Cascade Control

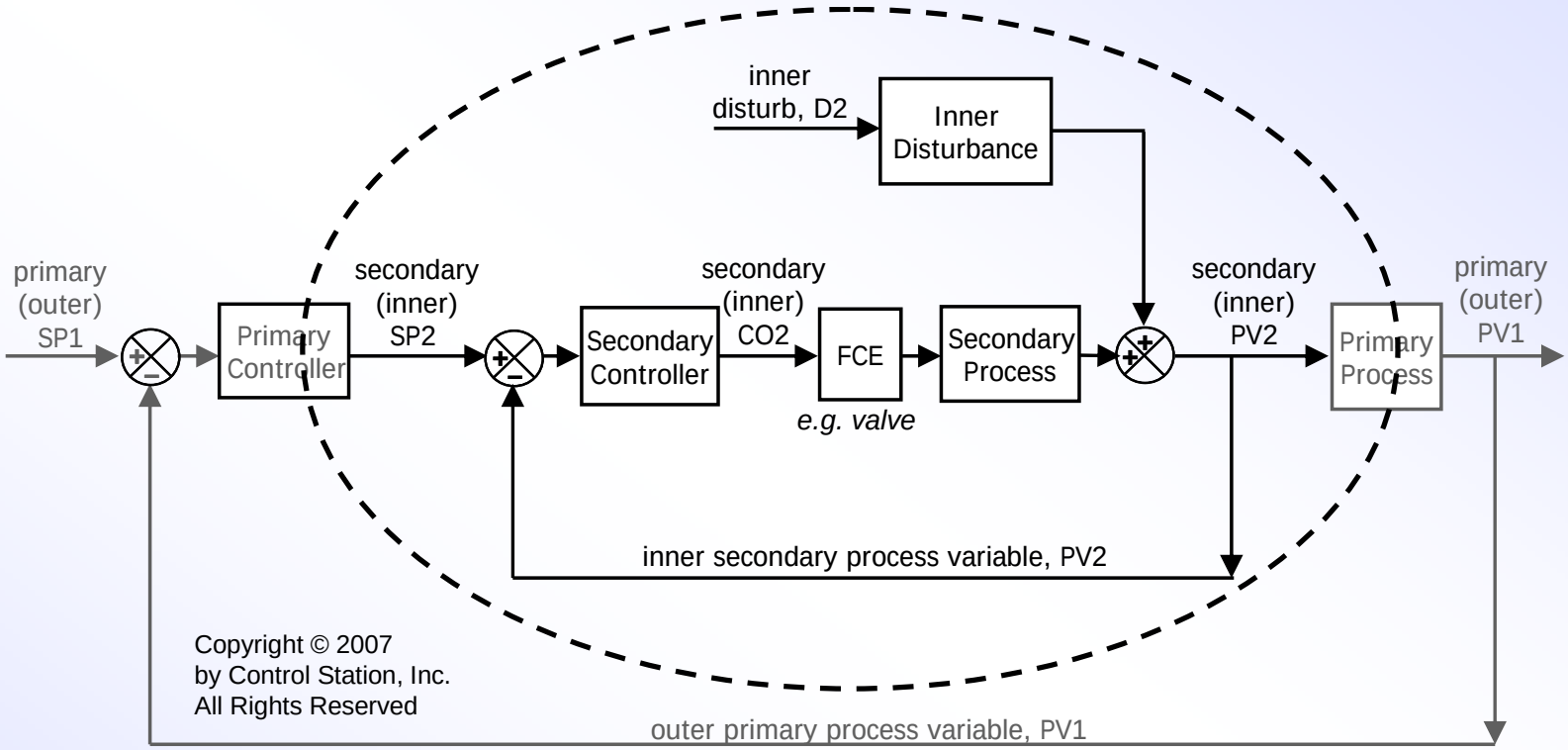
What We Will Learn in This Section

- The Cascade Control Architecture
- Benefits of a Cascade Strategy
- Design and Tuning a Cascade Controller
- Application to a Flash Drum Process
- Application to a Jacketed Reactor

