VERSABUILT

The Robot2CNC Protocol - v1.0.1

Last Updated: 31 March 2020

Overview

The Robot2CNC Protocol (v1.0) is designed to be a simple API level abstraction of CNC Primitives. While the terminology varies between cnc manufacturers, their functionality is more or less the same. This protocol provides a standard API to control a cnc, regardless of it's make. This should enable a robot automation expert to control various cnc's with little effort.

Versioning

This document is versioned in accordance to the Semantic Versioning 2.0.0 standard. Each update must be reflected in the version at the title of this file. Libraries targeting this protocol must label the exact version they are implement.

Network Channel

- Protocol: TCP/IP
- Port: 9002 (Default)
- Timeout: None
- Encoding: UTF-8

The Robot2CNC will listen for new connections on port 9002 by default on startup, after a TCP/IP close on an active connection or after a CLOSE command is completed. By default, there is no read timeout, a timeout value can and should be set by the client to prevent a rebooted client from holding on to the connection. The socket is bi-directional. In general, the Robot2CNC receives commands from the client and sends responses as events occur. Commands are queued up if the previous command has not completed. Queued commands and responses are guaranteed to be sent in the order received.

Robot2CNC Protocol Format

Commands and responses over the Robot2CNC sockets are semicolon terminated text strings. The parameters of commands and responses are delineated by commas. The first parameter is the action and the remaining parameters define the action. Capital letters are used in string parameters except where noted. Numeric parameters can be integer, decimal or hexadecimal. Hexadecimal values are only supported and preceded by '0x'. Whitespace, CR and LF are ignored in the protocol. All commands

return a response. If a command returns a response, the response must start with the exact command it is responding to.

Example Command

- Command: `VERSION;`
- Response: `VERSION,1.0.0;`

Error Messages

Messages from the Robot2CNC to the client. These can represent various errors and events the Robot2CNC experiences. When a command is sent by the client that the Robot2CNC cannot complete for some reason, an `ERROR` response is sent back to the client.

ERROR

- Response: `ERROR,{CODE},{ERROR_MESSAGE},{COMMAND},{COMMAND_PARAMETERS};`

Example Responses

```
ERROR,101,Invalid command,SCURRY;

ERROR,202,CNC Communication Error,CNC_STATUS;

ERROR,303,CNC Communication Error,SELECT_PROGRAM,81004;
```

General Commands

Version

- Command: `VERSION;`
- Response: `VERSION,{Protocol SemVer number};`

Get the version of this protocol the Robot2CNC is running.

Example Response

```
VERSION,1.0.0;
```

CNC Status

- Command: `CNC_STATUS;`
- Response: `CNC_STATUS,{STATUS};`

Status (depending on CNC make): {IDLE / RUNNING / COMPLETE / ALARM}

Trigger an status message from the Robot2CNC. Allows the client to get the value of the status on demand.

Back to top

Select Program

- Command: `SELECT_PROGRAM,{PROGRAM};`
- Response: `SELECT_PROGRAM,{PROGRAM};`

Select the given program on the CNC.

Back to top

Run Program

- Command: `RUN_PROGRAM,{PROGRAM};`
- Response: `RUN_PROGRAM,{PROGRAM};`

Select the given program on the CNC and then Cycle Start the CNC.

Back to top

Cycle Start

- Command: `CYCLE_START;`
- Response: `CYCLE_START;`

Cycle Start the CNC.

Back to top

Read Macro Variable

- Command: `READ_MACRO,{VARIABLE_ADDR};`
- Response: `READ_MACRO,{VARIABLE_ADDR},{VALUE};`

Read macro variable value from a given address.

Back to top

Write Macro Variable

- Command: `WRITE_MACRO,{VARIABLE_ADDR},{VALUE};`
- Response: `WRITE_MACRO,{VARIABLE_ADDR},{VALUE};`

Write macro variable value base on a given address.

Back to top

Ping

- Command: `PING,{HOST},{PORT};`
- Response: `PING,{HOST},{PORT},{RESPONSE};`

Determine the connection status of a host. Useful for testing connectivity of the CNC.

