



## ANEJO III. REVISIONES DE GRBL CONTROLLER

Con el paso del tiempo el programa está adquiriendo diferentes funcionalidades. A continuación se muestra un informe, en inglés, con todas las modificaciones que ha ido sufriendo el programa desde la versión 3.0 (la primera compatible con GRBL). Esto es crucial para entender desde donde se partió, hasta donde se ha llegado, y por qué es tan importante tanto el programa como el apoyo que está recibiendo.

El texto se ha extraído directamente desde GitHub, publicado de forma pública por el autor en la web Zapmaker.org. El texto se presenta sin traducir, ya que se emplea un lenguaje muy sencillo que hasta resulta agradable de leer, y supone un ahorro de tiempo muy importante por parte de la redacción.

### V3.6.1

Executable release rollup.

IMPORTANT: Manual/Jog control defaults to new mode which doesn't update the position.

This was done to ensure that jog motions are as fast as possible. However, if you want to have the original 3.5.1 and earlier behavior, go to Tools | Options | Display and change radio button to "Always Request but without Idle Check". Sending files still updates position, it is only manual/jog that does not.

### V3.6

#### Enhancements

- Options dialog is now tabbed to better fit widgets.
- Ability to handle 4th axis as either A, B, or C.
- LETARTARE: Added UVW axes.
- #45: Added ability to completely or partially turn off querying position from grbl to help improve responsiveness. The new default is to disable waiting for idle before a jog or manual command, which speeds up those operations. Additionally, you can completely turn off position reporting during jog or manual control which will result in light-greyed LCD numbers. A button to query the position has been added, which will provide up-to-date position at the time the button is pushed. Look in Options | Display tab for new button options. You can also set the rate at which the query takes place from 0.5-10 seconds, if it is enabled. Setting a time that is too long may result in the system waiting too long to retrieve the final position before control is handed back. Also, the current position dot on the Visualizer turns from red to green when the position is not accurately known to match the color of the path.

#### Bug fix

- Fixed fourth axis adjust buttons and coordinate display greyed out when they shouldn't be (under specific operating circumstances).



### V3.5.1

#### **Enhancements**

- Made subtle modifications to the command preload when sending a file so that the command queue on grbl is as full as possible. Previously we were depleting the queue and then filling it again. This has possible negative performance implications, although testing shows them to be negligible, probably because we immediately send a command after the last ok is received.
- The status list view now supports only a maximum of 200 items when active. This was done for the slower processors like the raspberry pi to prevent excessive CPU usage when many commands lines exist in a file. When the user grabs the scrollbar, all items are restored back into the view.
- A new progress bar has been added showing the current items in the grbl queue during a file send. If the number of items for some reason drops to 0 the label text will flash red. For this 0 count to occur is a very rare event.
- Added the ability to set the delay between sending of characters to grbl. The theory is that grbl can drop characters if the sending computer is very fast. There is still some debate on whether this is happening, so I provided the ability to set this value in the options from 0 to 20 ms. Note: The default of 0 ms delay is back, like in 3.4.6 and unlike in 3.5 which has a hardcoded 10 ms delay.

#### **Bug fix**

- Modified the status window redraw to always ensure items show on the bottom when the redraw timer expires.

### V3.5

#### **Bug fix**

- For very fast computers, they send characters with virtually no delay between which requires grbl to process the serial port as fast as it can. The problem is that grbl has an interrupt service routine that can slow down the processing it does of incoming characters on its serial port. This causes lost characters in its input buffer, which results in hung gcode file processing or errors. To fix, I have added a slight delay (10ms) after sending every character. This has been confirmed to solve the lost character problem on grbl.

### V3.4.6

#### **Bug fix**

- Scrolling of content now completes each time. Previously there were conditions where auto-scrolling would fail and you would have to manually scroll to see some lines.
- Fixes to support 4-axis Mega2560 (tested with actual Mega2560 running 0.81 c-axis). Includes fix to command filter to support A/B/C axis commands.
- Fixed minor issue where connecting with wrong baud rate can cause long delay before you can again control the Open/Close button.
- Incorporated more translations from LETARTARE.



### V3.4.5

#### **Bug fix**

- When modifying the ui files for auto widget rescaling, broke a number of programmatically greyed out widgets, mainly, you couldn't reopen the port after closing it. Now those are fixed.