Cascade Control

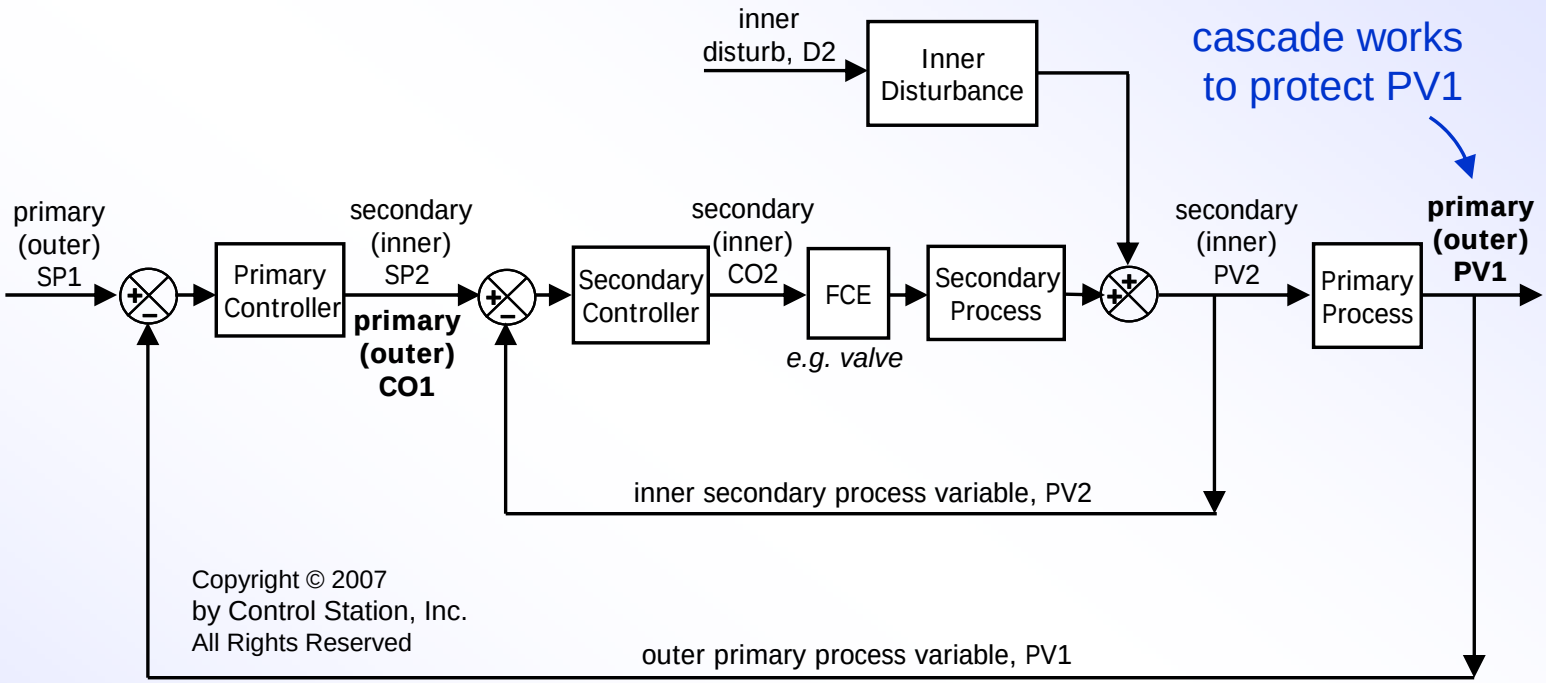
- Architectures for improved disturbance rejection
  - Feed Forward
  - Cascade
- Both require additional instrumentation and engineering time in return for a controller better able to reject disturbances
- Neither architecture benefits nor detracts from set point tracking performance

