

ProtoCycler Command Center Overview

Contents

- Introduction 2
- Basic Use 2
 - Installation 2
 - Overview 3
 - Connecting 4
 - Readouts 5
 - Controlling..... 5
 - Modifying Automatic Profiles 6
 - Updating ProtoCycler+ Firmware 7
- Advanced Use 8
 - Advanced Manual Control 8
 - Data Logging and Scripting 8
- Appendices..... 10
 - Appendix 1 – Readouts Description..... 10
 - Appendix 2 – Profile Settings Description..... 12
 - Appendix 3 – Manual Command Description 13

Introduction

Welcome to the ProtoCycler Command Center software, or PCC! This software allows you to monitor and control all aspects of ProtoCycler's operation. While the software may seem overwhelming at first, most users will find that the "Basic" use cases – connecting to ProtoCycler, viewing and controlling manual extrusion, and modifying automatic extrusion profiles – is sufficient. More advanced users are also able to exert a greater range of control over ProtoCycler using advanced commands, and can fully automate both data collection and control.

This guide is intended as a comprehensive reference and includes multiple appendices that cover each aspect in further detail as required. As a recycling business, we encourage you to keep a digital copy handy at all times on the computer on which PCC is installed, rather than printing the full manual. *Please note that this manual applies specifically to version 3.2.3 of PCC, but may be largely applicable to other versions as well.*

Basic Use

This section covers basic use of PCC. After reading, you should be able to install PCC, be familiar with its GUI and layout, understand how to connect to ProtoCycler and both monitor and control the basic parameters of extrusion. In addition, this section covers automatic profile updating and firmware updating, so that you can continue to improve your ProtoCyclers operation with PCC. While updating is not required, periodic updates are easy to perform and will result in significant performance or ease of use improvement.

