

Release Notes for CPAP Lite Software

Version 2.4.5

- HEATER CONTROLS: Implementation of Robin's heater requirements:
 - ◆ Heat wire should servo on t_prox, setting at 39 degrees
 - ◆ Heat Plate should servo on t_dist, setting at 37 degrees
 - Heat plate shouldn't exceed 79 degrees, so while ramping up the heat plate, we will actually servo on the heat plate thermistor at "79 degrees", and when we're within 2 degrees of 37 on t_dist, we'll switch over and servo on t_dist at 37 degrees.
- Change "ki" for heat wire from "0" to "1". We were never actually reaching 39 because the integration term was 0.
- Remove printouts of "Prev_flow: .. prev_p_exp:" used for Occlusion alarm debugging
- Fixes for FLASH behavior: make sure we release the semaphore before doing "flash_wait_busy()", because it needs semaphore, too. Also, use semaphore in "flash_erase_sector()", since it was missing.

Version 2.4.4

- OCCLUSION ALARM: Implementation of Tanner's occlusion conditions:

The occlusion alarm should sound while both of the following are true:

 1. The average inspiratory flow decreases by a minimum of 1 LPM over the course of 3 seconds and remains there for at least 5 seconds
 2. The % difference between the inspiratory and expiratory pressure decreases by a minimum of 10% over the course of 3 seconds and remains there for at least 5 seconds.
- NOTES: When the alarm is met, the message appears "Got Occlusion Indication wait 1 seconds" and the red LED's will flash if connected. There is a 5 second delay to "kill" the occlusion alarm, when this happens the message appears "Alarm ending delay complete". There are numerous debug statements on the terminal meant to help debugging the occlusion alarm which will be removed later. If there is a problem, recording all the debug statements will help in trouble-shooting.

Version 2.4.3

- FiO2 CALIBRATION (to O2 sensor) CHANGES:
 - ◆ Allow any calibration factor from 21 to 100 (before, I was limiting it to 80-120%).
 - ◆ Redo the FiO2 calibration whenever the user changes the FiO2 setpoint. Wait 10 seconds after the FiO2 setpoint is reached by the PID loop, then run the FiO2 calibration with the O2 sensor. The O2 sensor takes > 5 seconds to catch up to the FiO2 readings

◆ Requirements to do FiO2 calibration:

1. FiO2 actual is within 0.4 of FiO2 setpoint
2. 10 seconds elapsed since last FiO2 setpoint change
3. Blower > 200
4. O2 sensor reading > 20.0 (when warming up, the O2 sensor reads typically 1 or 2)
5. flow_ox > 0.35

◆ When the O2 calibration factor is found, limit all further FiO2 settings to that factor. For example, if the calibration factor is 80, then don't allow any FiO2 setting above 80.

- I2C ERRORS: Only keep track of I2C errors from the Flow sensor board. I2C errors from the Flow sensors will generate a board reset when they total 100. But I2C errors from the battery monitor or from the power monitor or from the blower will NOT generate errors, no matter how many.
- FIO2 PID LOOP: Use a bias of 4100 for the prop valve instead of 3000. At low O2 tank pressure, the prop valve has no effect below about 4500 setting.

Version 2.4.2

- NEW FEATURE: Set the PID maximum for each of the 4 PID loops using "kmax(x,value)". For instance, to set the heat_plate maximum to 10000 for the PID loop, use "kmax(2,10000)", where: `x=0: fio2, x=1: press, x=2: heat_plate, x=3: heat_wire`. If you do a "saveAll" after entering the maximum, the value you entered is saved to FLASH.
- BUG FIX: the blower limit is now held for 5 seconds after the flow condition that caused the limit is removed. Also the blower "integral" term is limited, so when the condition is active, the integral does NOT increase. Both of these together should ensure that the blower should not increase beyond whatever the blower setting is at when the flow condition is sensed.
- BATTERY MONITOR ON POWER BOARD: is now enabled on the I2C bus. If the cable to the Power board (connecting to P12 on the main board) is missing, this will generate a stream of I2C errors.