#### **Enhancement**

- Incorporated LETARTARE's localization file for French so that it is possible to see locale-specific text in the installed version of Grbl Controller (see trlocale subdir).

### V3.4.4

#### **Bug fix**

- Incorporated LETARTARE's 4-axis (C-axis) implementation (Issues #29 and #34) (added option to enable/disable 4-axis mode) - potential issue that needs investigating: Can this be enabled on the fly or if it requires restart of Grbl Controller?
- Incorporated LETARTARE's translation support (French).
- Incorporated LETARTARE's fix for command history (#26).
- Incorporated Kosme's fix to issue #17 (#33). Z axis slider now moves in 0.1 inch increments when in inches mode.
- Incorporated Kosme's fix to issue #27. If work units are inches, now displays inches, not mm.

#### **Enhancements**

- Ability to resize entire screen and widgets. Scales down to close to 600x400 pixels for tiny displays; also remembers last size and position of screen. (#37).
- Incorporated Kosme's fix to issue #10 (#33). Popup appears to remind you to zero position if starting a new job.

### V3.4.3

#### **Bug fix**

- Unsupported G43/G44 commands include following Z command which causes problems when left standalone. The entire line must be removed.
- Now treat M2 and M30 as unsupported as they cause grbl lockup when using aggressive preloading.
- Moved program version define into its own header file so fewer files need recompiling with a version change.



### V3.4.2

#### **Bug fix**

- If a gcode command line is an arc with X,Y,Z and z-rate limiting enabled, fixed to properly split the z rate to not use the G2 command and use G1 instead as grbl was generating an error with the G2.

### V3.4.1

#### **Code sync**

- Fixes to code now allows code to compile under Qt5 without error and also resolves some localization issues. Continues to properly compile under Qt4 also. No normally noticeable functional differences.

### V3.4

#### **Enhancements**

- Z-axis slider now allows rapid control of Z-axis positioning. This feature is still being evaluated for proper behavior due to some user-interface issues. If you run into trouble, please report a bug.
- All axis control buttons have been permanently enabled regardless of whether or not an axis command is in progress - this allows more rapid updating. Note - there is no error checking for the case of entering too many commands for the Grbl buffer. This is unlikely but possible.
- Now supports setting of baud rates from 9600 to 115200.
- Aggressive preload enabled is now the default. If the user upgraded from an older Grbl Controller and had it disabled, a popup will warn them that it is being enabled. They still have the option to disable, if desired.
- Logging is turned on by default (only for new installs).
- All detected errors are collected and displayed after a file is run in the status view.
- If filtering unsupported commands, all commands filtered out are listed after the file is run.
- Logging now gives the line number of each command sent from the file and the corresponding ok or error also contains the line number.
- An option has been provided to reduce the precision of lines that exceed Grbl's line buffer, which by default is 50 characters. The code removes the arguments with the greatest precision first and ending with at least one decimal place of precision. Errors are generated if it cannot remove sufficient precision for the available buffer space. The buffer size on Grbl is configurable in the options, as older Grbls have 50 characters and newer have 70.

#### **Bug Fixes**

- If Grbl stops responding (waiting for an ok), Soft Reset Grbl now functions correctly (so you can soft reset without having to restart Grbl Controller). This affects aggressive preload mode only.



- Commands to start/stop coolant coupled with dwell commands and aggressive preload cause Grbl's modal logic to stop sending OK responses (on purpose). This was solved by waiting for the OK of the coolant off before sending additional commands. It is not clear whether this fixes all possible such modal problems, so use carefully and report any anomalies.

#### V3.3.9

##### **Enhancement**

- We now are able to remember and restore ports that are not visible to QextSerialEnumerator so that user doesn't have to retype the ghost port every time they start Grbl Controller.

#### V3.3.8

##### **Enhancement/Fix**

- Modified code that detects presence of Grbl upon serial port opening, the original assumption was that we would get the version string upon serial port open, but some Arduino-compatible boards like the AlaMode did not behave this way. Fixed the code so that if there is no response upon serial port open, we send a soft reset (Ctrl-X) and wait for the version string. Tested to work with AlaMode and with traditional Arduino Uno.

