2G ESP
Water Sampling

5/25/22 Brent Roman brent@mbari.org

Syringe Pump Driven Sampling

- The 2G ESP uses its 25ml Sampling Syringe
  - to pull water through filter in Collection stage
  - while monitoring syringe pressure
- Syringe stops when pressure in it falls too low
  - until pressure equalizes
  - resumes at 90% of the previous pumping rate
- Different Samplers optimized for each filter type
  - BACsampler, HABsampler, DAsampler, etc.
    - see utils/sampler.rb, utils/gensampler.rb,
    - and protocol/2G/[BAC,HAB,DA].rb

2G Sampler Parameters (1 of 5)

Defined in utils/gensampler.rb:

- vacuum defaults to 10..13 [PSI]
  - target vacuum range while pulling through filter
- minRate defaults to 0.2 [ml/second]
  - Filter is clogged when filtering at minRate
    - generates vacuum > vacuum.max
- avgRate defaults to 0.2 [ml/second]
  - Constant filtering rate (used only during simulation)

2G Sampler Parameters (2 of 5)

Defined in utils/sampler.rb:

- bubblePt defaults to nil [PSI]
  - Filter's characteristic bubble point (*must be spec'd*)
- evacVolume defaults to 10 [ml]
  - How far to pull back SS to evacuate puck
- endVacuum defaults to 3 [PSI]
  - Vacuum must fall below this before starting next stroke
- maxDelay defaults to 40.seconds [seconds]
  - How long to wait for endVacuum to be reached
    - Filter Clogged if vacuum still > endVacuum after maxDelay

2G Sampler Parameters (3 of 5)

Defined in utils/sampler.rb:

- backoff defaults to 0.9 [seconds/seconds]
  - new filtering rate after restarted due to high vacuum
- evacDelay defaults to 40.seconds [seconds]
  - how long maintain high vacuum during puck evac
- numberOfEvacs defaults to 1
  - number of times to repeat puck evacuation
- maxPumpVolume defaults to 40 [ml]
  - max vol of air pumped to reach bubblePt during evac

2G Sampler Parameters (4 of 5)

Defined in utils/sampler.rb:

- equalized defaults to 1.0 [PSI]
  - Intake blocked if pressure will not equalize to within this psi
- minPSI defaults to 2.5 [PSIa]
  - Minimum sampling pressure (to avoid boiling sample)
- maxResidue defaults to 0.1 [ml]
  - don’t take another stroke if remaining vol < maxResidue
- primeVolume defaults to 7 [ml]
  - volume of initial prime stroke
- maxPumpVolume defaults to 40 [ml]
  - max vol of air pumped to reach bubblePt during evac

2G Sampler Parameters (5 of 5)

Defined in utils/sampler.rb:

- meterInterval defaults to 30.seconds [seconds]
  - Flow meter reporting interval
- meterDelta defaults to .1 [Liter/min]
  - report flow changes only if > meterDelta
- exhaleConfig defaults to nil [I2C::Servo::Configuration]
  - SS configuration for exhalation

Sampling in the Lab

```
-> mySampler = Sampler.new bubblePt:26
```

… verify parameters are to your liking ...

```
-> mySampler.maxDelay=60.seconds #increase maxDelay
-> CC.loadPuck :sh1     #load puck from startTube
-> mySampler.prime      #take the optional prime stroke
-> mySampler.sample 55    #sample up to 55ml more
-> mySampler.avgRate=2    #simulate faster!
-> mySampler.sample 125   #sample up to 125ml more
-> mySampler.puckEvac     #dry the puck
-> CC.unloadPuck        #put spent puck into discard tube
```

collectSample

```
-> CC.loadPuck(:sh1) {collectSample(mySampler).puckEvac}
```

- Loads sh1 puck from startTube
- primes and samples default goal volume with mySampler
- dries the puck
- discards it into startTube-1
- replace mySampler with any of the predefined samplers
  - BACsampler, HABsampler, DAsampler, etc.

Sampler Pressure Sensing

```
-> Sampler.intakePSI   #abs pressure above the filter
-> Sampler.outletPSI   #abs pressure below the filter
-> Sampler.PSI    #[abs pressure above, abs pressure below]
-> Sampler.calibrate   #tare pressure sensors
```

- A dry puck must be clamped in the collection stage
- Must tare after a new puck is clamped

- For hardware debugging and initial calibration:

```
-> CS.status.outPress   #intake pressure in raw ADC counts
-> SS.status.outPress   #outlet pressure in raw ADC counts
```

- Raw pressure ADC counts range 0..4095

External Sampling Aids

- External (Dumb) Pumps
  - Switched Lift Pumps
- Intake selector valves
  - Choose between shallow and deep sample intakes
- Passive Flow Meters
  - Verify water flowing through sample intake loop
- Active Flow Controllers
  - Combine meter with pump controller to regulate flow
  - Also controls an external intake selector valve

Configuring External Pumps

- Switch via ESP's :sampler external power switch
- Sampler::PumpPrimeDelay
  - defines how long pump runs to prime external loop
  - before ESP intake opens
  - typically defined in machine's configure.rb as:

```
class Sampler
   const :PumpPrimeDelay, 3.minutes
end
```

Intake Selector Valve

- External Valve selects between deep and shallow intakes
  - Only installed on GLERL ESPs
  - Connected as a Gripper
    - to the Sampler Dwarf microcontroller
  - Example config at end of ESPnessie's configure.rb:

```
require 'depthsel'
:Sample.denotes DepthSelectorValve "Depth Selector Valve", :sampler, 1
Sample.maxDuration=30 #allows up to 30 seconds for valve to move
```

- Valve must be explicitly positioned before sampling begun

```
-> Sample.shallow   #select shallow intake
-> Sample.deep      #select deep intake
```

Passive Flow Meters

- Two types in use:
  - Analog input (by WHOI)
    - may function as a simple flow presence/absence switch
    - or uncalibrated % of full flow
    - in ESP’s Can data
  - Serial input (by NOAA NWFSC/APL)
    - on ESP’s external sampler RS-232 serial port
    - logs Liters/minute every minute
    - automatically enabled during sampling

Active Flow Controller

- Developed by APL with MBARI
  - combines digital flow meter with pump speed controller
  - ESP sets desired input loop flow rate during sampling
  - Powered by External Sampler power switch
  - Communicates on External Sampler serial port

Active Flow Controller Config

- Configuration in configure.rb:

```
class Sampler
  const :PumpPrimeDelay, 3.minutes #prime before sampling
  const :PrimeFlow,    2.0..7.0    #Liters/min flow range while priming
  const :MaintainFlow, 1.0..3.5    #flow range while sampling
end
```

- Configuration in corecal.rb:

```
  flowScale = 0.00276964  #scales counts/min flow to liters/min
  @@flow = Skew  flowScale,-flowScale, format: '%.1fL/min', threshold:0,
                 driver: Gauge::Flow::UWFlowControl,
                 port:'/dev/extSampler', switch: :sampler
```

Active Flow Controller Use

- See lib/gauge/uwflowcontrol.rb

```
-> Flow.start      #start monitoring flow
-> Flow.selectIntake :shallow  #select shallow intake
-> Flow.selectIntake :deep     #select deep intake
-> Flow.intake     #verify intake selection
:deep
-> Flow.prime      #run pump at faster priming flow rate
-> Flow.maintain #run pump at slower sampling flow rate
-> Flow.rate       #returns current flow rate in Liters/minute
3.2
-> Flow.disablePump     #stop pump
-> Flow.stop       #stop pump and stop monitoring flow
```