Version 2.4.1

- NEW FEATURE: Watchdog implemented. If code hangs up for more than 4 seconds, the "watchdog" should bite and reset the processor. The watchdog is fed in the "Idle Task", which only runs when all the other tasks are not running. (Terminal command "triggerWatchdog" will put the code into an unrecoverable loop, and the watchdog should bite and reset the board after about 4 seconds.)
- NEW FEATURE: Battery backup of RTC. The time should survive power cycles after the watch battery is in place and the "BOD" fuses on the MCU are set. (In the Microship Studio project, go to "Device Programming", click "Apply", select "Fuses" and change "BOD33_DIS"

to unselected, and change “BOD33_ACTION” to “The BOD33 puts the device in backup sleep mode”. Then click “Program” and the battery backup should work.)

- BUG FIX: the blower speed wasn’t updating normally, so remove the “return early” code.

Version 2.3.13

- BUG FIX: Zero pressure sensors works now. There was a conflict between different measurements of the ADC, so there were intermittent bad readings causing the problem (reading other ADC channels when trying to read pressure). Need to put the “semaphoreTake()” BEFORE the set_adc(channel) selection, not after.
- Remove “Baby Pressure Offset”, which was the offset (usually about -0.3) that Baby Pressure showed with the blower off. Now, this will not be used, and the Baby Pressure value will be straight from the equations.
- O2 SENSOR: O2 calibration will not be allowed until the O2 Sensor has been on with status “good” for 10 minutes. The 10 minute timer starts with the O2 sensor status is OK (status 1) and the O2 reading is greater than 5 (for the first 1-2 minutes, the O2 reading is usually 1 or 2).
- BUG FIX: getting “Timer Task” Stack overflow errors, increase stack size from 64 to 100.
- TIMING: reduce the “sample_adc(50)” to “sample_adc(20)”. I don’t believe that a large sampling of the ADC is needed.
- I2C ERROR IMPROVEMENTS: when reading the Flow sensors over I2C, do a “ENTER_CRITICAL()” function to prevent task swapping or RTOS interference. Increase delays in “flow_reset_sensor” to put 200msecs after releasing the RESET line for the Flow sensor power.
- CALIBRATE BUG: when “calibrate” starts, disable the FiO2 and Baby Pressure PID loops. Previously, if calibrate was attempted while under PID control, the PID loop would try to keep taking over from the calibration task.

Version 2.3.12

- DEBUG PROP VALVE: Set DAC0 (pin PA02) for every update to the Prop Valve. PWM is still running (pin PA12), so select desired control signal using jumper on JP1. DAC0 is selected by jumping pins JP1-1. PWM is selected by jumping JP1-2. (See silkscreen on back of board.)
- BUG FIX: In the Flow Sensor reset logic, hold reset for 160msecs instead of 80, and after resetting, delay 250 msecs before resuming I2C operations.
- I2C ERROR RESET: If >100 I2C errors are recorded, then consider that the flow sensors are unrecoverable and do a soft MCU reset.
- DEBUG FREERTOS: Collect RTOS task statistics, display with new command “runTimeStats”. New timer “TIMER_1” runs at 20Khz to collect task statistics.

Version 2.3.11

- BUG FIX: FiO2 will calibrate to the O2 sensor if the flow_ox is > 0.3. It was previously set to calibrate only if flow_ox > 2.0, which meant at low FiO2 settings it never calibrated.
- FiO2 CONTROL LOOP: change bias of PID loop to be FIO2_MIN of 0 (instead of 50), FIO2_MAX is 75 (instead of 95) and FIO2_BIAS is 30 (instead of 63). (FIO2_BIAS is where the control loop starts hunting)

Version 2.3.10

- FEATURE CHANGE: FiO2 will now calibrate to the O2 sensor if the O2 sensor reading is > 25 (changed from > 60).
- BUG FIX: programming “.bat” file now working again

