ISU KLM Meeting

September 13

Noah Brenny, Avinash Khatri

Summary

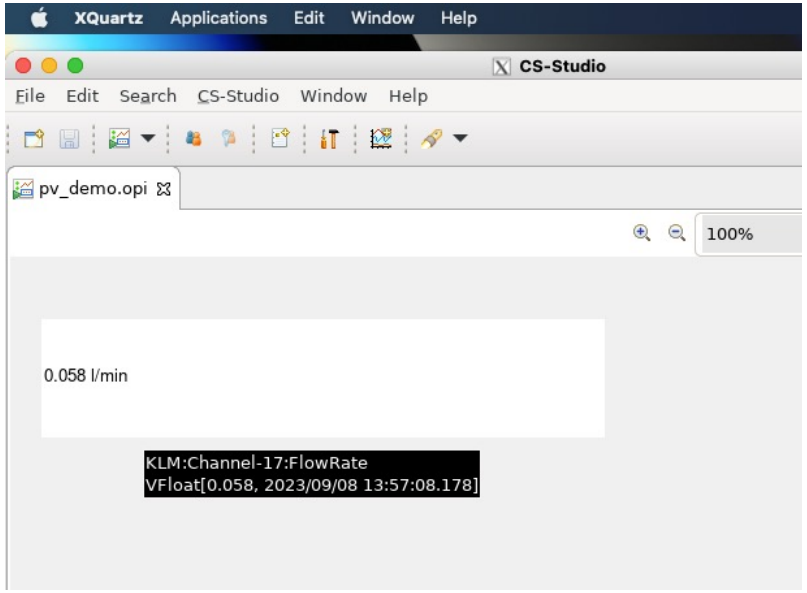
1. b2nsm networking update
2. GUI first steps
3. GUI additions
4. GUI next steps
5. Other items

Networking Update

- Nakao-san agreed to add one RaspberryPi to the b2nsm network
  - pi2 now has an IP on b2nsm: klmpi2.b2nsm.kek.jp has address 172.22.35.228
  - No news on KLM private network yet, but Michele indicates it should be simple once Sourav is free ~ September 30
- pi2 has ethernet connection to KLM switch
  - Can use ssh from bdaq: “ssh pi2@172.22.35.228”, password: Belle2
  - Reminder: readout board attached to BB3 outer RPC return side bubbler
- Can read see PVs on network: see next slide

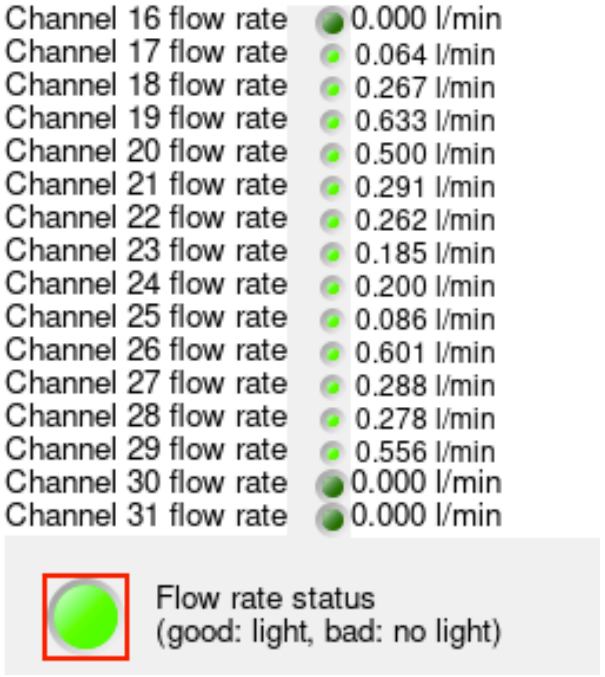
Bubbler Monitoring GUI

- Tommy Lam suggested we use a CSS installation on klmpc03 for development
- First step: we can see PVs on CSS