#### V3.3.7

##### **Enhancement**

- Made COM port field editable

#### V3.3.6

##### **Bug Fixes**

- Fixed decimal on the LCD display to always show three decimal points so that it is easier to read the display (it doesn't jump around anymore). Tested to hold up to 9999.999 and -9999.999, which should handle 20 x 20 meter Shapeokos.

#### V3.3.5

##### **Bug Fixes**

- Changed how we detect Grbl upon opening on COM port. This change will probably break older V0.51 capability. Previously we sent out a linefeed to trigger a response, however, this caused problems on some Arduino variants and would not result in correct sync with the controller. Fixed by no longer sending the linefeed and instead just watch the port for Grbl response.
- Changed the default "short" wait timeout to 2 seconds from 1 second in case of slow responses.



#### V3.3.4

##### **Enhancements**

- Added option checkbox to choose to filter gcode or send it as-is to the controller. The filtering removes gcode that is not supported in v0.8 of Grbl, which reduces the error reports from Grbl. Some files generate unexpected codes like G91.1 which can actually affect the motion in unexpected ways- filtering helps prevent this from happening.

##### **Bug Fixes**

- Mach3 files have no whitespace in the gcode. Fixed code to handle such files. Affects the Visualizer only.
- The filtering logic was fixed to properly filter gcode when sending a file – previous versions would filter out gcode that Grbl supported.
- Changed validator for double (i.e. float) values so that code compiles under older Qt versions.
- Changed Visualizer display to say Width-X and Height-Y instead of Width and Depth to make it less confusing.

#### V3.3.3

##### **Bug Fix**

- Modal G arc (where G2 or G3 is specified once followed by X/Y/I/J) was failing for certain arc combinations in the visualizer due to resetting the cw to false with each line. Fixed to not reset the cw variable and retain from previous lines.

#### V3.3.2

##### **Bug Fix**

- G-command can now be set once and multiple X/Y commands then issued on lines following (Allows "ShapeOko\_Calibration\_Pattern\_01b.ngc to be rendered).

#### V3.3.1

##### **Bug Fixes**

- Hardened the visualizer's file parser so that it supports spaces between positional arguments and values.
- No longer skips last line in file if that line contains a valid command.
- No longer crashes if line contains one or more stray whitespace characters.
- Also accepts semi-colon as a comment line (in addition to existing open paren and percent).



### V3.3

#### **Enhancements**

- Added "Visualizer" feature. It is now possible to see top-down view of the file to be milled.
  - Shows actual file units (inches or millimeters) independent of coordinate display.
  - Auto-scaling.
  - Supports limited subset of commands (G0/G1/G2/G3).
  - Visualizer on tab panel shared with Axis control.
  - Actual last spindle cutter position is shown; auto-scales to include both cutter and file.
  - When file is being run, overlay of all commands sent to Grbl is shown in green.
  - Displays dimensions of design, along with design's width and depth in file's units.

### V3.2

#### **Enhancements**

- Support for "Aggressive Preload" of commands when sending file to Grbl.
  - File's lines are sent to Grbl as fast as Grbl can take it, allowing Grbl to optimize acceleration. Some files have been observed to take 1/3 the time to complete!
  - Modeled after Grbl's own python script (computes/predicts how much space is available in Grbl's input buffer).
  - Due to the additional processing, CPU usage is greater than normal mode and it has been optimized to run fine on a Raspberry Pi.
  - Normal (non-preload) mode is the default
  - Response processing code now waits until a line has been received rather than collect a long stream to behave like the python script.
  - LCD updates have been set to be roughly once every second independent of rate of line sends/preloads.
- Z-rate limit feature now includes default xy rate setting.
  - Z rate limiting only occurs during file sends. If z is included with x&y then they are split into two commands automatically and the xy rate setting is used for the xy portion.
  - Since feed rate is modal, code will fix any subsequent xy items that are missing feed rate specifiers (cambam correctly adds them, but others may not).
- Replaced basic logging with Log4Qt logging capabilities.



- Support for error, warning, info, debug message types. All messages are timestamped.
  - More diagnostic messages added, mainly around sending and receiving from com ports.
  - Only error and warning are displayed in release build, all are displayed in debug.
  - If the user enables file logging, all messages are stored regardless of build type.
  - File is output to user's home folder.
  - If you enable or disable logging to file, you must restart Grbl Controller.
  - You can split out diagnostic from status messages by grepping for (DIAG) or (STATUS).
- Changed menus so that the first menu is File > Exit. Menus now use accelerators.

