

protocol_editor.py

A spreadsheet-style grid editor for `protocols-N.txt` files, built to replace editing them in LibreOffice Calc or Excel.

Why it exists

A protocol file has a ragged structure: many 9-field frequency records followed by exactly one 11-field control record. A spreadsheet has no idea about that. Calc pads short rows, reformats numbers, and applies no per-field validation — so a file can come back out looking fine and still be unreadable to the device.

This editor shows the same grid, but it knows the real record structure, enforces the device's own field limits cell by cell, and keeps the `Rec #` numbering consecutive automatically.

Field limits

The bounds are taken from `rotaryLimits()` in `rotary_Decompile.ino`, so the editor enforces the same limits as the Generator itself:

Field	Range
Freq 1–4	0.04 – 65535.00 Hz (0.0373 Hz is the MCPWM floor; 16-bit timer maximum at the top)
Duty 1–4	0.00 – 1.00
Mode	1 – 4 (1 Simple, 2 Protocol, 3 Sweep, 4 Adjust)
Freq Time	0 – 9999 seconds
Prog Time	0 – 9999 minutes
Sweep Inc	0.00 – 9999.99
File #	0 – 99
Group ID fields	whole number, 1 or higher

Maximum 275 records per file.

A cell that fails validation turns **pink** immediately as you type. Group ID fields depend on how many records exist, so their upper bound is checked as a cross-record pass when you save rather than per keystroke.

Using it

Start empty:

```
python3 protocol_editor.py
```

Open a file directly:

```
python3 protocol_editor.py --file protocols-3.txt
```

Must be run from the folder containing `protocol_tool.py`, which it imports for file reading and writing.

The window

Toolbar: New · Open... · Save · Save As... · Insert Row Above · Insert Row Below · Delete Row

Frequency grid — the scrolling upper section, one row per frequency record, with a fixed header that stays put. Click a row to select it; Insert and Delete act on the selection, and Rec # rennumbers itself afterward.

Control record — the single 11-field row below the grid, shown with its own labelled headers. The first field is always the literal 0 and is not editable.

Keyboard: Enter and the arrow keys move between cells, so you can work down a column without reaching for the mouse.

Saving

Save runs the full validation pass across every cell plus the cross-record group ID check. If anything is out of range, the save is refused and the offending cells are marked, so a bad file never reaches the device.

Files are written in the exact canonical format `protocol_tool.py` expects, which means a file saved here uploads without any further correction.

Requirements

Needs `tkinter`. On Debian/Mint it may be a separate package:

```
sudo apt install python3-tk
```

Typical workflow

1. `protocol_tool.py` download — pull the file off the Generator
2. `protocol_editor.py --file protocols-N.txt` — edit it
3. `protocol_tool.py` upload — send it back

Related

- `protocol_tool.py` — moves the files this editor edits
- `protocol_tool_gui.py` — click-driven alternative for the transfer step