Traditional Feedback Loop is in the Dashed Circle



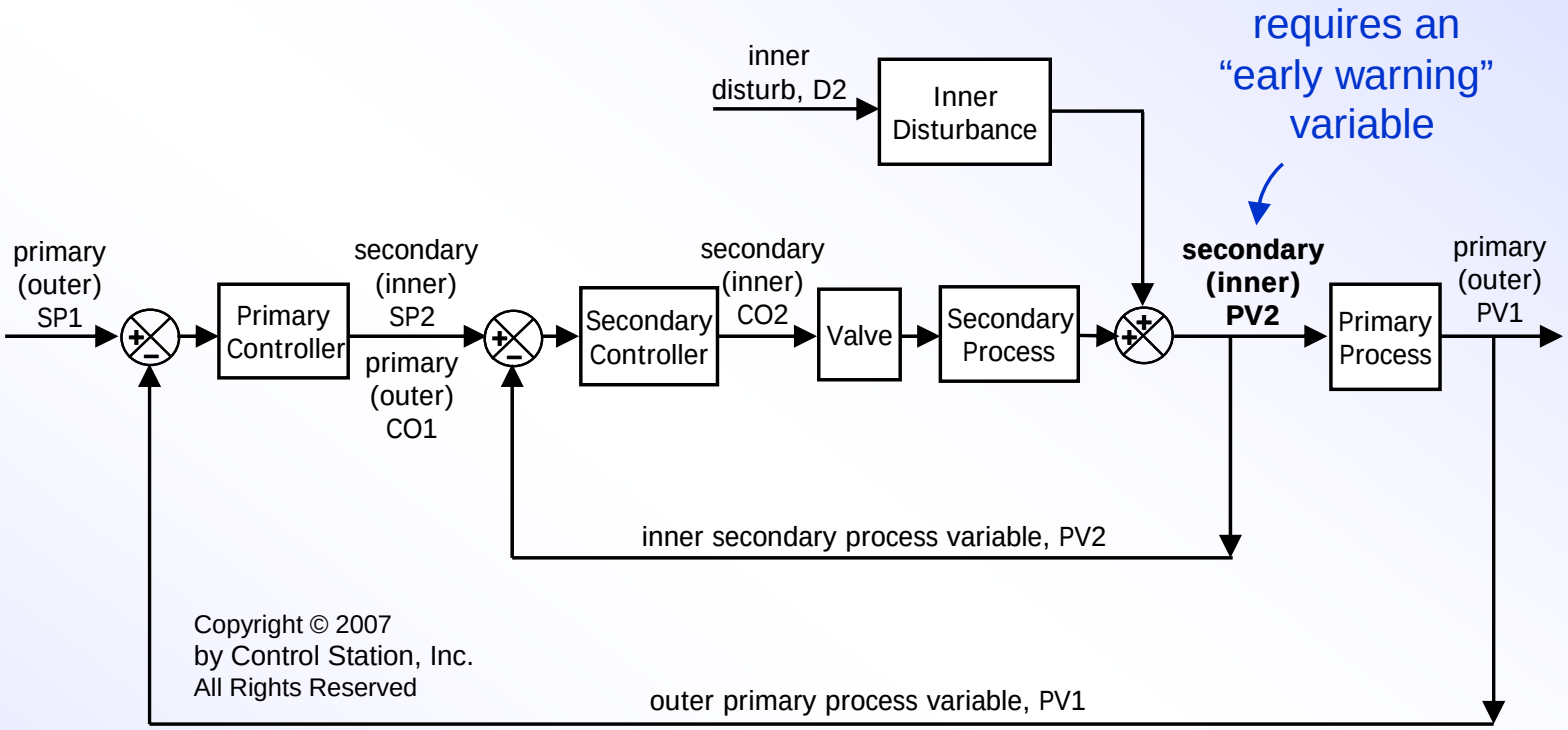
- A cascade is comprised of two ordinary PID controllers
- The inner secondary loop has a traditional feedback structure, and it is nested inside the outer primary loop

Nested Loops Work to Protect Outer Primary PV1



- Cascade architectures seek to improve the disturbance rejection performance of PV1

Early Warning is Basis for Cascade Success



- Success in a cascade design depends on the measurement and control of an "early warning" process variable PV2