Version 2.3.9

- FEATURE CHANGE: The battery monitor on the Power board is disabled in firmware. To remove electrically from the circuit, disconnect cable from P12 (disconnects I2C). Should improve I2C error problem.
- FEATURE CHANGE: New equations for press_baby, press_ckt_exp, press_ckt_insp and press_ckt.value, calculate c_leak and c_insp in "calibration" when c_factor is calculated.

```
press_baby.value = press_ckt.value - (c_leak * flow_leak.value * flow_leak.value);
```

```
flow_leak.value = flow_insp.avg - flow_exp.avg;
```

```
* float press_ckt_exp = press_exp.avg + (0.00218 * flow_exp.avg * flow_exp.avg) + (0.0623 * flow_exp.avg) - 0.0817;
```

```
float press_ckt_insp = press_insp.avg - (c_insp * flow_leak.value * flow_leak.value);
```

```
press_ckt.value = (press_ckt_exp + press_ckt_insp) / 2.0;
```

MUST BE DONE AS PART OF "CALIBRATION" TO CALCULATE c_Leak AND c_insp

```
calx->press_ckt_exp = calx->press_exp = (0.00218 * flow_exp.avg * flow_exp.avg);
```

```
calx->c_leak = calx->press_ckt_exp / (calx->flow_leak * calx->flow_leak);
```

```
calx->c_insp = (calx->press_insp - calx->press_ckt_exp) / (calx->flow_insp * calx->flow_insp);
```

- In "configASSERT()" print out error message.
- Increase FreeRTOS heap size from 19200 to 24000
- Create binary semaphore for ADC. Prevents different consumers of ADC (on different threads) from interfering with channel selection of the ADC.
- FILTERING CHANGE: Increase standard Flow averaging to 10 (instead of 5); increase standard Pressure averaging to 10 (instead of 6).

Version 2.3.8

- FEATURE CHANGE: O2 analog sensor enabled (use command “o2Status” to read out values)

- O2 CALIBRATION IMPROVEMENT: To toggle the “O2 Calibration” output, which must be high for 1 second, change the code to increase the width of the pulse:

```
gpio_set_pin_level(O2_CALIBRATE_PIN, true);
os_delay(2100); // was os_delay(1100)
gpio_set_pin_level(O2_CALIBRATE_PIN, false);
```

- O2 CALIBRATION PROTECTION: When calibrating the FiO2 to the O2 sensor every 10 minutes, put restrictions on to prevent bad calibration:
 - flox_ox > 2.0
 - O2 reading > 60
 - blower speed > 2000
 - FiO2 within 2% of setpoint in PID loop
 - Resulting calibration must keep factor in range $80 < o2_factor < 120$.

Version 2.3.7

- BUG IMPROVEMENT: Only do averaging of 6 samples for p_insp and p_exp for regular measurements and PID loops, but during c_factor calibration and zeroing pressure sensors do 30-sample averaging for smoother, more accurate results.
- BUG IMPROVEMENT: For I2C errors, When reading I2C for flow, if the result is an error, try waiting 400usec and turn on the flow sensor measurements again (command 0x1000 to the SFM3400)

Version 2.3.6

- BUG IMPROVEMENT: For I2C errors, Put 2usec delay in reading flow sensors (I2C address 0x40).
- Add Flow Sensor soft reset in "init_flow" for resetting I2C errors.
- Add blower control limits during press_baby PID loop: If p_limit > setpt+10 (per Robin's equation), try to limit blower speed (not tested).
- Put PID "max" value into "pidDump" for debugging blower speed limits
- New "delay_usec(n)" function in utils_user.c.

Version 2.3.5

- BUG FIX: Working LCD screens and touch buttons with Rev. 4.5 main board. SPI bus working with LCD and with external FLASH, nor FLASH corruption problems detected. New SPI bus semaphore so different RTOS tasks can share the SPI bus peacefully. Big and small numbers

now updating on screens. NOTE: LCD screens DISABLED in current software to be backwards compatible with old boards.