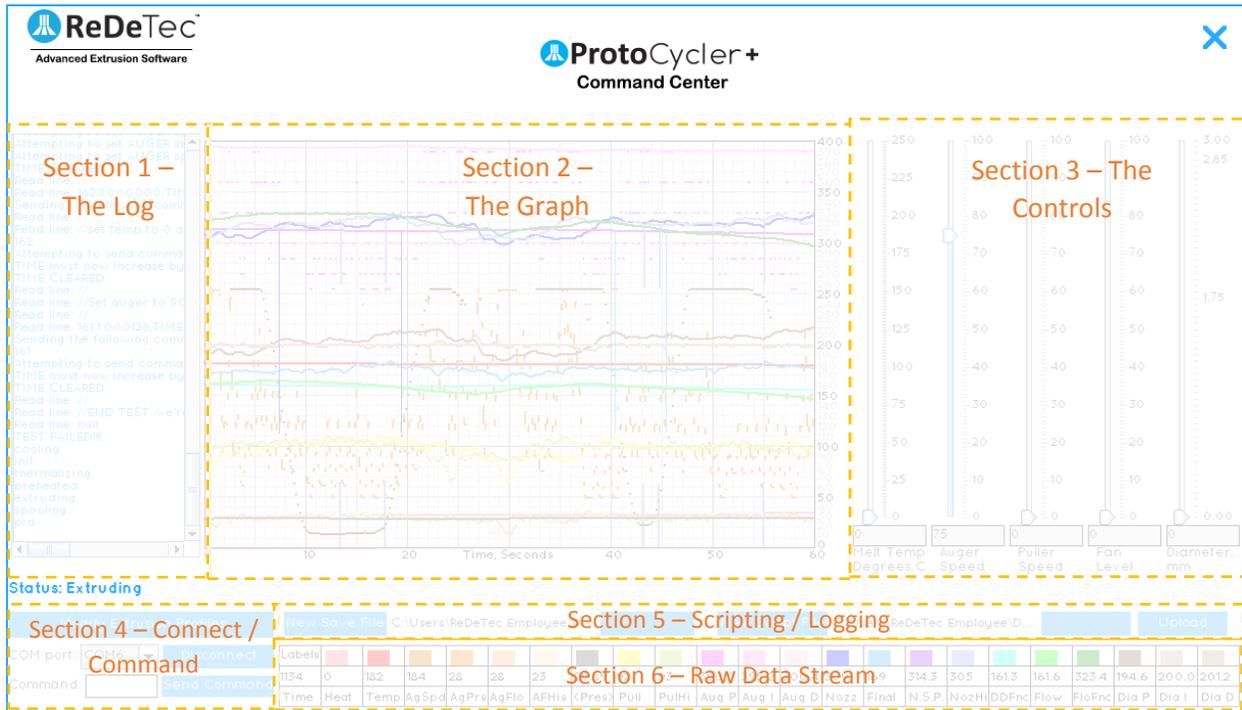
Installation

To install PCC, simply use the provided link in your welcome e-mail, and follow it to completion. If you have lost your welcome e-mail or need a new link, reach out to tech support and we'll provide you with a new one. ProtoCycler Command Center is also available on the software section of our website, www.redetec.com – the most recent version will always be posted. Note that, periodically, updates may be released with new functionality. These updates are always optional, but may require updating the firmware on ProtoCycler itself (see "[Updating ProtoCycler+ Firmware](#)").

PCC is currently available and verified on Windows 7-10 machines. No Mac version is currently available, though we are working on it. PCC requires both Arduino and Java (along with the RXTX package) to fully communicate with ProtoCycler. All of these are included with the installation package and should be present on your system once installation is complete.

Overview

PCC is a graphical program with 6 key sections, as outlined below:



1. The Log

The log is used to display, in a text form, any updates from either PCC or ProtoCycler itself. It will display updates such as “connected”, “spooling”, or other useful information. Each log item is exactly one line, so scrolling to the left and right may be required. Mostly used for debugging purposes or status updates.

2. The Graph

The graph displays all of the data being streamed live from ProtoCycler that enters the Raw Data Stream (See “Section 6 – Raw Data Stream” for more details). As there’s quite a bit of data, different parameters can be turned on and off by clicking on the corresponding colour in the Raw Data Stream. For instance, to hide the “heat” parameter, click the pink square above the “heat” readout in the Raw Data Stream. To bring it back, click the same (now grey) square. To toggle *all* readouts on or off, click the “labels” button in the Raw Data Stream.

The graph displays the past minute of data, with a resolution of 10 data points per second. The Y value for all data is shown on the right of the graph, and the Raw Data Stream is used as a legend. Note that the graph also displays the raw reading of the diameter sensors, shown as two response profiles on the left / right hand side of the graph.

3. The Controls

The controls are used to easily modify the 4 most commonly adjusted parameters of extrusion – the temperature, auger, puller, and fan. Note that the scale for Temperature and Diameter are in real units (degrees C and mm, respectively), while the Auger, Puller, and Fan level are a

percentage (0-100). For all sliders, the real value is displayed in a text box below the slider, and clicking above or below the slider position will increment or decrement the value by 1. The command is only sent when the slider is *released*, so you can click and drag until the desired value is reached, then release the slider to send the command.

4. **Connect and Command**

A bit of a multipurpose section, this box allows you to connect to ProtoCycler, modify the automatic profiles, and send commands. It's fairly self-explanatory, and further details will be covered in the respective sections for each button. The "Modify Extrusion Profiles" button stands alone and will replace the sliders with a section to modify the stored profiles. The COM port selection and "Connect" button are used to connect (or disconnect) to ProtoCycler. Finally, the Command / Send Command section is used to send manual commands to ProtoCycler (see "[Advanced Manual Control](#)")

5. **Scripting / Logging**

Similar to Connect and Command, this section is somewhat self-explanatory, and is mostly limited to advanced control. The "New Save File" and "Collect Data" buttons are used to log CSV files of all incoming data. The "New Script File" is used to select a script for automated scripting operation, *or* to select a new firmware file to upload. "Start Script" is used to begin that script, whereas "Upload" is used to upload that firmware file.

6. **Raw Data Stream**

This section displays the live, realtime readouts from ProtoCycler that are then graphed. It allows you to see the exact numerical value of a specific readout, toggle whether that readout is displayed on a graph, and serves as the graph's legend. See "Section 2 – The Graph" for more details.

Connecting to your ProtoCycler

With the supplied USB cord attached to both ProtoCycler and your computer, power on ProtoCycler. If this is the first time you have ever used ProtoCycler with your computer, it may need to install the driver. ProtoCycler uses the Arduino Leonardo drivers to communicate and will be identified as such. Once ProtoCycler is connected to your computer it will be assigned a COM port – this can be found in the Windows Device Manager if required.

