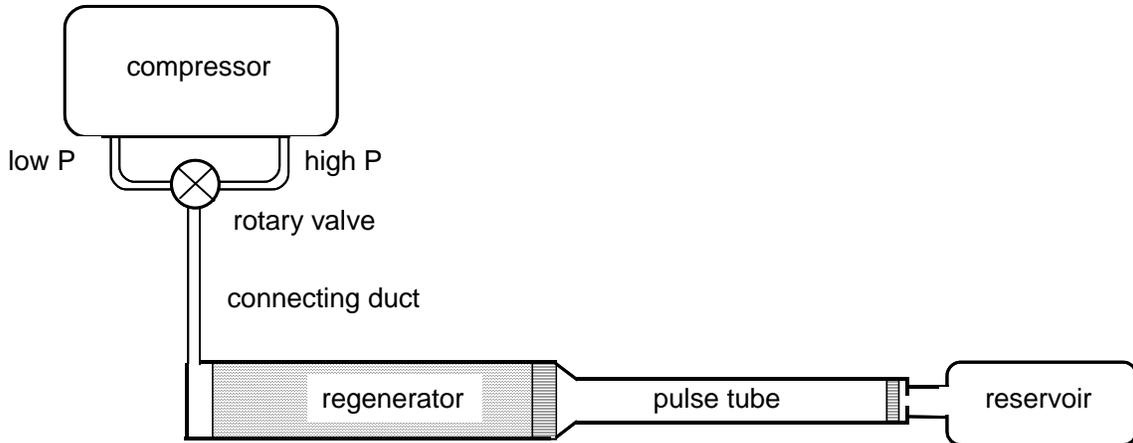


# Sage Model Notes

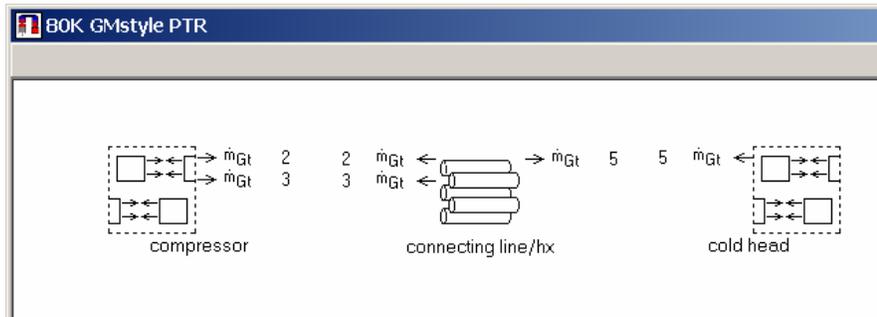
## GMSingleStage.ptb

D. Gedeon  
9 January 2009

A model for a single-stage GM-style pulse-tube refrigerator represented schematically like this:



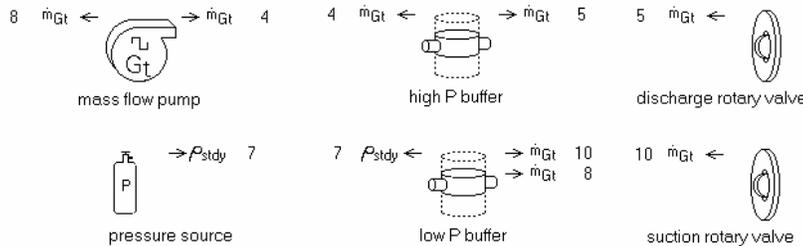
At the top level the Sage model consists of a compressor submodel and a cold-head submodel with a connecting line between the two:



The *compressor* provides a positive (directed toward right) high-pressure flow through connection '2' and a negative low-pressure return flow through connection '3'. The *connecting line/hx* serves as the compressor aftercooler during the high-pressure flow and also rejects heat from the cold-head during the low-pressure flow. In a more detailed model there might be separate heat exchanger components at either end of the connecting line, depending on the actual hardware implementation.

# Compressor

Within the compressor a mass flow pump provides a steady mass flow rate (set by input  $FRhoUA$ ), removing gas from a low pressure space (*low P buffer*) and discharging into a high pressure space (*high P buffer*).



