]

length(v)

Example Command

length(“here I am”)

v equals string “here I am”

Returns 9

log(b,f)

Example Command

log(10.,4.)

b	base 10
f	log of 4

Returns 0.60206

norm(a)

Examples of Command

norm(-5.3) Returns 5.3

norm(-8) Returns 8

norm(p[-.2,.2,-.2,-1.57,0,3.14]) Returns 3.52768

point_dist(p_from,p_to)

Example Command

point_dist(p[.2,.5,.1,1.57,0,3.14], p[.2,.5,.6,0,1.57,3.14])

p_from   is first point p[.2,.5,.1,1.57,0,3.14]
p_to   is second point p[.2,.5,.6,0,1.57,3.14]

Returns distance between the points regardless of rotation or 500 mm

pose_add(p_1,p_2)

Example Command

pose_add(p[.2,.5,.1,1.57,0,0], p[.2,.5,.6,1.57,0,0])

p_1   is first point p[.2,.5,.1,1.57,0,0]
p_2   is second point p[.2,.5,.6,1.57,0,0]

Returns   p[0.4,1.0,0.7,3.14,0,0]

pose_dist(p_from,p_to)

Example Command

pose_dist(p[.2,.5,.1,1.57,0,3.14], p[.2,.5,.6,0,1.57,3.14])

p_from   is first point p[.2,.5,.1,1.57,0,3.14]
p_to   is second point p[.2,.5,.6,0,1.57,3.14]

Returns distance between the points regardless of rotation or 500 mm

pose_inv(p_from)

Example Command

pose_inv(p[.2,.5,.1,1.57,0,3.14])

p_from   is point p[.2,.5,.1,1.57,0,3.14]

Returns   p[0.19324,0.41794,-0.29662,1.23993,0.0,2.47985]

pose_sub(p_1,p_2)

Example Command

pose_sub(p[.2,.5,.1,1.57,0,0], p[.2,.5,.6,1.57,0,0])

p_1 is first point p[.2,.5,.1,1.57,0,0]
p_2 is second point p[.2,.5,.6,1.57,0,0]

Returns p[0.0,0.0,-0.5,0.0,.0.,0.0]

pose_trans(p_1,p_2)

Example Command

pose_trans(p[.2,.5,.1,1.57,0,0], p[.2,.5,.6,1.57,0,0])

p_1 is first point p[.2,.5,.1,1.57,0,0]
p_2 is second point p[.2,.5,.6,1.57,0,0]

Returns p[0.4,-0.0996,0.60048,3.14,0.0,0.0]

pow(base,exponent)

Example Command

pow(5.,3)

base 5

exponent 3

Returns 125.

r2d(r)

Example Command

r2d(1.57)

r 1.5707 rad

Returns90 deg

rotvec2rpy(rotation_vector)

Example Command

rotvec2rpy([3.14,1.57,0])

rotation_vector	[3.14,1.57,0]	rx=3.14, ry=1.57, rz=0
Returns	[2.80856, .16202, 0.9]	roll=2.80856, pitch =.16202, yaw=0.9

rpy2rotvec(rpy_vector)

Example Command

rpy2rotvec([3.14,1.57,0])

rpy_vector	[3.14,1.57,0]	roll=3.14, pitch=1.57, yaw=0
Returns	[2.22153, 0.00177, -2.21976]	rx=2.22153, ry =0.00177, rz=-2.21976

sin(f)

Example Command

sin(1.57)

f angle of 1.57 radians (90 deg)

Returns 1.0

sqrt(f)

Example Command

sqrt(9)

f 9

Returns 3

tan(f)

Example Command

tan(.7854)

f angle of 0.7854 radians (45 deg)

Returns 1.0

Part 4 – Module interfaces

get_analog_in(n) **(obsolete but operational function)**

Example Command

get_analog_in(1)

n analog input 1

Returns value of analog input #1

get_analog_out(n) **(obsolete but operational function)**

Example Command

get_analog_out(1)

n analog output 1

Returns value of analog output #1

get_configurable_digital_in(n)

Example Command

get_configurable_digital_in(1)

n configurable digital input 1

Returns True or False

get_configurable_digital_out(n)

Example Command

get_configurable_digital_out(1)

n configurable digital output 1

Returns True or False

get_digital_in(n)

Example Command

get_digital_in(1)

n digital input 1

Returns True or False

get_digital_out(n)

Example Command

get_digital_out(1)

n digital output 1

Returns True or False

get_euromap_input(port_number)

Example Command

get_euromap_input(1)

port_number euromap digital input on port 1

Returns True or False

get_euromap_output(n)

Example Command

get_euromap_output(1)

port_number euromap digital output on port 1

Returns True or False

get_configurable_digital_in(n)

Example Command

get_configurable_digital_in(1)

n configurable digital input 1

Returns True or False

get_flag(n)

Example Command

get_flag(1)

n value of flag 1

Returns True or False

get_standard_analog_in(n)