Cascade Design

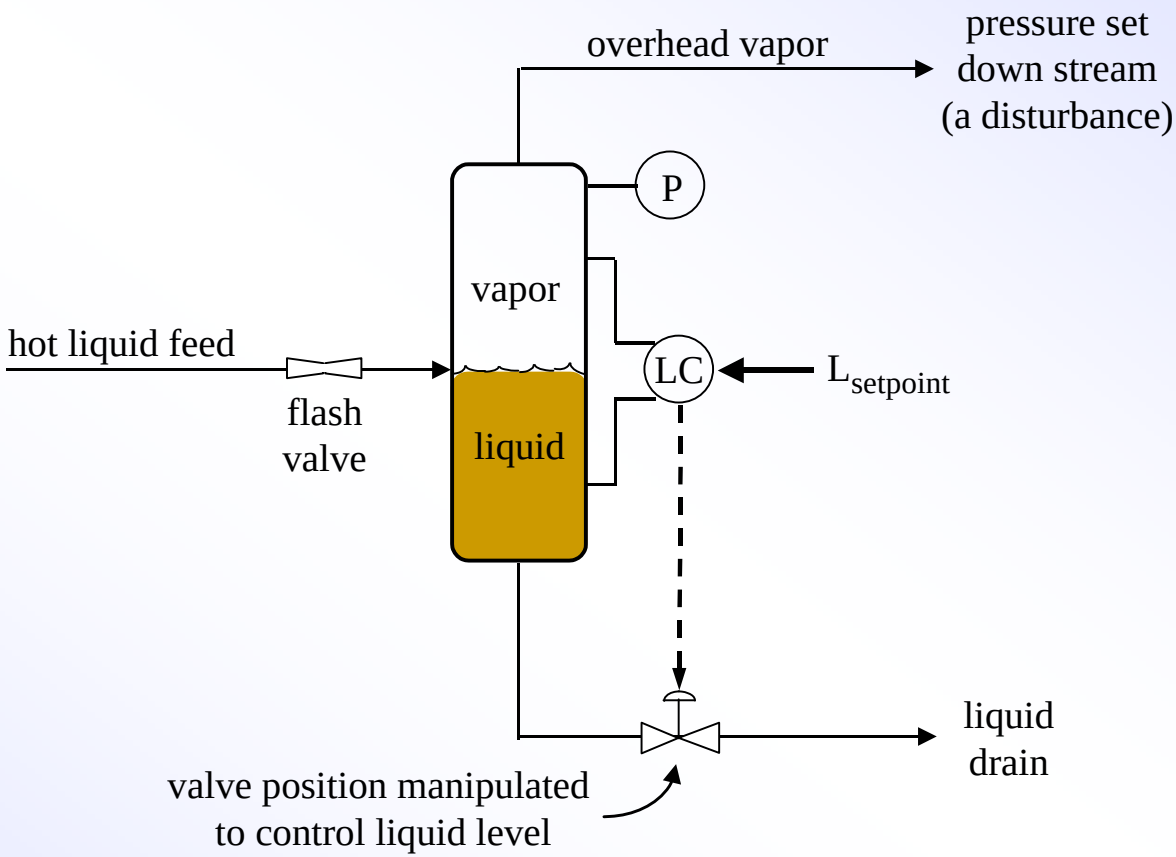
- Characteristics for selecting early warning PV2 include:
  - it must be measurable with a sensor
  - the same FCE (e.g., valve) used to manipulate PV1 also manipulates PV2
  - the same disturbances that are of concern for PV1 also disrupt PV2
  - PV2 responds before PV1 to disturbances of concern and to FCE manipulations

Cascade Design

- A cascade design requires:
  - two sensors
  - two controllers
  - one final control element (FCE)
- The output of the outer primary controller, rather than going to a valve, becomes the set point of the inner secondary controller
- Because of this nested architecture:

Success requires that
the settling time of the inner secondary inner loop
is significantly faster
than that of the outer primary outer loop