Response can be:

- `Reachable` Host and port are reachable, a connection could be established
- `Refused` Host was reachable but refused our connection. Check that the port is open
- `Unreachable` Host and port could not be reached. Check that the host is online and has correct IP address.
- `Unknown` An error occurred on the Robot2CNC unit. Connection could not be determined as a result.

Back to top

Get Digital Input Value

- Command: `GET_DI,{IO_Address};`

- Response: `GET_DI,{IO_Address},{VALUE};`

Get the value of a given digital input. `VALUE` will be `"ON"` or `"OFF"`.

Back to top

Set Digital Output Value

- Command: `SET_DO,{IO_Address},{VALUE};`
- Response: `SET_DO,{IO_Address},{VALUE};`

Set the value of a given digital output to the passed value. `VALUE` can be `"ON"` or `"OFF"`.

Back to top

Set Configuration Value

- Command: `SET_CONF,{FIELD},{TYPE},{VALUE};`
- Response: `SET_CONF,{FIELD},{TYPE},{VALUE};`

Set the value of a field (or subfield) in a configuration file. Subfields can be accessed with standard js dot notation (i.e. field.subfield).

Type can be:

- `STRING`
- `NUMBER`
- `BOOLEAN` (Can be `TRUE` or `FALSE` or `true` or `false`).

Examples:

- `SET_CONF,drivers.haasNGCCNC.link.host,STRING,192.168.1.1;`
- `SET_CONF,drivers.haasNGCCNC.dispatcherProgram,NUMBER,9000;`

Back to top

Get Configuration Value

- Command: `GET_CONF,{FIELD};`
- Response: `GET_CONF,{FIELD},{VALUE};`

Get the value of a field (or subfield) in a configuration file. Subfields can be accessed with standard js dot

Get the value of a field (or subfield) in a configuration file. Subfields can be accessed with standard js dot notation (i.e. field.subfield).

Back to top

Toggle Remote Support

- Command: `TOGGLE_REMOTE_SUPPORT,{VALUE};`
- Response: `TOGGLE_REMOTE_SUPPORT,{VALUE};`

Enable or disable remote support access. `VALUE` can be `"ON"` or `"OFF"` .

Restart Robot2CNC Service

- Command: `RESTART;`
- Response: `RESTART;`

Restart the Robot2CNC service. Please allow 30 seconds for the service to boot.

Restore Factory Default Settings

- Command: `RESTORE_TO_FACTORY_DEFAULTS;`
- Response: `RESTORE_TO_FACTORY_DEFAULTS;`

Restore Robot2CNC to factory default settings. This will restart the Robot2CNC service. Please allow 30 seconds for the service to boot.

Reset CNC

- Command: `RESET_CNC;`
- Response: `RESET_CNC;`

Reset the CNC. This will change a "COMPLETE" status to "IDLE".

Enable Haas Cell Safe

- Command: `ENABLE_HAAS_CELL_SAFE;`
- Response: `ENABLE_HAAS_CELL_SAFE;`

Enable the haas cell safe signal.

Disable Haas Cell Safe

Disable Haas Cell Safe

- Command: `DISABLE_HAAS_CELL_SAFE;`
- Response: `DISABLE_HAAS_CELL_SAFE;`

Disable the haas cell safe signal.

Get Edition

- Command: `GET_EDITION;`
- Response: `GET_EDITION,<EDITION>;`

Returns the Robot2CNC unit edition. Edition can be "R2C Basic" or "R2C Mill Panel".

List Digital Inputs

- Command: `LIST_DIGITAL_INPUTS;`
- Response: `LIST_DIGITAL_INPUTS,<...INPUTS>;`

Returns a list of available digital inputs.

List Digital Outputs

- Command: `LIST_DIGITAL_OUTPUTS;`
- Response: `LIST_DIGITAL_OUTPUTS,<...OUTPUTS>;`

Returns a list of available digital outputs.

Set IP

- Command: `SET_IP,<INTERFACE>,<IP>;`
- Response: `SET_IP,<INTERFACE>,<IP>;`

Set the IP address of an interface on the device.

Example: `SET_IP,eth0,192.168.1.1;`

12000 W Franklin Rd, Boise Idaho