To connect ProtoCycler to the PCC, select the correct COM port from the drop down list in the **Connect and Command** section of the GUI. Note that the list will automatically update each time it is opened. With the correct port selected, click "connect", and confirm that you'd like to connect. Then, on ProtoCycler, navigate to the "connections" menu, and confirm your connection there. *Note – There is a 10 second timeout on both the computer and ProtoCycler at which point a connection attempt stops being active. If both the computer and ProtoCycler are not connected by the user within 10 seconds of each other, the connection will fail.*

Once ProtoCycler is connected, the PCC will become operational, and manual extrusion may begin if desired. To disconnect ProtoCycler, click the "disconnect" button on the PCC, or simply close the PCC. You can also disconnect from ProtoCycler itself. *Note that powering off ProtoCycler before disconnecting*

may cause the PCC to malfunction, or fail to reconnect, etc – in some cases this may require restarting your computer to fix.

Data Readouts

One of the most exciting aspects of using the PCC software is being able to watch your ProtoCycler operate in real time. A wealth of data is displayed on both the **Raw Data Stream** and **The Graph**, and it can be overwhelming for some users to watch it all at once. It's recommended that any data that's *not* important to you is disabled from being graphed, by clicking the coloured square above its readout in the **Raw Data Stream**. By focusing only on the key readouts you need, it becomes easier to observe what is happening.

Some readouts – such as “Temp” – will be relatively obvious. Others, such as “DDFnc”, will be less so! A full list of all readouts, what they mean, and if they're more of a basic or advanced parameter, is included in [Appendix 1 – Readouts Description](#).

Basic Control of ProtoCycler

Controlling the basic aspects of extrusion is possible using the sliders in **The Controls**. Note that the Melt Temp and Fan Level are “direct” controls. This means that dragging the slider will have an immediate effect on the listed parameter. Auger Speed, Puller Speed, and Diameter are slightly more complex. The Auger speed will actually adjust the auger *pressure*, unless the Auger PID is disabled (it is always enabled by default). Similarly the Puller Speed and Diameter controls always override each other. For instance, if you set puller speed to 50, there is no guarantee that the diameter will be maintained at any set value as setting Puller Speed disables all diameter feedback. Conversely, if you set the Diameter slider, the puller speed will constantly change as required regardless of the setting on the Puller Speed slider.

Each slider controls the following aspect of extrusion:

Temperature – controls the temperature of extrusion. Note that extrusion temperature may vary compared to what you 3D print with! For example with PLA, ~180-185 is a good starting point, whereas you may print with it closer to 190-195.

Auger Speed – controls the speed at which the auger rotates, or the pressure if the Auger PID is enabled (which it always is unless you disabled it yourself using advanced manual controls). The Auger pressure dictates how much force ProtoCycler uses to extrude, and along with the temperature, directly influences the speed of extrusion. Note that settings over 75% may lead to increased risk of stalling, where the auger is unable to maintain the pressure requested. This will be noted by a loud clicking or grinding noise, at which point the pressure should be lowered immediately until the auger is able to recover.

Puller Speed – controls the speed of the puller wheel. Using this slider disables diameter feedback automatically, and is useful to get a rough set point when experimenting with new plastics, or to override the diameter feedback system if needed.

Fan Speed – directly controls the fan. Similar to 3D printing, you always want it on max for plastics like PLA, and on lower settings for plastics like ABS. Note that even with ABS you do want *some* cooling, as the plastic must be sufficiently solid by the time it hits the puller wheels so that it won't deform! Note as well that in automatic extrusion, the fan speed is *flow compensated*. This means that if your flow rate drops (or increases) significantly below the average / expected value, the fan will slow down (or speed up) accordingly, to maintain a consistent cooling rate. *Flow Compensation* is not available with manual extrusion.

Diameter – this slider enables full diameter control – both feedback and PID – to attempt to maintain the diameter of the filament to the set value. It will override the Puller Speed control, and is in turn overridden by the Puller Speed control. While the diameter control algorithm is robust, it's not superhuman, and relies on a number of advanced parameters that are only tunable using advanced control. Therefore attempting to set automatic diameter *may* fail, and the outputs – particularly the puller speed and nozzle / final diameters – should be monitored for any instability for a brief period, once this control is enabled.