Example: Flash Drum Process



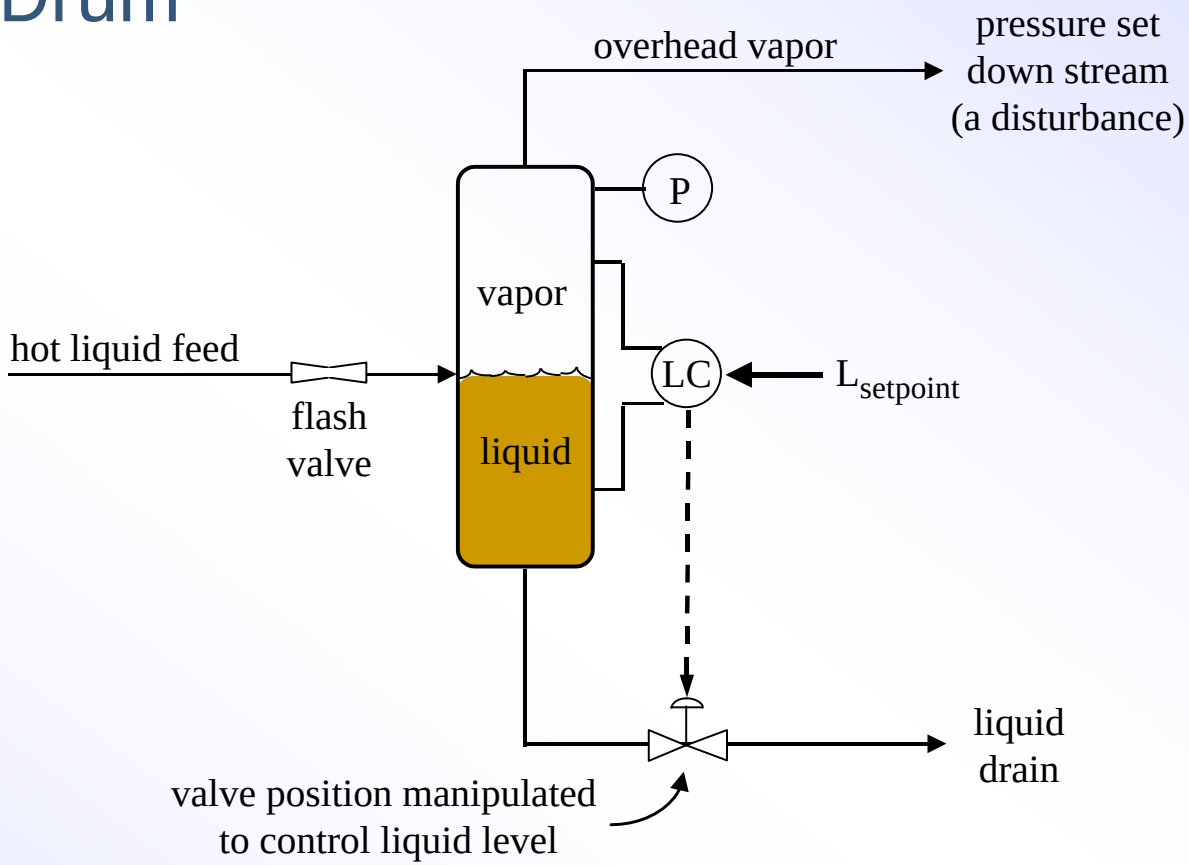
- Level must never fall so low that vapor is sent down liquid drain nor rise so high that liquid enters the vapor line

control station

Flash Drum – Single Loop Architecture

- Design Objective → control liquid level in the drum
- Choose valve position as manipulated variable
  - If level too high, open valve
  - If level too low, close valve
- Concern is that drain flow rate changes as a function of
  - valve position
  - hydrostatic head (height of the liquid)
  - pressure of vapor pushing down on liquid (a disturbance)

Flash Drum



- If pressure of vapor phase is constant, then as drain valve opens and closes, the liquid drain flow rate increases and decreases in predictable fashion
- Single loop architecture would then be satisfactory