- BUG PROGRESS: Getting I2C bus errors (scrolling off screen on terminal). Fix for now is the reset flow sensor board and reset I2C bus on main processor, whenever more than 8 I2C errors are detected. Increase I2C Bus semaphore timeout from 40 to 209.
- IMPROVEMENT: Remove FiO2 averaging (per CG). Change pressure PID coefficients to: P=10, I=5, D=0. Change FiO2 coefficients to: P=1.0, I=1.0, D=1.0 for new Enfield proportional valve. Change flow averaging from 50 samples to 5. Change pressure averaging from 50 samples to 3. Change FiO2 averaging from 50 samples to 5. (per CG)
- BUG FIX: Encoding knobs now working. To make them “active” for the setpoints, you must execute the “toggleKnobControl” command, then the knobs directly control pressure and FiO2. In code, “Knob1” is replaced with “Knob_fio2”, “Knob2” is replaced with “Knob_press”.
- BUG FIX: Serial number for boards should now be stored and retrieved correctly

Version 2.3.4

- BUG FIX: in c_factor calculation, make flow_leak = flow_insp-flow_exp. In all previous firmware versions the bad equation was:

- `calx->flow_leak = get_flow_insp() - get_flow_ox();`
- Corrected equation:
 - `calx->flow_leak = get_flow_insp() - get_flow_exp();`

(NOTE: this bug fix is only for the c_factor calculation during calibration; “flow_leak” was calculated correctly in the regular program flow.)

Version 2.3.3

- NEW FEATURE: New operating system “FreeRTOS” implemented. Tasks all running on separate threads, scheduled by RTOS so each runs for an equal part of the time, unless it is suspended. All tasks on equal priority except “Alarm” thread at higher priority. No statistics implemented yet.
- NEW FEATURE: Alarms now implemented:

Implement 4 alarms:

- 1) low pressure ($\text{press_baby} < \text{setpt} - 1.5$)
- 2) high pressure ($\text{press_baby} > \text{setpt} + 1.5$ OR
 $\text{press_baby} > 20$ OR
 p_insp almost equal p_exp)
- 3) low FiO2 ($\text{fio2} < \text{setpt} - 7$)
- 4) low battery ($\text{vBatt} < 20$)

All alarms have a 5 second delay before firing. All alarms have a 3 second delay before cancelling.

All pressure alarms are only active during "Run" state when the pressure is servoing under PID control loops, not during setup or ramping (alarms are disabled during setup and ramping, except battery alarm is always active)

- BUG FIX: Battery voltage now sent up to Host App

Version 2.2.11

- NEW FEATURE: Battery monitor on Power board is now communicating with the main CPU. Using the "battery" command we can now display current, voltage, temperature and charge: Batt. voltage: 25.13V, current: 0.23ma, charge: 95.9%, Temp: 27.3degC
Battery registers are being read once per second, then converted to floating point numbers as needed.
- BUG FIX: When hitting the "Exit" button on the "Low Battery" alarm screen, return to the "Run" screen.

Version 2.2.10

- BUG FIX: When entering kp, kd, and ki parameters (e.g. kp(1,3.4) for baby pressure PID proportional value), the values are setup for FLASH storage. If the command "saveAll" is sent before rebooting, the values are stored in FLASH. The values entered in kp(), kd() and ki() will be used even after reboots or power cycles, as long as the "saveAll" command was entered.

Version 2.2.9

- NEW FEATURE: By enabling the Oxygen Sensor (setting O2_SENSOR to "1" in main.h) the Modbus O2 sensor interface is enabled over the USART_0 port, using pins PA22 and PA23. The O2_SENSOR = 0 by default, so the O2 sensor is normally disabled and the Modbus is not configured by default.
- NEW FEATURE: When the oxygen sensor is enabled, the FlowWorks Hostapp will display O2 readings on the Pneumatic screen (requires v1.1.10 or above).