Modifying Automatic Profiles

Different blends of plastic may require different settings than others. Similarly, new profiles may be developed over the community or by ReDeTec over time. For this reason, PCC allows you to create, delete, and modify the automatic profiles stored on ProtoCycler (though only a limited number of profiles may be stored at any given time).

When you click on “Modify automatic profiles”, the sliders are replaced by the basic automatic profile settings section. To return to manual control and finish modifying the automatic profiles, simply click “Return to Manual control”.

The basic automatic profile section includes buttons to create, delete, or rename the existing profile. The basic values for the current profile are also displayed. Any change in values can then be tested (without saving), or saved to memory. There is also an advanced button that pops up a new window, which shows all of the more advanced settings for an automatic profile – these settings can be saved or cancelled as well, but cannot currently be tested in real time.

Saving a profile does not require you to be extruding, but you must be extruding to test any changes in real time. Profiles do take a few seconds to save, so always ensure that the “Successfully Saved Profile” Dialog appears before closing the profile saving window.

When developing a new profile yourself, a suggested workflow is to find all of the parameters yourself in manual mode, and then create the profile using those settings. A full description of the various settings that can be saved in a profile are available in [Appendix 2 – Profile Settings Description](#). *Note – many of the profile settings are not necessarily intuitive, or may not be used by the current firmware or software version. It is therefore recommended that you only modify profiles yourself if you're confident enough to do more of the advanced features under the “Advanced Use” section. However, we have included modifying profiles under basic use so that you're able to make use of predeveloped profiles.*

Updating ProtoCycler+ Firmware

On occasion, firmware updates will be released for ProtoCycler. These will mostly be aimed at improving performance and functionality, and are therefore strongly recommended (though not required).

ProtoCycler has a special section of code loaded onto it that is used to update the firmware, and which cannot be overwritten. Because of this, even if a firmware update fails for some reason, it is always possible to re-enter firmware update mode and try again. Note also that firmware should not remove or overwrite any of your automatic profiles.

Firmware is updated in the following sequence:

1. With ProtoCycler turned off and your computer turned on, connect the USB cable to both ProtoCycler and the computer. Open up PCC – note that the COM port for ProtoCycler will not appear, as ProtoCycler is turned off.
2. While holding any of the menu buttons on ProtoCycler, turn on the power. This will turn on ProtoCycler into firmware updating mode – the screen should simply say “Update” on it.
3. Click the COM port menu – as it refreshes, ProtoCycler’s COM port should appear. Select the correct COM port ***Do not connect to ProtoCycler!***
4. With ProtoCycler connected, click “New Script File”, and navigate to / select the updated firmware hex file.
5. Finally, click “upload” – read the warnings and confirm that you’d like to proceed. The log should begin showing progress of the firmware updating. ***Note that firmware updates take several minutes (5+) and that ProtoCycler, the PCC, and the connection between them cannot be interrupted during this time!!!***
6. The firmware uploads in two stages. First the new pages are written, then they are verified. Occasionally, verification will fail, and just that page will be rewritten and verified. This is normal.
7. When Firmware updating is done, PCC will automatically disconnect and restart ProtoCycler with the new firmware – the log will also display a number of verification messages that the firmware was successfully verified and updated. If for some reason the firmware update is unsuccessful, disconnect ProtoCycler on the PCC, power off ProtoCycler, and start again at step 1.

Advanced Use

Advanced Manual Control

In addition to using the basic sliders, a number of additional commands can also be sent to ProtoCycler to further influence its extrusion behaviour. These advanced commands do things like change PID parameters for the various control systems, trigger the spreading servo to go to a certain position, trigger spooling to begin, modify the auger speed lower limit, change the timescales of various calculations, etc. *Modifying these commands is for experienced users only, and should not be attempted by those who are not intimately familiar with both plastic extrusion, and ProtoCycler's operation.*

All commands are sent in the following format of two alphabet characters followed by exactly 6 digits (including decimal) indicating a number:

xx#####

Note that to send the value "1", "000001" must be used!

These commands are typed into the box labelled "Command:", at which point the "Send Command" button is used to send the command over. Only one command can be sent at a time, and it must follow the exact (case sensitive) format as shown above. A full list of commands is available in [Appendix 3 – Manual Command Description](#).