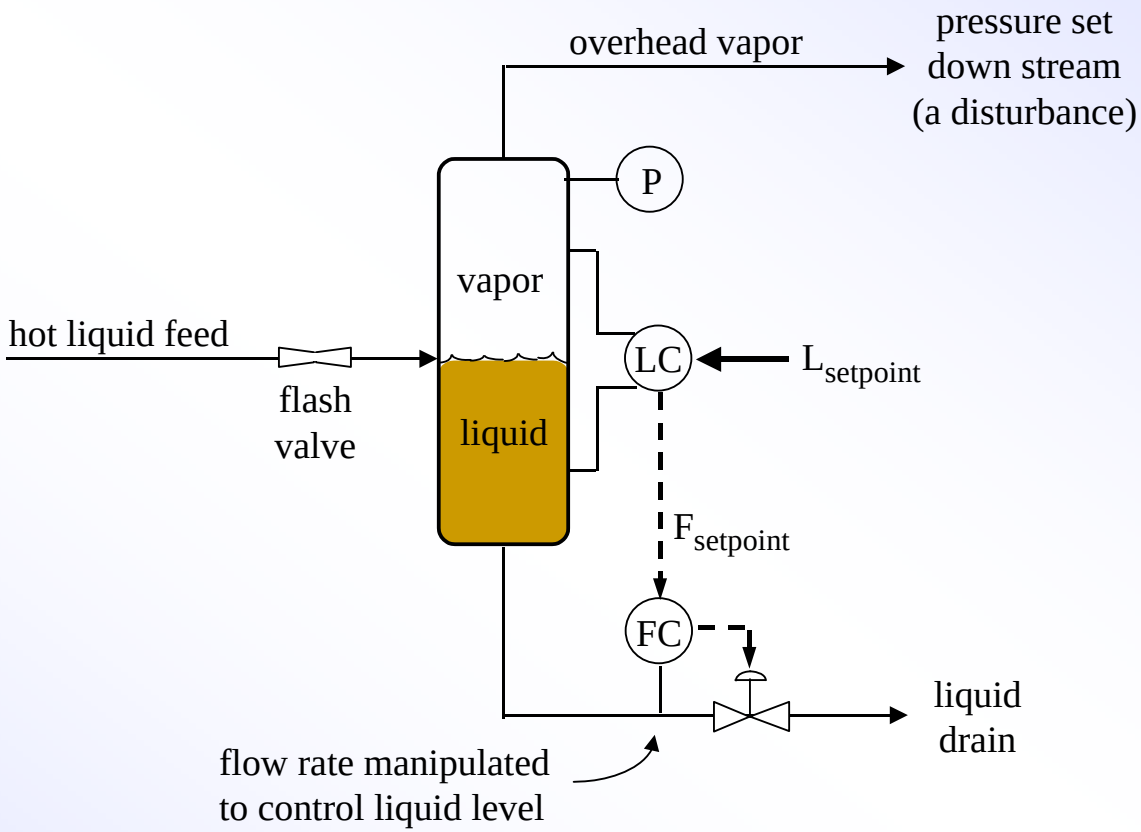
Flash Drum – Single Loop Architecture

- Suppose the vapor phase pressure starts decreasing:
  - This disturbance causes pressure pushing down on the liquid interface to decrease
  - If the valve position were to remain constant, the liquid drain flow rate would similarly decrease
  - Consider that if a pressure decrease occurs quickly enough, the controller can be opening the valve yet the liquid drain flow rate can continue to decrease

  This contradictory outcome can confound the controller

- Observation → It is liquid drain flow rate, not valve position, that must be adjusted for high performance disturbance rejection

