UR Forum-Help-Q&A

HOME ~ GENERAL FAQ ~ SOLUTIONS ~ PROGRAMMING ~ MISCELLANEOUS ~ TRAINING ~ PRODUCTS ABOUT CONTACT ZACOBRIA WEBSHOP

UR Script: Send Commands from host (PC) to robot via socket connection.

Z

Visit Zacobria Webshop

UR Script: Send Commands from host (PC) to robot via socket connection.

The example "UR Script: Script programming from the teaching pendant."on CB2 forum used the "Script" function available in the teaching pendant to type Script code or import entire files with script code program into the robot and run from the teaching pendant. UR Script: Script programming from the teaching pendant.

Note: Variable will not work when sending raw script commands because variables (and other programming technique like loop – If-Else etc) needs to be controlled either by the external host program (e.g. a Python program) – or within a Polyscope program. Such variable cannot be left in between a host program and a Polyscope program – there is no controller in between.

To send variables – make loops – use If-Else and to get informations back from the robot then the client-server method can be considered.

UR Script: Client-Server example.

Application Description:

This example focus on making a script that entirely control the robot remotely from a host computer by Script programming.

The format is the same as in previous chapter and as documented in the Script manual, but the teaching pendant is not used for programming in this example. On the robot a server on TCP port 30002 is running to receive script commands and thereby control the robot.

Python® is used as the scripting language for making these program for testing the script programming with the robot. The Script file can also still be edited in a Notepad.

With this method the data flow is sort of "one way" i.e. it is possible to send commands to the robot, but it is not meant to read data back i.e. inputs or other read data cannot be send back to the host with this method. To send back data see the "Client-Server" example. **Function description:**

1. Script program for Socket connection to test the connection:

The first program example is just to test the connection and illustrate how to send a script file through via the Ethernet to the TCP port 30002 on the robot.

In this case the robot has already been configured to IP address 192.168.0.9 and the host must be in the same subnet e.g. in this example defined as 255.255.255.0

Programm description:

The first Program 1 just turns digital output number 2 ON and the robot does not move at all and the program looks like this.

Echo client program import socket HOST = "192.168.0.9" # The remote host PORT = 30002 # The same port as used by the server

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST, PORT))

s.send ("set_digital_out(2,True)" + "\n")
data = s.recv(1024)

s.close()

print ("Received", repr(data))

```
# Echo client program
import socket
HOST = "192.168.0.9" # The remote host
PORT = 30002 # The same port as used by the server
```

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST, PORT))

s.send ("set_digital_out(2,True)" + "\n")

data = s.recv(1024)

s.close()

When the program is run the robot sends back status data which is normal.

The program is run from the host computer and will only have one passage because there is no loop in the program.

The library "socket" in imported into the program so the Socket connection is available. The "Host" in this case is the Robot and "PORT" is the open port on the Robot that listens for Script code.

Then the Script code necessary to connect and send on the Socket. The "s.send" command send that actual script code for the robot to execute which in the case is ("set_digital_out(2,True)" Which is like previous chapter. The + "\n") at the end of the line is a linefeed because the Robot need a linefeed after each command. The Socket connection needs to be closed again with the s.close() command. The print ("Received", repr(data)) command prints the output from the robot on the Host or PC monitor – in this case status data.

The second Program 2 is similar and just turn digital out number 2 OFF again and the robot does not move at all.



2. Script program for Socket connection that makes the robot move:

This program send in Script mode via socket port runs through 6 Waypoints.

The first 3 positions are defined by joint position data.

Echo client program

>>>

The next 3 positions are defined by a pose. Notice the "p" in front of the data wich are represented as X, Y, Z, Rx, Ry, Rz.

import socket import time HOST = "192.168.0.9" **#** The remote host PORT = 30002 # The same port as used by the server print "Starting Program" count = 0while (count < 1): s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) s.connect((HOST, PORT)) time.sleep(0.5) print "Set output 1 and 2 high" s.send ("set digital out(1, True)" + "\n") time.sleep(0.1) s.send ("set digital out(2, True)" + "\n") time.sleep(2) print "Robot starts Moving to 3 positions based on joint positions" s.send ("movej([-1.95, -1.58, 1.16, -1.15, -1.55, 1.18], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej([-1.95, -1.66, 1.71, -1.62, -1.56, 1.19], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej([-1.96, -1.53, 2.08, -2.12, -1.56, 1.19], a=1.0, v=0.1)" + "\n") time.sleep(10) print "Robot starts Moving to 3 positions based on pose positions"

s.send ("movej(p[0.00, 0.3, 0.4, 2.22, -2.22, 0.00], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej(p[0.00, 0.3, 0.3, 2.22, -2.22, 0.00], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej(p[0.00, 0.3, 0.2, 2.22, -2.22, 0.00], a=1.0, v=0.1)" + "\n") time.sleep(10) print "Set output 1 and 2 low" s.send ("set digital out(1,False)" + "\n") time.sleep(0.1) s.send ("set digital out(2,False)" + "\n") time.sleep(0.1) count = count + 1print "The count is:", count print "Program finish" time.sleep(1) data = s.recv(1024)s.close() print ("Received", repr(data)) print "Status data received from robot" # Echo client program import socket import time HOST = "192.168.0.9" # The remote host PORT = 30002 # The same port as used by the server print "Starting Program" count = 0while (count < 1): s = socket.socket(socket.AF_INET, socket.SOCK_STREAM) s.connect((HOST, PORT)) time.sleep(0.5) print "Set output 1 and 2 high" s.send ("set_digital_out(1,True)" + "\n") time.sleep(0.1) s.send ("set_digital_out(2,True)" + "\n") time.sleep(2) print "Robot starts Moving to 3 positions based on joint positions" s.send ("movej([-1.95, -1.58, 1.16, -1.15, -1.55, 1.18], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej([-1.95, -1.66, 1.71, -1.62, -1.56, 1.19], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej([-1.96, -1.53, 2.08, -2.12, -1.56, 1.19], a=1.0, v=0.1)" + "\n") time.sleep(10) print "Robot starts Moving to 3 positions based on pose positions" s.send ("movej(p[0.00, 0.3, 0.4, 2.22, -2.22, 0.00], a=1.0, v=0.1)" + "\n") time.sleep(10) s.send ("movej(p[0.00, 0.3, 0.3, 2.22, -2.22, 0.00], a=1.0, v=0.1)" + "\n") time.sleep(10)

print	"Set	output	1	and	2	low"
1						

s.send ("set_digital_out(1,False)" + "\n")

time.sleep(0.1)

s.send ("set_digital_out(2,False)" + "\n")
time.sleep(0.1)

count = count + 1
print "The count is:", count

print "Program finish"

time.sleep(1)
data = s.recv(1024)

s.close()
print ("Received", repr(data))

print "Status data received from robot"

Result feedback from robot when the program is executed

Contribute moderator
Option 1 \$50.00 SGD

Disclaimer: While the Zacobria Pte. Ltd. believes that information and guidance provided is correct, parties must rely upon their skill and judgement when making use of them. Zacobria Pte. Ltd. assumes no liability for loss or damage caused by error or omission, whether such an error or omission is the result of negligence or any other cause. Where reference is made to legislation it is not to be considered as legal advice. Any and all such liability is disclaimed.

If you need specific advice (for example, medical, legal, financial or risk management), please seek a professional who is licensed or knowledgeable in that area.

Author: By Zacobria Lars Skovsgaard

Accredited 2015-2018 Universal Robots support Centre and Forum.

Tweet

Follow @zacobria

General Disclaimer .