Version 2.2.8

- BUG FIX: FiO2 calculation is unstable at low flows (low blower settings), so don't calculate FiO2 until blower above 100.
- FEATURE: state machine cycles through available Alarm screens (using "Next Alarm" button) and coordinates with FlowWorks on Windows.
- FEATURE: add yellow, green and red LED's to states "ramping up", "running" and "alarms".

Version 2.2.7

- BUG FIX: Make FiO2 Control loop more responsive by changing PID parameters (using 50PSI pressure)
- When Calibration finishes, move state to “RUN” state so user doesn’t have to push “Next”
- INTERLOCK: Only allow FiO2 loop (Prop valve to open) when blower has started moving
- When FiO2, Baby Pressure are within limits (5% and 0.35cmH2O), move from “ADJUSTING TO NEW SETPOINT” screen to “Running” screen.

Version 2.2.6

- BUG FIX: fix assignments of Pressure and FiO2 setpoints; use 3.0 and 21.0 as default (if nothing in FLASH), don’t read setpoints from knobs unless knob selection is enabled.
- NEW BEHAVIOR: flow of states between “ready to run” to “ramping” to “running” now smooth; “Pause” button now shuts down system
- NEW BEHAVIOR: the “Self Test” screen will now start a “c_factor_calibration”. The old behavior was to start a “power_on_self_test”, which was wrong.

Version 2.2.5

- BUG FIX: when using the HeatPlate PID loop with maximum value of 10000, the standard board smokes. Reduce the PID maximum value to 2000 to prevent board damage.
- ENHANCMENT: handle states of “running” and “pause” in the state machine

Version 2.2.4

- NEW BEHAVIOR: The output stream from “avgStream”, will now include 5 extra parameters at the end: HeatWireSetpt, HeatWireActual, HeatWireControl, HeatWireError, HeatWireIntegral. (NOTE: this may be removed in an upcoming release)
- NEW BEHAVIOR: The 2 output streams “avgStream” and “pidStream” can now work together or separately. If both are requested, they will alternate (avgStream,pidStream,avgStream, etc.). The output rate can still be set with “avgStreamRate(hz)”.
- FIX: the “Self Test” feature will no longer be “blocking”. It will be run as a state machine, reentrant, updated every 20msecs, allowing all other normal functions to run at the same time.

Version 2.2.3

- NEW FEATURE: Create state machine and LCD screen drivers, move through screens with button control.
- HOST SUPPORT: Host can change screen by sending button changes, e.g. “button(4)” for “Next”, and Host can read back screen status to know which screen to display.

- BUG FIX: In the “avgStream” make sure that FiO2 is always less than or equal to “100”.

Version 2.2.2

- NEW FEATURE: Support Windows Host App (called “FlowWorks”) by uploading data and status on demand to support extensive GUI interface showing sensors and controls.
- NOTE: FlowWorks on the laptop won’t work with any firmware version less than 2.2.1.

Version 2.2.1

- BUG FIX: heat wire PWM now working. Now all 6 PWM controls work together.
- Convert “calibration” and “wait for blower to stop” routines to all “re-entrant” code, so nothing stalls (hogs CPU) waiting for completion
- NEW FEATURE: sendStatus and sendData commands coming from Host App “FlowWorks” now supported

Version 2.1.20

- FEATURE CHANGE: Max plate heater control value in the heat plate PID is increased from 5000 to 10000 to get the full 6A of current running to the heater plate when under PID control.
- FEATURE CHANGE: Now support knobs using encoder instead of potentiometer. Signals KNOB1 and KNOB2 will now generate interrupts (`irq_FIO2_PB00_encoder` and `irq_PRESS_PB01_encoder`), and SW1 and SW2 are connected to the other 2 encoder signals, which are combined to sense “CW” and “CCW” rotation pulses.
- FEATURE: The “power” command now will include the battery voltage of the on-board watch battery (read through the ADC)
- KNOWN LIMITATION: The “heat wire” PWM control is not working, so it is disconnected. The heat wire will NOT work in this release.