Solution: Flash Drum Cascade Architecture

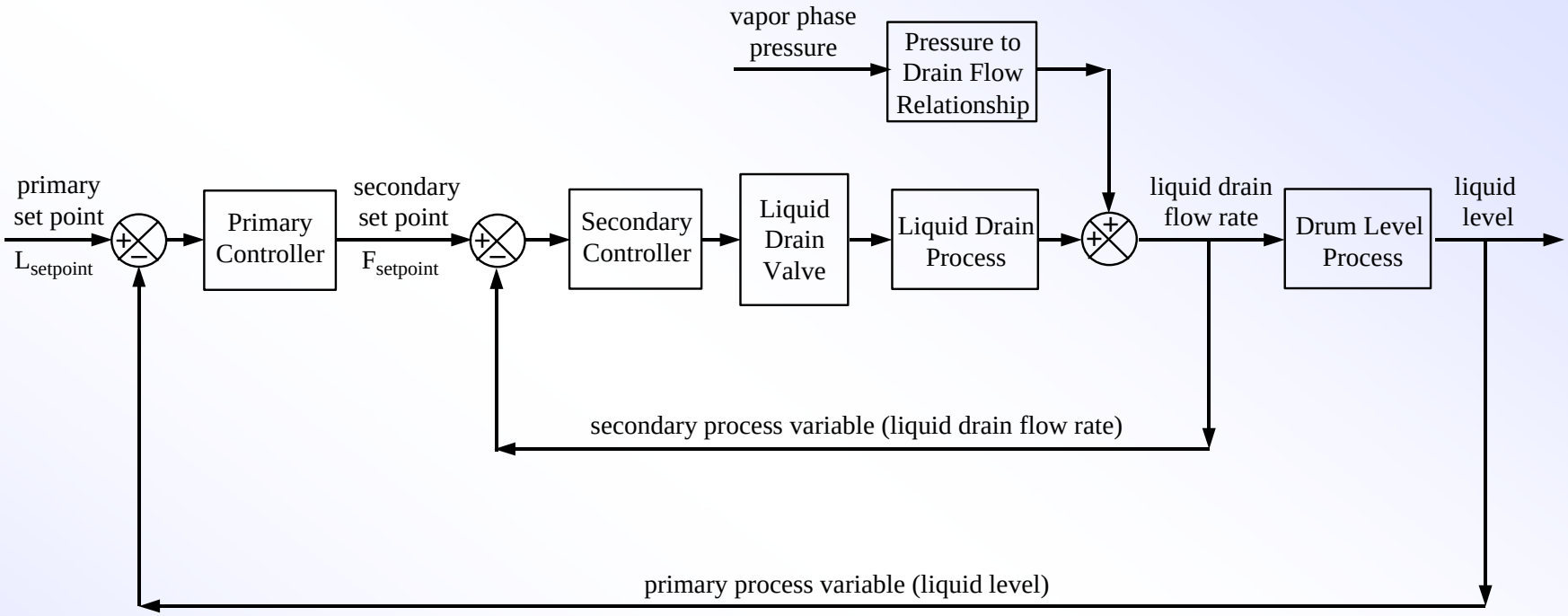


- Two controllers (level control; drain flow rate control)
- Two sensors (measuring liquid level; liquid drain flow rate)
- One final control element (valve in the liquid drain stream)

A Cascade Control Solution

- Liquid level is the outer primary PV1 and controlling it remains the main objective
- For inner secondary PV2 choose liquid drain flow rate:
  - liquid drain flow rate is measurable with a sensor
  - the same valve used to manipulate liquid level (PV1) also manipulates the liquid drain flow rate (PV2)
  - changes in vapor phase pressure that disturb PV1 also impact PV2
  - drain flow rate is *inside* the liquid level in that it responds well before liquid level to changes in valve position and changes in vapor phase pressure

Flash Drum Cascade Architecture



- Liquid level control (main objective) is outer primary loop
- Liquid drain flow rate is inner secondary loop
- Output of primary controller is set point of secondary controller
- Flow control dynamics are much faster than level control dynamics so this is consistent with design criteria

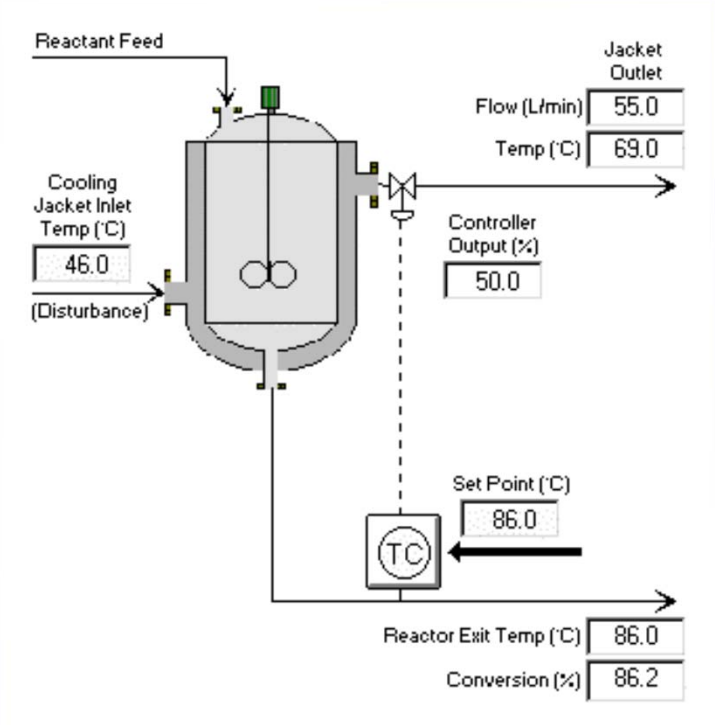
Flash Drum Cascade Architecture

- If liquid level is too high, the primary level controller now calls for an increased liquid drain flow rate rather than simply an increase in valve opening
- The flow controller then decides whether this means opening or closing the valve and by how much
- Thus, a vapor phase pressure disturbance gets addressed quickly by the secondary flow controller and this improves disturbance rejection performance