Example Command

get_standard_analog_in(1)

n standard analog input 1

Returns value of standard analog input #1

get_standard_analog_out(n)

Example Command

get_standard_analog_out(1)

n standard analog output 1

Returns value of standard analog output #1

get_standard_digital_in(n)

Example Command

get_standard_digital_in(1)

n standard digital input 1

Returns True or False

get_standard_digital_out(n)

Example Command

get_standard_digital_out(1)

n standard digital output 1

Returns True or False

get_tool_analog_in(n)

Example Command

get_tool_analog_in(1)

n tool analog input 1

Returns value of tool analog input #1

get_tool_digital_in(n)

Example Command

get_tool_digital_in(1)

n tool digital input 1

Returns True or False

get_tool_digital_out(n)

Example Command

get_tool_digital_out(1)

n tool digital output 1

Returns True or False

modbus_add_signal(IP, slave number, signal address, signal type, signal name))

Example Command

modbus_add_signal(“172.140.17.11”, 255,5,1,”output1”)

IP	IP address 172.140.17.11
Slave number	255
Signal address	5
Signal type	1 digital output
Signal name	output1

modbus_delete_signal(signal name))

Example Command

modbus_delete_signal(”output1”)

Signal name output1

modbus_get_signal_status(signal name,is_secondary_program)

Example Command

```
modbus_get_signal_status(”output1”,False)
```

Signal name	output1
Is_secondary_program	False (NOTE: must be set to False)

modbus_send_custom_command(IP, slave_number, function_code, data)

Example Command

```
modbus send custom command("172.140.17.11",103,6,[17,32,2,88])
```

IP	IP address 172.140.17.11
Slave number	103
Function code	6
Data	[17,32,2,88]

Function code and data are specified by the manufacturer of the slave Modbus device connected to the UR controller

modbus_set_output_register(signal name, register_value, is_secondary_program)

Example Command

```
modbus_set_output_register(”output1”, 300, False)
```

Signal name	output1
Register value	300
Is_secondary_program	False (NOTE: must be set to False)

modbus_set_output_signal(signal name, digital_value, is_secondary_program)

Example Command

```
modbus_set_output_signal(”output1”, True, False)
```

Signal name	output1
Digital value	True
Is_secondary_program	False (NOTE: must be set to False)

modbus_set_runstate_dependnet_choice(signal name, runstate_choice)

Example Command

```
modbus_set_runstate_dependent_choice(”output2”, 1)
```

Signal name	output2	
Runstate dependent choice	1	set low when a program is not running

modbus_set_signal_update_frequency(signal name, update_frequency)

Example Command

modbus_set_signal_update_frequency(”output2”, 20)

Signal name	output2
Signal update frequency	20 Hz

read_input_boolean_register(address)

Example Command

read input boolean register(3)

address	input boolean register 3

read_input_float_register(address)

Example Command

read input float register(3)

address	input float register 3

read_input_integer_register(address)

Example Command

read input integer register(3)

address	output integer register 3

read_output_boolean_register(address)

Example Command

read output boolean register(3)

address	output boolean register 3

read_output_float_register(address)

Example Command

read output float register(3)

address	output float register 3

read_output_integer_register(address)

Example Command

read output integer register(3)

address	output integer register 3

read_port_bit(address)

Example Command

read port bit(3)

address	port bit 3

read_port_register(address)

Example Command

read port register(3)

address	port register 3

rpc_factory(type,url)

Example Command

rpc factory("xmlrpc", "http://127.0.0.1:8080/RPC2")

type	xmlrpc
url	http://127.0.0.1:8080/RPC2

rtde_set_watchdog(variable_name, min_frequency, action=’pause)

Example Command

rtde set watchdog(**"input int register 0"**, 10, **"stop"**)

variable name	input int register 0
min frequency	10 Hz
action	stop the program

set_analog_inputrange(port,range) (obsolete but operational function)

Example Command

```
set_analog_inputrange(1,0)
```

port	analog input port 1 (on controller)
range	0 0-5V

set_analog_out(n,f) (obsolete but operational function)

Example Command

```
set_analog_out(1,2)
```

n	analog output port 1 (on controller)
f	2 volts

set_analog_output_domain(port,domain)

Example Command

```
set_analog_output_domain(1,1)
```

port	analog output port 1 (on controller)
domain	1 (0-10 volts)

set_configurable_digital_out(n,b)

Example Command

```
set_configurable_digital_out(1,True)
```

n	configurable digital output 1
b	True

set_digital_out(n,b)

Example Command

```
set_digital_out(1,True)
```

n	digital output 1
b	True

set_euromap_output(port_number, signal_value)

Example Command

```
set_euromap_output(1,True)
```

port_number euromap digital output on port 1

signal_value True

set_euromap_runstate_dependent_choice(port_number, runstate_choice)

Example Command

```
set_euromap_runstate_dependent_choice(1,1)
```

port_number euromap digital output on port 1

runstate choice 0 = set low when a program is not running

set_flag(n,b)