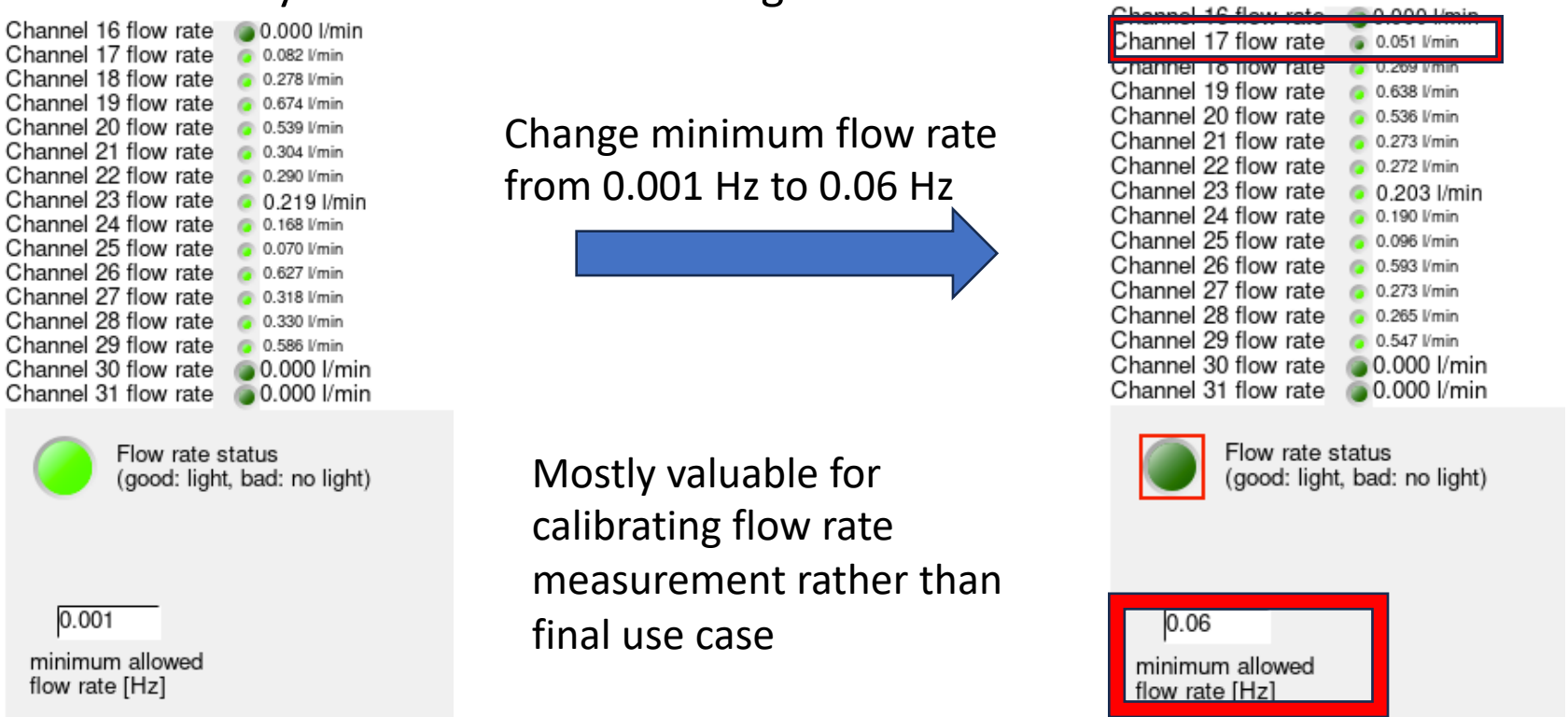
Bubbler Monitoring GUI

- Second step: basic GUI
  - LEDs indicate channel status (bubble rate > 0 Hz)
  - and(channels<16-31>) LED indicating if all channels are running
  - Channels 16, 30, and 31 are disconnected at the bubbler panel, so exclude them from the and() LED



Bubbler Monitoring GUI

- Third step: various additions
  - Adjust minimum allowed flow rate that light up an LED
    - e.g. if our code tells us that a disconnected channel's bubble rate is 0.001 Hz, only channels with a bubble rate larger than 0.001 Hz lights up an LED
    - User can vary the value while running



Bubbler Monitoring GUI

- Third step: various additions
  - Maybe a channel other than 16, 30, 31 is expected to have less than minimum bubble rate
    - Add Boolean controls that indicate/control this behavior
    - Boolean buttons only visible when a channel has less than minimum bubble rate