### Bug Fixes

- Fixed problems with z-rate limit feature, previous version was not implemented correctly.
- Fixed subtle issues with sending bytes to COM port.
  - On really fast Windows PCs, the default timeout values would occasionally result in a line not being sent. This was fixed by changing the write timeout from 10 to 500 ms by hardcoding it into the QextSerialPort library. This was only observed in aggressive preload mode, never in normal mode.

### Known issues

- New response line detect mode requires more processing because mainly of the faster stream of status messages. To get around this, certain features have been disabled in the status view so that CPU usage is manageable on the Raspberry Pi, specifically the horizontal scroll bar is not available. Turn on file logging if you wish to see the entire line.
- When using aggressive preload, any error messages returned by Grbl are not synced to send messages (they come back randomly). This can be confusing.
- If running preload, it can take longer for steppers to stop after a Stop command due to the number of items in Grbl's queue. The only way to stop sooner is to press Close / Reset button.
- Under extreme CPU loads (unlikely to occur with recent optimizations), the "runtime" display does not display accurate times, oddly, it displays the same time that would occur if not under load.

### V3.1.2 (unofficial release, optimizations for Raspberry Pi)

#### Enhancements

- Optimized multiline write to status window to use faster list-based approach.
- Removed unnecessary main window repaint after every line added to status window.
- Fixed options window text positioning so that it appears proper on the pi.





### V3.1.1 (unofficial release, adjustments for Raspberry Pi)

#### **Enhancements**

- Changed measurement units in LCD from "inches" to "in" (cosmetic).
- Shortened text and adjusted main window layout for the text size on the pi.

### V3.1

#### **Enhancements**

- Supports Grbl 0.8c. Still supports 0.8a. 0.51 is supported but without LCDs.
- Grbl settings dialog is now a table of dynamic length depending on # of params.
- LCD position data is now obtained from Grbl and is \*not\* computed.
- Added LCD displays for both machine and work coordinates.
- Cycling of Grbl via COM port toggling has been completely removed, except that Close button has now been renamed Close / Reset.
- Position data on Grbl is persisted as long as possible to allow for repositioning, etc.
- Zero work coordinates has been added (G92).
- Go-to-home feature added (lifts tool 5 mm and traverses to x=0, y=0).
- Option to put manual command (back) into absolute mode after jog button command.
- Grbl Soft reset (Ctrl-X) button added.
- Z-axis rate limiting capability added. Only affects file, not jog or manual commands.
- User is required to specify which measurement mode they are using, mm or inches:
  - Default is millimeters.
  - Grbl is configured to operate in either mm or inches.
  - If a file is sent in inches and user is in mm, it will auto-restore to mm after the file has been sent (and vice-versa, mm->inches).
- Removed annoying "No movement expected for command" message.
- Flags that cross threads have been changed to use QAtomicInt.
- Version is now shown in title bar.
- A basic log file feature has been added.
- Runs on the Raspberry Pi!



## **Known Issues**

- The registry key product name has been changed from GCodeSender to GrblController.
- Switching between millimeters and inches works, but has not been fully tested.
- Go to home has not been fully tested with inches mode.
- Manually issuing \$13 or G20/G21 can confuse the program

## V3.0

### **Enhancements**

- Nearly a full code rewrite to ensure ease-of-use and robustness.
- Percentage complete progress bar for files; also shows time elapsed sending file.
- Reset button now cycles COM port which causes Arduino to stop steppers and reinitialize position to 0,0,0 (No more manual Arduino resets!).
- Settings configurations are now stored using QSettings class for platform independence.
- Only available COM ports are now shown. Uses third-party QextSerialPort library.
- App remembers last COM port used, last folder, last file extension used after restart.
- Ability to configure wait-for-response time.
- Status list window now shows both sent and received data.
- Error responses are ignored during file send (nearly 100% of the time they are not important).
- Z-jog speed using buttons is configurable.
- Radio buttons to select mode have been removed in favor of enabling/disabling widgets based on current context. This simplifies usage.
- Popups have been removed in favor of writing errors to status list window.

### **Bug fixes**

- Overall code hardening.
- Properly implementing Qt signals and slots across the board.