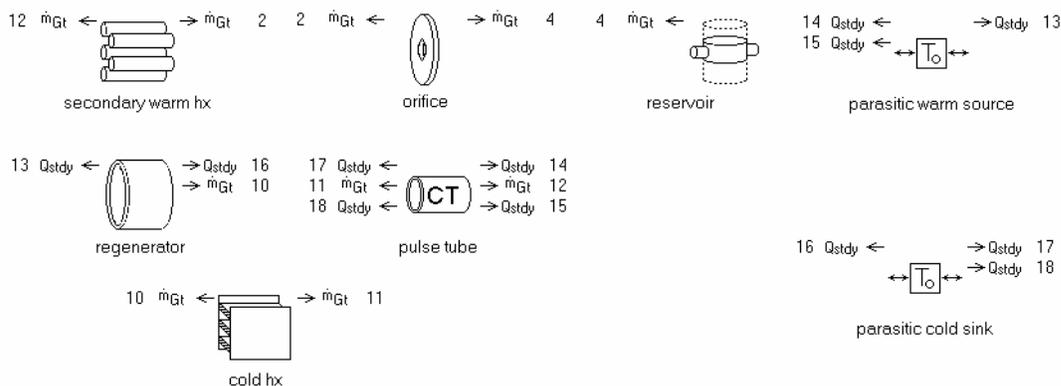
The rotary valve is modeled in two pieces using two time-dependent valve components. The discharge piece allows flow from the *high P buffer* in the positive direction with a sinusoidally varying flow area (FRrestrict input) during the open half the cycle. During the closed half of the cycle it remains closed (actually with a very small flow area for numerical reasons). The suction piece is similar, except its open time is phase-shifted 180 degrees from the discharge piece. It allows return flow in the negative direction back into the *low P buffer*. This is pretty much what is going on in a real GM rotary valve.

Sage's optimizer adjusts compressor mass flow rate (FRhoUA.Mean) subject to the constraint that the discharge pressure (FPmean.Mean of discharge buffer gas) equals 2.0 MPa. The supply pressure is automatically maintained by the charge bottle component. If you change component dimensions, you may have to re-optimize to bring the discharge pressure to its correct value. Either that, or adjust compressor input variable FRhoUA.Mean by hand.

**Ideal adiabatic compressor power** is available in user variables Wcompressor (top level) and Wadiabatic (defined within compressor component). These are calculated from the adiabatic compressor power equation, based on the mass-flow-rate through the compressor and the supply and discharge pressures.

# Cold Head

In the top-level edit window the components are laid out as an in-line arrangement with warm parts at the top and cold parts at the bottom.



Discharge flow from the compressor enters at the negative (left) boundary of the *regenerator* and proceeds through the *cold hx*, *pulse tube*, *secondary warm hx*, *orifice* and into the *reservoir*. During the suction part of the cycle the flows are reversed. The orifice serves to phase-shift the flow relative to the pressure so as to accomplish an approximation of the stirling cooling cycle.

**Cold Temperature** The file is currently set up for a cold-end temperature of 80K. To change it change the *Tinit* input for the *cold hx* and *T* for the *parasitic cold sink*. These remain fixed at the specified temperature. Also, you might want to change *Tinit* at the positive end of the regenerator and negative end of the *pulse-tube*. These changes are not strictly necessary because the regenerator and pulse-tube temperatures are solved, with *Tinit* only providing initial values.

**Net cooling power** is available in the top-level user-defined variable *Qlift*. Included in *Qlift* are the heat flows absorbed by the helium in the cold hx, less the conduction losses down the regenerator and pulse-tube canisters and the radiation loss down the pulse tube.

## Optimization

In addition to the compressor variables mentioned above, there are other variables selected for optimization with the objective of maximizing cooling power (*Qlift*). These are the orifice hole diameter (*Dorf*) regenerator length (*Length*), diameter (*Din*) and wire diameter (*Dwire*). Also the pulse-tube length (*Length*) and diameter (*Dtube*). The pulse-tube length is constrained to be at least six times the diameter to avoid modeling in a non one-dimensional flow regime. This constraint may be relaxed or modified, depending on your experience with pulse-tubes.