Version 2.1.19

- FEATURE CHANGE: Instead of printing out the column headers in "avgStream" every 15 lines, only print them once on startup, and never again. Put a comma at the start of every line printed in avgStream to make Excel imports cleaner.
- Example Log from TeraTerm with timestamps and commas (no headers):

```
[2021-04-01 21:36:28.216] ,000.53,000.14,01.83,00.00,20.76,19.91,20.34,27.50,021.07,000.38,000.65,001.18
[2021-04-01 21:36:28.415] ,000.53,000.14,01.83,00.00,20.77,19.90,20.34,27.50,021.07,000.38,000.65,001.18
[2021-04-01 21:36:28.614] ,000.53,000.14,01.83,00.00,20.77,19.90,20.34,27.50,021.07,000.38,000.65,001.18
[2021-04-01 21:36:28.813] ,000.53,000.14,01.83,00.00,20.77,19.90,20.34,27.50,021.08,000.38,000.66,001.17
[2021-04-01 21:36:29.016] ,000.53,000.14,01.82,00.00,20.77,19.91,20.34,27.49,021.08,000.38,000.65,001.17
[2021-04-01 21:36:29.216] ,000.53,000.14,01.82,00.00,20.77,19.91,20.34,27.49,021.08,000.39,000.66,001.17
[2021-04-01 21:36:29.416] ,000.54,000.14,01.82,00.00,20.77,19.91,20.34,27.49,021.08,000.39,000.66,001.17
[2021-04-01 21:36:29.616] ,000.54,000.15,01.83,00.00,20.77,19.91,20.34,27.49,021.09,000.39,000.66,001.16
[2021-04-01 21:36:29.816] ,000.54,000.15,01.83,00.00,20.76,19.91,20.34,27.49,021.10,000.39,000.66,001.16
```

Version 2.1.18

- BUG FIX: Make the interval in printouts in “avgStream” more precise. There was a bug where a 5Hz output rate (“avgStreamRate(5)”, the default) was resulting in 220msecs period instead of 200 msec period. This is fixed, and should give more accurate intervals for all settings of avgStreamRate(x).

Version 2.1.17

- IMPROVEMENT: Change the blower open loop startup accel from .22 to 57 to reduce startup lag. Change the closed loop accel from .04 to .77 to speed up blower speed changes. This decreases the blower lag on startup from 7 seconds to 2 seconds, and speeds up the movement between blower speeds.
- INTERLOCK IMPROVEMENT: Add a feature to limit the blower output depending on the pressure setpoint. By measuring typical blower vs. baby pressure values, I created a zone of possible blower settings, and limit the output of the pressure PID loop to these values to prevent blower run-away when the circuit is opened.
- BUG FIX: the zero_press_baby calculation is fixed to work now
- INTERLOCK: if the temperature sensors are disconnected, the heater PID loops are disabled for both the heat plate and heat wire
- INTERLOCK: if the blower isn't running, the heat wire PID loop is disabled (need air moving through the tube to reach the temperature probe near the baby, or the heat wire could run away)
- FEATURE: the up-arrow will now go through the last 10 commands (the down-arrow moves the opposite way in the buffer)

Version 2.1.16

- BUG FIX: if the temperature probe is removed, or the circuit is completely removed from the humidifier, while the baby pressure is being controlled by the PID loop, we will now just hold the blower at whatever value it was at until normal pressures are restored, then continue the PID loop. We use the baby pressure < -1.0 as the indicator that the pressure is not valid to indicate we should NOT run the PID loop for baby pressure, just leave the blower at whatever value it was at.
- BUG FIX: as a backup to the previous fix, the PID integral term for baby pressure will be limited to 511. There were situations (like opening the circuit) where the integral term would keep increasing out of control, which means the PID loop took minutes to respond while the blower was at maximum and the integral term was greater than 3,000.