Example Command

```
set_flag(1,True)
```

n value of flag 1

set_runstate_configurable_digital_output_to_value(outputid, state)

Example Command

```
Set_runstate_configurable_digital_output_to_value(5, 2)
```

outputid configurable digital output on port 5

runstate choice 2 = High when program is not running

set_runstate_gp_boolean_output_to_value(outputid, state)

Example Command

```
set_runstate_gp_boolean_output_to_value(5, 2)
```

outputid configurable digital output on port 5

runstate choice 2 = High when program is not running

set_runstate_standard_analog_output_to_value(outputid, state)

Example Command

```
set_runstate_standard_analog_output_to_value(1, 2)
```

outputid configurable digital output on port 1

runstate choice 2 = Max when program is not running

set_runstate_standard_digital_output_to_value(outputid, state)

Example Command

```
Set_runstate_standard_digital_output_to_value(5, 2)
```

outputid	standard digital output on port 5
runstate choice	2 = High when program is not running

set_runstate_tool_digital_output_to_value(outputid, state)

Example Command

```
Set_runstate_tool_digital_output_to_value(1, 2)
```

outputid	tool digital output on port 1
runstate choice	2 = High when program is not running

set_standard_analog_input_domain(port, domain)

Example Command

```
set_standard_analog_input_domain(1,0)
```

port	analog input port 1
domain	1 (0-10 volts)

set_standard_analog_out(n,f)

Example Command

```
set_standard_analog_out(1,4)
```

n	standard analog output 1
f	4 volts (or mA depending on domain setting)

set_standard_digital_out(n,f)

Example Command

```
set_standard_digital_out(1,True)
```

n	standard digital output 1
f	True

set_tool_analog_input_domain(port,domain)

Example Command

```
set_tool_analog_input_domain(1,1)
```

port	tool analog input 1
domain	1 = 0-10 V

set_tool_digital_out(n,b)

Example Command

```
set_tool_digital_out(1,True)
```

n	tool digital output 1
b	True

set_tool_voltage(voltage)

Example Command

```
set_tool_voltage(24)
```

voltage	24 volts

socket_close(socket_name=’socket_0’)

Example Command

```
socket_close(socket_name=”socket_0”)
```

socket_name	socket_0

socket_get_var(name, socket_name=’socket_0’)

Example Command

```
socket_get_var(“POS.X”, socket_name=”socket_0”)
```

socket_name	socket_0

socket_open(address, port, socket_name=’socket_0’)

Example Command

```
socket_open(“192.168.5.1”, 50000, ”socket_10”)
```

address	192.168.5.1
socket	50000
socket_name	socket_10

socket_read_ascii_float(number, socket_name=’socket_0’)

Example Command

```
socket_read_ascii_float(4,”socket10”)
```

Number	4	Number of floats to read
socket_name	socket_10	

socket_read_binary_integer(number, socket_name=’socket_0’)

Example Command

```
socket_read_ascii_float(4,”socket10”)
```

Number	4	Number of integers to read
socket_name	socket_10	

socket_read_byte_list(number, socket_name=’socket_0’)

Example Command

```
socket_read_ascii_float(4,”socket10”)
```

Number	4	Number of byte variables to read
socket_name	socket_10	

socket_read_line(socket_name=’socket_0’)

Example Command

```
socket_read_line(”socket10”)
```

socket_name	socket_10

socket_read_string(socket_name=’socket_0’,prefix=’ ‘, suffix=’ ’)

Example Command

```
socket_read_string(”socket10”,prefix=”>”,suffix=”<”)
```

socket_name	socket_10

socket_send_byte(value,socket_name=’socket_0’)

Example Command

```
socket_send_byte(2,”socket10”)
```

value	2
socket_name	socket_10

Returns True or False (sent or not sent)

socket_send_int(value,socket_name=’socket_0’)

Example Command

socket_send_int(2,”socket10”)

value	2
socket_name	socket_10

Returns True or False (sent or not sent)

socket_send_line(str,socket_name=’socket_0’)

Example Command

socket_send_int(“hello”,”socket10”)

str	hello
socket_name	socket_10

Returns True or False (sent or not sent)

socket_send_string(str,socket_name=’socket_0’)

Example Command

socket_send_int(“hello”,”socket10”)

str	hello
socket_name	socket_10

Returns True or False (sent or not sent)

socket_set_var_name(name,value,socket_name=’socket_0’)

Example Command

socket_set_var_name(”POS_Y”,2200,”socket10”)

name	POS_Y
value	2
socket_name	socket_10

write_output_boolean_register(address, value)

Example Command

write_output_boolean_register(3,True)

address	3
value	True

write_output_float_register(address, value)

Example Command

write_output_float_register(3,37.68)

address	3
value	37.68

write_output_integer_register(address, value)

Example Command

write_output_integer_register(3,12)

address	3
value	12

write_port_bit(address, value)

Example Command

write_port_bit(3,True)

address	3
value	True

write_port_register(address, value)

Example Command

write_port_bit(3,100)

address	3
value	100