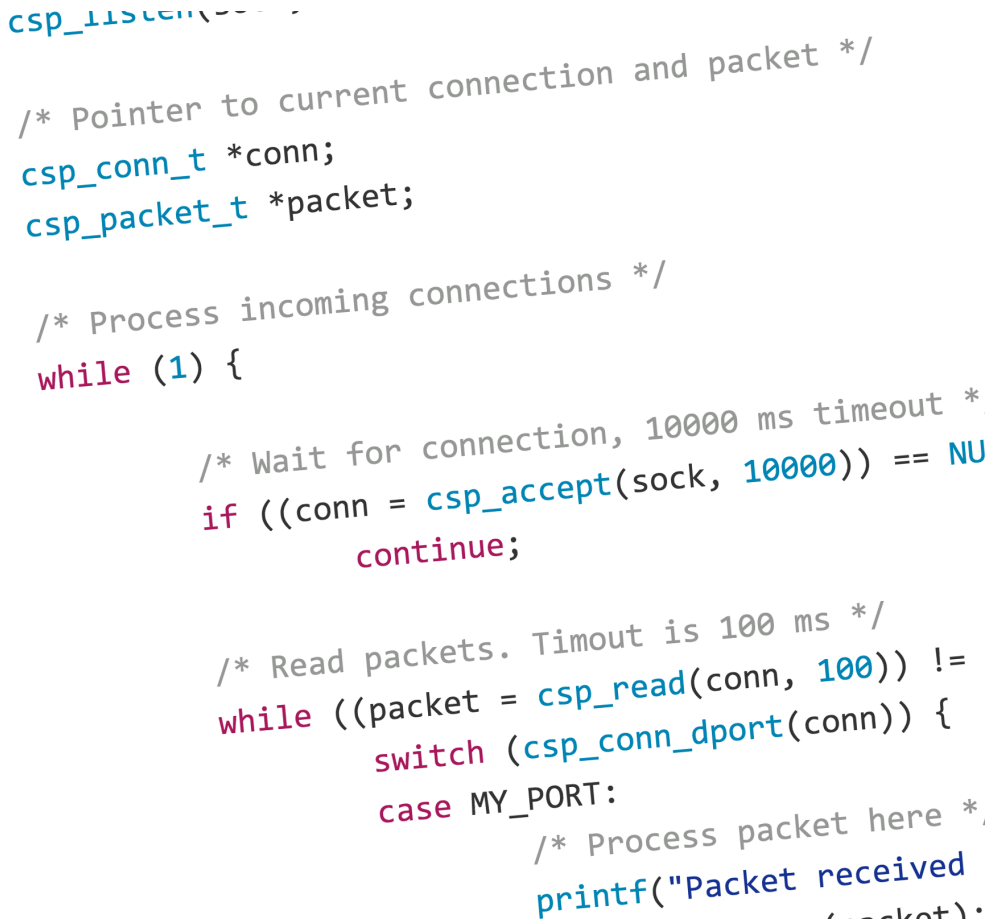
GOMSPACE



GomSpace

CSP Client

Manual

Software Documentation

Release v1.3

GomSpace ApS • Alfred Nobels Vej 21C • DK-9220 Aalborg East • Denmark
www.gomspace.com • info@gomspace.com • +45 96356111

Table of Contents

1. Introduction

This manual describes the CSP Client software package.

The CSP Client includes the client interfaces for most GomSpace standard product. It can be used for interfacing to e.g. NanoCom AX100 or NanoCam C1U. It can also be used as an example of how to develop GomSpace compatible applications for Linux.

The basis of the CSP Client is the individual modules and their functionalities. The following modules are included:

- **libcsp** Network library
- **libgosh** Shell interface
- **liblog** Logging systems
- **libutil** Various utilities
- **libftp** File transfer (client interface)
- **libparam** Parameter system (client interface)
- **clients** Client interface for individual products

Installation and build instructions are described in the next sections.

1.1 CSP Client Installation and Build

1.1.1 Source Code Installation

Install Source Code:

1. Create a workspace folder for the source code, e.g. ~/workspace: *mkdir ~/workspace*
2. Change working directory to the workspace folder: *cd ~/workspace*
3. Copy source code tar ball (*csp-client-<version>.tar.bz2*) to ~/workspace
4. Unpack source code: *tar xfv csp-client-<version>.tar.bz2*

You now have the source code installed in *~/workspace/csp-client-<version>/*

1.1.2 Building the Source Code

Prerequisites: Recent Linux with python (example build with ubuntu 14.10)

Install requirements:

```
$ sudo apt get install build-essential libzmq3-dev
```

The executable build script is called *waf* and is placed in the root-folder of the software repository. Waf uses a recursive scheme to read the local *wscript* file(s), and submerge into the *lib/lib. . . ' folders and find their respective 'wscript* files. This enables an elaborate configuration of all modules included in the CSP Client.

To configure and build the source code:

1. Change directory to source code folder: *cd ~/workspace/csp-client-<version>*

2. Configure source code: *waf configure*. This will configure the source code and check that the tool chain (compiler, linker, ...) is available
3. Build source code: *waf build* (or just *waf*).

The *waf build* command will compile the source code, link object files and generate an executable in the build folder with the name *csp-client*.

1.1.3 Client Modules

To include all client modules from the clients sub-folder:

```
$ waf configure
```

To include selected client modules (e.g. nanopower and nanocom-ax) from the clients sub-folder:

```
$ waf configure --with-clients=nanopower,nanocom-ax
```

Remember to re-build source after a *waf configure*:

```
$ waf build
```

Command line arguments

The CSP Client main application csp-client has the following command line arguments:

```
-a Address
-c CAN device
-d USART device
-b USART buad
-z ZMQHUB server
```

Example 1: Starting csp-client with address 10 and connecting to a ZMQ proxy on localhost:

```
$ ./build/csp-client -a 10 -z localhost
```

Example 2: Starting csp-client with address 10 using usart to /dev/ttyUSB0 at baudrate 500000:

```
$ ./build/csp-client -a 10 -d /dev/ttyUSB0 -b 500000
```

Example 3: Starting csp-client with address 10 using CAN interface can0:

```
$ ./build/csp-client -a 10 -c can0
```