In most cases, ProtoCycler will simply echo that a command has been received. In rare cases and depending on firmware, further information may be provided upon receipt of a command. Note as well that it is possible to send a command identical to what a slider would accomplish, *but the slider will not move in this case*. For instance, if the slider is used to set the temperature to 100 C, and then a manual command is sent to change the temperature to 150, the slider will still indicate it is set at 100 C. The Raw Data Stream will, as always, display the real time temperature of the hot end regardless of which method is used to set it. This behaviour is echoed across all the various commands, sliders, and readouts (i.e. the same is true for the puller speed, auger behaviour, fan speed, etc – the sliders will not update if a command is manually sent, but the Raw Data Stream will always display the real data regardless of what triggered the change).

Data Logging and Scripting

One of the most powerful features of the PCC is the ability to log data, and automate nearly every aspect of operation. This allows you to write custom scripts that can, for instance, step through a range of temperatures and pressures for an unknown plastic, and log data for each of them for later processing to determine the best temperature / pressure combination.

Logging data is relatively simple. While extrusion is underway, simply click the "New Save File" button, navigate to the directory of your choice, and create a file – *be sure to include the ".csv" extension. For instance, to save a file called "Extrusion test 1", name the file "Extrusion test 1.csv"*. Then, click "Collect Data" to collect data. To stop collecting, click the same button again. Note that you can record many periods of data in the same file without overwriting the previous collection – repeatedly clicking the

“Collect Data” button will continually append the new data streams to the same file. *However, opening an existing file will wipe and overwrite all data on it!*

Scripting is significantly more difficult, and is unfortunately not currently available with the consumer version of ProtoCycler Command Center. If this is something you’d like to see us prioritize in future releases, please let us know!

Appendices

Appendix 1 – Readouts Description

Below is a full list of all readouts available in PCC. The format for each is the short-form title as it appears on the **Raw Data Stream**, The full name, and whether it's an advanced or basic readout.

Time – Running Time – Basic

The time since unit was powered on, measured in seconds.

Heat – Heater Power – Basic

The current power being sent to the heater, out of 128 (128=full power, 64=50% power, etc)

Temp – Temperature – Basic

The current melt section temperature, in degrees Celsius

AgSpd – Auger Speed – Basic

The current auger speed – unitless. The actual speed the auger is spinning at this instant.

AgPrs – Auger Pressure – Basic

The current auger pressure – unitless. There is very minor filtering of this value to eliminate noise, but it's otherwise the actual auger pressure at this instant.

AgFlo – Auger Flow – Basic

A weighted combination of both speed and pressure – some firmware versions may weight it 100% pressure or vice versa.

AFHis – Auger Flow History – Advanced

The “Filament time” history of the Auger flow when the current dNozz section of filament was being extruded.

<Pres> - Average Auger Pressure – Basic

A running weighted average of the auger pressure.

Pull – Puller Speed – Basic

The current puller speed in real time.

PulHi – Puller Speed History – Advanced

The “Filament time” history of the puller speed when the current dPull section of filament was at dNozz.

Aug P – Auger Feedback P term – Advanced

The “P” gain for the auger pressure PID algorithm – note, this is centered around 300 (as opposed to zero) for legibility. Therefore values less than 300 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

Aug I – Auger Feedback I term – Advanced

The “I” gain for the auger pressure PID algorithm – note, this is centered around 300 (as opposed to zero) for legibility. Therefore values less than 300 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

Aug D – Auger Feedback D term – Advanced

The “D” gain for the auger pressure PID algorithm – note, this is centered around 300 (as opposed to zero) for legibility. Therefore values less than 300 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

Nozz – Nozzle Diameter – Basic

The diameter of the filament as measured at the nozzle sensor.

Final – Final Diameter – Basic

The diameter of the filament as measured at the final / puller sensor after the cooling fan.

N.S.P. – Nozzle Set Point – Advanced

The diameter control algorithms set point for the nozzle diameter – the diameter the system tries to achieve at the nozzle sensor.

NozHi – Nozzle History – Advanced

The “Filament time” history of the nozzle diameter when the current dPull section of filament was at dNozz.

DDFnc – Draw Down Function – Advanced

The expected ratio of diameters between the nozzle and puller sensors, as calculated on an ongoing basis.

Flow – Flow Rate – Advanced

The (unitless) flow rate of plastic through the system.

FloFnc – Flow Rate Function – Advanced

The expected ratio of auger flow to plastic flow through the system.

