

Progwhiz.com

Robotics Server Manager API

Rev 1.2

Table of Contents

Introduction	2
API Definition	3
API Message Descriptions	4
Start Server	5
Stop Server	6
Open COM Port	7
Close COM Port	8
Change COM Port	9
Change TCP Port	10
Change Password	11
Change User Limit	12
Change Script #1	13
Change Script #2	13
Change Script #3	13
Change Script #4	14
Change Script #5	14
Change Script #6	14
Functional Flow	15
Winsock Client	15

Introduction

The following document outlines the Manager command packets compatible with the Progwhiz Robotics Server Module. This API will enable software developers to create applications to manage the behaviour of the Server application by change operating parameters:

- ✓ TCP/COM Ports
- ✓ Passwords
- ✓ Server Start/Stop

www.progwhiz.com

API Definition

The API is a uni-directional protocol. This protocol supports sending Manager commands to the server, the classes are defined into the following groups:

API Sending

- Start Server
- Stop Server
- Open COM Port
- Close COM Port
- Change COM Port
- Change TCP Port
- Change Password
- Change User Limit

API Message Descriptions

The following descriptions will show the format of each command and the meaning and value sets of each parameter. Please note the messages are passed as textual strings. The rules are as follows:

- Brackets [] denotes a single Byte passed as a single character
- No spaces allowed
- Single digit Values shown once not in Brackets [] are each a single byte
- (for e.g. 20 is really two(2) bytes which are 50 and 48 and the 20 is a textual representation)

www.progywhiz.com

Start Server

Message Format Layout

#Paramaters	Data1	Data2	Data2
1	Startup	NA	NA

Example

Command Message: **Startup**

www.progwhiz.com

Stop Server

Message Format Layout

#Paramaters	Data1	Data2	Data2
1	Shutdown	NA	NA

Example

Command Message: **Shutdown**

www.progywhiz.com

Open COM Port

Message Format Layout

#Paramaters	Data1	Data2	Data2
1	Open ComPort	NA	NA

Example

Command Message: **Open ComPort**

www.progywhiz.com

Close COM Port

Message Format Layout

#Paramaters	Data1	Data2	Data2
1	Close ComPort	NA	NA

Example

Command Message: **Close ComPort**

www.progywhiz.com

Change COM Port

Message Format Layout

#Paramaters	Data1	Data2
2	New ComPort	ComPort<1 to 255>

Example

Command Message: **New ComPort21**

New Com Port is **21**

www.progywhiz.com

Change TCP Port

Message Format Layout

#Paramaters	Data1	Data2
2	New Port	Port<1 to 65535>

Example

Command Message: **New Port2122**
New TCP is **2122**

www.progywhiz.com

Change Password

Message Format Layout

#Paramaters	Data1	Data2
2	New Password	AlphaNumeric

Example

Command Message: **New Password**the**pass**
New Password is **the**pass

www.progywhiz.com

Change User Limit

Message Format Layout

#Paramaters	Data1	Data2
2	New Limit	1 to 499

Example

Command Message: **New Limit35**

New User Limit is **35**

www.progywhiz.com

Change Script #1

Message Format Layout

#Paramaters	Data1	Data2
2	New Script1	Scriptname.ext

Example

Command Message: **New Script1Scriptname.ext**
New Script1 name is **Scriptname.ext**

Change Script #2

Message Format Layout

#Paramaters	Data1	Data2
2	New Script2	Scriptname.ext

Example

Command Message: **New Script2Scriptname.ext**
New Script2 name is **Scriptname.ext**

Change Script #3

Message Format Layout

#Paramaters	Data1	Data2
2	New Script3	Scriptname.ext

Example

Command Message: **New Script3Scriptname.ext**
New Script3 name is **Scriptname.ext**

Change Script #4

Message Format Layout

#Paramaters	Data1	Data2
2	New Script4	Scriptname.ext

Example

Command Message: **New Script4Scriptname.ext**
New Script4 name is **Scriptname.ext**

Change Script #5

Message Format Layout

#Paramaters	Data1	Data2
2	New Script5	Scriptname.ext

Example

Command Message: **New Script5Scriptname.ext**
New Script5 name is **Scriptname.ext**

Change Script #6

Message Format Layout

#Paramaters	Data1	Data2
2	New Script6	Scriptname.ext

Example

Command Message: **New Script6Scriptname.ext**
New Script6 name is **Scriptname.ext**

Functional Flow

There are two (2) methods to establish connections with the API:

➤ **Winsock Client / Server**

Winsock Client

- 1) Set **ActiveX** Server Property **TCP_Manager_Secure = False**
- 2) Connect to Server (IP, Port and Password configured properly)
- 3) Send ***password***
- 4) Receive ***'Validated'*** if valid password or will Close Connection