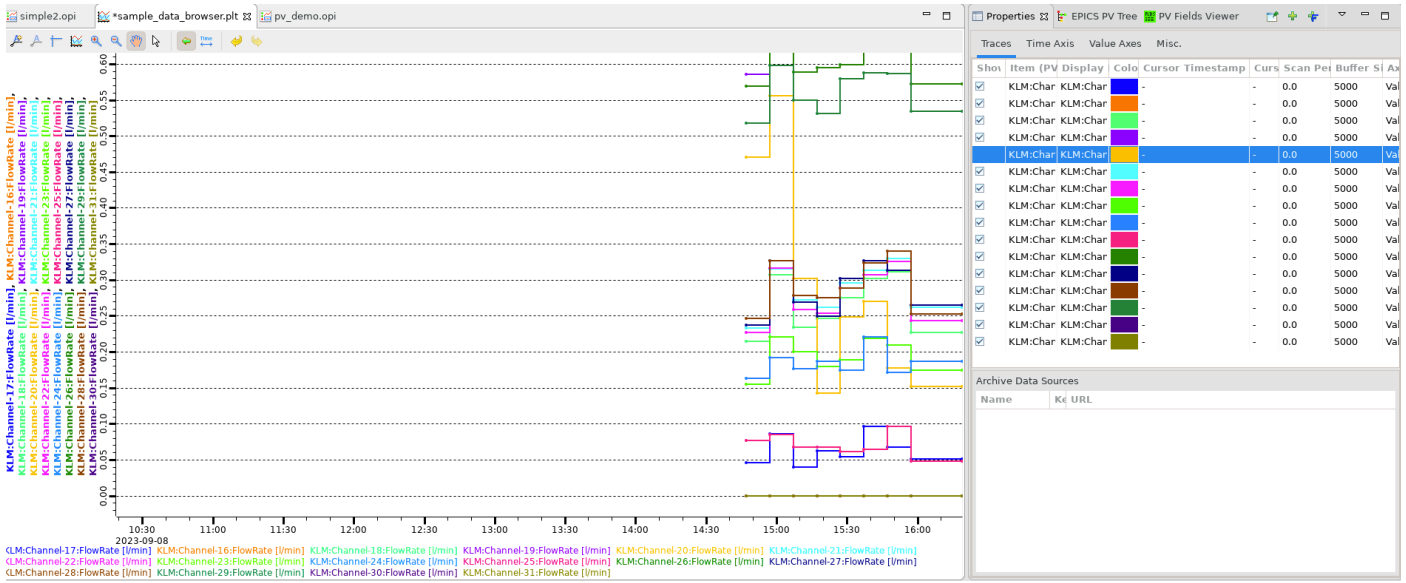
Expected case

Case where minimum bubble rate is expected to be > 0.1 Hz

Bubbler Monitoring GUI

- Third step: various additions
  - Open a time series plot of the 16 channels' bubble rates

Click to open
Data Browser of
all 16 channels



This still needs some adjustment

Bubbler Monitoring GUI

- Next steps
  - Would like to have a home page with and() LEDs for each RaspberryPi with subpages like the one we have developed, reachable by a click
- Maybe we can see how it works developing with Phoebus
  - After attending “Discussion for migration from CSS to Phoebus” meeting, it seems it might be best to just start from Phoebus
  - These GUI files are not very complicated, so they should be fine
  - Might still be worth a shot to start right from Phoebus

Other Items

- PV naming
  - What is the configuration again?
    - For each Pi there are how many channels?
    - For each sector there are how many Pis?
  - Currently the PV names for one board, one ribbon cable are “KLM:Channel-<*i*>:FlowRate” with *i* 00-63
  - Would prefer to switch to “KLM:Bubbler:Sector<*j*>:Channel<*k*>:FlowRate”
    - More consistent with existing PV names from other subdetectors
    - Sector number determined by hostname?
      - Can have identical copies of the software on each Pi
- Archiving
  - Not yet completed
  - Will contact Seokhee Park or whoever is now responsible for registering PVs to b2arch1