Dia P – Diameter P – Advanced

The “P” gain for the diameter PID algorithm – note, this is centered around 200 (as opposed to zero) for legibility. Therefore values less than 200 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

Dia I – Diameter I – Advanced

The “I” gain for the diameter PID algorithm – note, this is centered around 200 (as opposed to zero) for legibility. Therefore values less than 200 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

Dia D – Diameter D – Advanced

The “D” gain for the diameter PID algorithm – note, this is centered around 200 (as opposed to zero) for legibility. Therefore values less than 200 are negative. It is also scaled by a factor of 2 to make small changes easier to detect.

[Appendix 2 – Profile Settings Description](#)

The Profiles settings are split into basic and advanced settings. Each setting is listed by name, with a brief description of its relevance. Many if not all parameters receive a default value when a new profile is created. Therefore *all* parameters should be examined when you create a new profile yourself!

Basic:

Melt Temp – The melt temperature used during normal, steady state extrusion after startup is complete.

Pressure – The pressure used during normal, steady state extrusion after startup is complete.

Cooling – The cooling fan speed to use during normal, steady state extrusion after startup is complete.

Diameter – The diameter to strive for, usually 1.75mm

Auger KP – the P term of the auger PID control algorithm

Auger KI – the I term of the auger PID control algorithm

Auger KD – the D term of the auger PID control algorithm

Auger Imax – the maximum amount the I term is allowed to increase for the auger PID control algorithm

Diam KP – the P term of the diameter PID control algorithm

Diam KI – the I term of the diameter PID control algorithm

Diam KD – the D term of the diameter PID control algorithm

Diam Imax – the maximum amount the I term is allowed to increase for the diameter PID control algorithm

Advanced:

Pre Pressure – the pressure used during startup

Max Pressure – the max pressure allowed before an error is flagged, may not be used

Min Pressure – the minimum pressure allowed before an error is flagged, may not be used

Pre Pull – the pulling rate used during startup

Pre Cool – the cooling rate used during startup

Cool Function – how much to vary the cooling speed based on the flow rate

Pre heat – the temperature to thermalize to during startup

Max Heat – the maximum heat allowed before an error is flagged, may not be used

Min Heat – the minimum heat allowed before an error is flagged, may not be used

Flow Function Time Scale – how many loops the flow function should be averaged over

Draw Down Time Scale – how many loops the flow function should be averaged over

Flow Rate Time Scale – how many loops the flow function should be averaged over

Pre Heat Time – how long to thermalize for

Stabilization time – how long to stabilize for

Flow Function Influence – how important the auger flow is to the output flow

Expected Flow – what we expect the plastic to flow at

Smith Gamma – currently unused

Smith Influence – currently unused

[Appendix 3 – Manual Command Description](#)

Below is a full list of all manual commands available in PCC. The format for each is the short-form alphacode, the scripting equivalent, the full name, and the data type expected. The scripting shortform is also listed if applicable. *Please note that entry verification is not performed on commands! Also, case matters...kp \neq kP !*

ff – flow function time scale – Integer

fd – draw down function time scale – Integer

fr – flow rate function time scale – Integer

fi – flow function influence – Float between 0 and 1

tm – current melt temperature – Integer

ts – temperature scale factor – Integer (100=100%)

ma – current auger setting – Integer

mp – current puller speed – Integer

mc – current cooling fan speed – Integer

kp – Auger PID “P” constant – float

ki – Auger PID “I” constant – float

kd – Auger PID “D” constant – float

km – Auger PID I-max term – float

kP – Diameter PID “P” constant – float

kl – Diameter PID “I” constant – float

kD – Diameter PID “D” constant – float

kM – Diameter PID I-max term – float

dd – desired diameter – *Integer, for 1.75mm enter 175! For 1.65mm, enter 165, etc.* Will automatically engage both diameter feedback and diameter PID

pm – set maximum pressure before auger cutoff – Integer

ps – set pressure scale – Integer (100=100%)

pl – set the minimum auger speed during low pressures – Integer

sm – send the servo to the listed pulse length in microseconds – Integer

ss – start spooling, expecting an empty spool – Value ignored, use 000000 for consistency

se – Toggle smith control on / off (not currently used) – Value ignored, use 000000 for consistency

sg – smith control gain (not currently used) – float

si – smith control influence (not currently used) – float between 0 and 1