

SDR TCP/IP interface Version 0.1 (not yet updated for the version 1)

Main goal

The SDR TCP/IP interface is able to interface SdR receivers, as the direct or EXTIO interfaces. It is simple and can be used for an exchange between:

- * a client program (Multipsk, for example),
- * a SdR (local or remote) able to transmit IQ 16 bits samples, at a sampling rate of 48000 Hz, through a specific server.

For developers, it is proposed programs permitting a complete simulation of a server and a client (see below).

This version of Multipsk managed a basic application, i.e. the mandatory part of the version 0.1 of the protocol:

- * the default sampling frequency (48000 Hz),
- * the default hardware type managed: 3 (16 bits, IQ),
- * the mandatory procedures (Pversion, HWMPcapa, InitHWMP, Set_HWLO, Received, Bug_on_P).

The hardware type 4 (with only a frequency control, data coming from a specific sound card, as for the Funcube) is also managed by Multipsk.

It will be possible to manage either a SDR server located in the same PC or a remote SDR server. In other words, it would be possible to decode transmissions from your receiver located in another place, if authorized in your country.

Typical working for the user

In the standard option, the SDR server file is "External" to the Multipsk directory. But it can be internal, i.e. in the same directory as Multipsk. As an example, it is proposed, for a test, a simulation of server SDR_TCP_IP_TEST.exe. This pseudo-server generates and sends an IQ signal at 6000 Hz (square wave).

After having started Multipsk, on the Combo box select "**Test**" (for the "test server"), then push on the pink "**SdR TCP/IP**" button of the "**I/Q interfaces for SdR transceivers**" panel. Push on the "**Connection**" button. Once the connection done, the number after "**N=**" (number of data) increases. Close this TCP/IP window and push on "**RX/TX screen**", as usual. The 6000 Hz signal will be visible in the SdR waterfall.

Note: the SDR TCP/IP connection can be automatically done pushing on the "**Auto**" button (on the Configuration screen).

Extract from the manual:

.....

- The selection of the SDR TCP/IP server (called "MP" in the protocol) is done with a **Combo box**:

- The server is "**External**" to the Multipsk directory by default. In that case, it will be necessary to start up the server before Multipsk.
- A "**Remote**" server is located in another PC. Indeed, Multipsk can be a TCP/IP client and exchange data or commands with a SDR TCP/IP server local (on the same PC) or distant (on a distant PC).
The standard is a local server, internal or external, in "local loop" using the standard IP address 127.0.0.1 and located on the same PC. A remote server via Internet can be connected with an IP or a DNS address.
- Otherwise, the server is internal to the Multipsk directory with a name of the form "SDR_TCP_IP_xyz.EXE" (XYZ is associated to the SDR). The only available local server is "**Test**" (for "SDR_TCP_IP_TEST.EXE"), to be used only for information. In that case (internal server), the SDR TCP/IP connection can be automatically done pushing on the "**Auto**" button.

Moreover, the "**+Speaker**" may be pushed, to listen to the demodulated sound.

The "**Transceiver**" window on the "RX/TX" screen permits to control the frequency of the SDR.

.....
Click on the  button to reach help.

For the SdR manufacturer

For the ones having a SdR receiver (except RTL SDR dongles, directly managed by Multipsk), if you want to have your SdR directly interfaced to Multipsk (and, perhaps, to other programs in the future), you have to forward this message to your SdR manufacturer. This one could find interest to develop the necessary server interface, for the satisfaction of his/her clients.

Files in this ZIP

Here are presented the different files necessary to implement the protocol SDR TCP/IP INTERFACE Version 0.1.

- The « **SDR_TCP_IP_INTERFACE_V_0_1.pdf** » file gives the description of the protocol. Note that the version 0 is considered as obsolete.
- The « **Client_in_C++** » and « **Server_in_C++** » files are very simple programs only able to establish a TCP/IP communication. These programs are generated in C++ (C++Builder 6 Borland). They are supposed built in the C:\SDR_TCP_IP_INTERFACE directory.

- The « **Client_in_Delphi** » and « **Server_in_Delphi** » files are very simple programs only able to establish a TCP/IP communication. These programs are generated in Delphi 6 (Borland).
- The « **Complete simulation of a SDR TCP-IP client in Delphi** » file can be used as a model for a true SDR TCP/IP client program (called « CP » in the protocol), as for example a decoding program. It is issued from Multipsk.
- The « **Complete simulation of a SDR TCP-IP server in Delphi** » file can be used as a model for a true SDR TCP/IP server program (called « MP » in the protocol). It is interfaced with the SDR hardware and the client program. It has the general equivalent role of the EXTIO file. These two last programs are generated in Delphi 6 (Borland).

Questions of the SDR manufacturer can be asked either to the author (about the protocol) or, in English, to the Multipsk Io group (multipsk@groups.io) (about needs of SDR users or feedback).