Version 2.1.15

- BUG FIX: wait for blower speed to drop to 0 before continuing. Avoids problem of trying to zero pressure sensors before blower has a chance to drop to 0.
- LED strip color fully adjustable. Using command “leds(x,y,z)” the three colors R, G, B, can be adjusted, with the range in percent, 0-100. For instance bright white is “leds(100,100,100)”, dim yellow is “leds(10,10,0)”, medium blue is “leds(0,0,50)”. The 3 LED controls are now controlled by PWM signals.

Version 2.1.12

- Add “F_LEAK” to avgStream printout (at the end)
- Add “zeroBabyPress” command so if the baby pressure at “no pressure” is -0.3, this command will figure out the offset and use that to make the calculation of baby pressure more accurate
- (Move the state machine from “main.c” to new module “states.c”)
- (Fix bug in “calibratePress” routine.)

Version 2.1.9

- New repo using AtmelStart tool: EqualizeHealth_CPAP
- RTC and date and time implemented (using internal 32kHz oscillator)
- Power monitoring:
 - Bus: 24.15V, shunt: 3.711mV, current mA: 74.22mA, power: 275.42mW
- Command “calibrateFiO2” figures out the bias in the Prop Valve control for a given O2 tank pressure, typically around 73% at 50PSI.
- Tuning of FiO2 PID loop for less overshoot

Version 2.1.4

- New terminal interface
- Features:
 - Help screen, type “?” or “help”
 - Input ignores case (capitalizations don’t matter)

- Arguments for commands are given within parentheses, e.g. “PressSetpt(2.5)” will interpret the argument as value 2.5 for the baby pressure setpoint.
- “Backspace” and “up-arrow” supported, allows backing up to delete inputs, and up-arrow will return last command
- Help screen shows firmware version (e.g. Version: 2.1.4) and Board Serial Number (e.g. B100104)
- “rawStream” will output raw sensor readings at 50mS interval (e.g. 958 for pressure); inputting “rawStream” a 2nd time toggles the output off
- Typing anything will momentarily halt the stream
- “avgStream” will output converted sensor readings at 200mS interval (e.g. 1.57 for pressure)
- All output devices can be driven directly:
 - blower(200) – runs the blower (open-loop) at 200 out of maximum 511
 - propValve(50) – opens the O2 proportional valve at 50%
 - heatPlate(5000) – turns on the heater for the humidity chamber at 5000 out of 10000
 - heatWire(6000) – turns on the heater for the wire in the circuit at 6000 out of 10000
- For PID control, first set the setpoint for the 4 parameters:
 - FiO2Setpt(50) – sets the desired FiO2 reading to 50% oxygen (range 20 – 100)
 - heatPlateSetpt(50) – sets the desired heat plate temperature to 50 degrees C.
 - heatWireSetpt(30) – sets the desired circuit temperature to 30 degrees C.
 - PressSetpt(2.8) – sets the desired baby pressure to 2.8 cmH2O
- To engage active PID control:
 - togglePIDHeatPlate – will actively servo on the heatPlateSetpt
 - togglePIDHeatWire – will actively servo on the heatWireSetpt
 - togglePIDFiO2 – will actively servo on the FiO2Setpt using the proportional valve
 - togglePIDPress – will actively servo on the Baby Pressure using the blower
- “pidStream” will output the control and setpoint values for all the PID loops
- “pidDump” will output the PID loop values once, plus the PID constants: kp, ki, and kd.
- “showFlash” will show all the values that are primed to be stored in FLASH, or have been loaded from FLASH at startup
- “saveAll” will save the parameters in FLASH, e.g. c_factor, pressure offsets (from zeroSensors), and setpoints

- “SerialNumber(B100104)” will set the board serial number to the argument (e.g. B100104) and will be permanently stored in FLASH after the command “saveAll”.
- Reboot – will allow the board to be reset from the terminal (e.g. to abort all PID functions)
- “calibrate” – runs the “zeroSensor” command to calculate the pressure sensor offsets at no pressure, then runs the blower at several different speeds and calculates c_factor for the given disposable rig. These calibrated values will be permanently stored in FLASH with the “saveAll” command
-