Tuning a Cascade Implementation

- Cascade loop tuning uses our existing skills:
  - Begin with both controllers in manual mode
  - Select P-Only controller for the inner secondary loop (integral action increases settling time and offset is rarely an issue for the secondary process variable)
  - Tune the secondary P-Only controller for set point tracking and test it to ensure satisfactory performance
  - Leave secondary controller in automatic; it is now part of the primary process. Select a PI or PID controller for the primary loop, tune it for disturbance rejection and test it
  - With both controllers in automatic, tuning is complete

Exploring the Jacketed Reactor Process



- Well mixed vessel with exothermic (heat producing) reaction
- Residence time is constant so conversion of feed to product can be inferred from the reactor exit stream temperature
- Objective → maintain constant measured reactor exit stream temperature in spite of jacket inlet temperature disturbances

control station

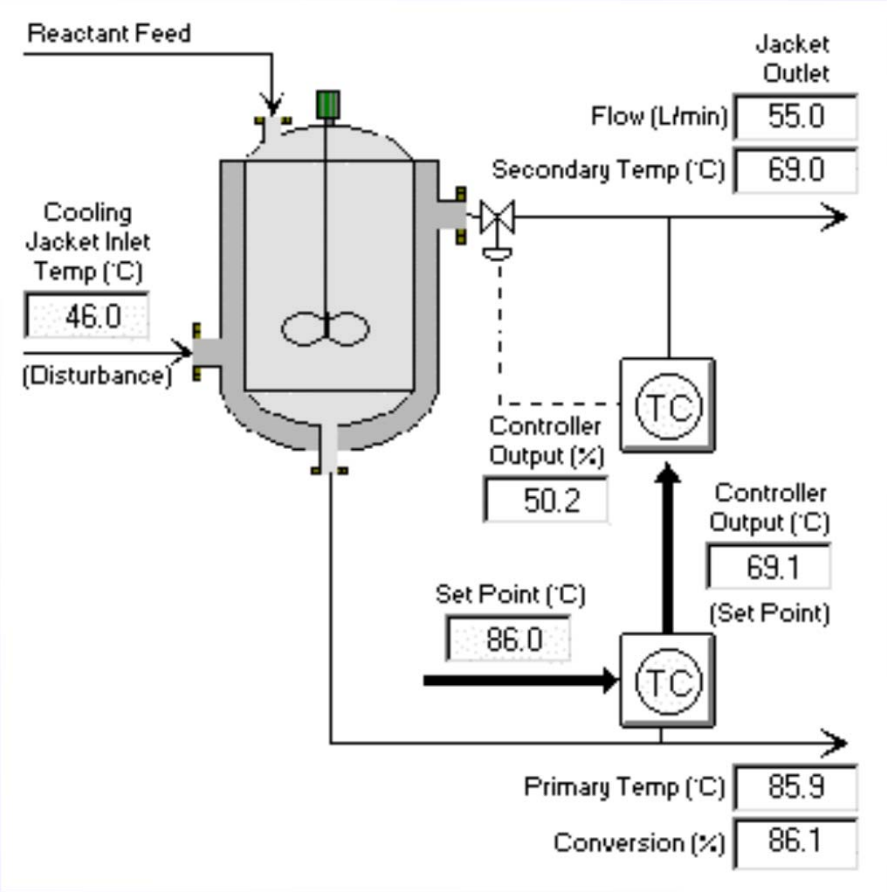
The Jacketed Reactor

- To control reactor exit stream temperature, the vessel is enclosed with a cooling jacket
- If the exit stream temperature (and thus conversion) is high, the controller opens a valve to increase cooling liquid flow rate
- This cools the reactor, slowing the heat producing reaction
- The disturbance variable of concern is the cooling jacket inlet temperature

Disturbances and the Jacketed Reactor

- Consider scenario where the temperature of the cooling liquid entering the jacket fluctuates, changing the ability of the cooling jacket to remove heat
- If the cooling liquid temperature becomes colder just as the reactor temperature starts to fall, the controller can lower the cooling liquid flow rate yet be removing more heat than before
- Again, a contradictory result can confound the controller and impact disturbance rejection performance

Cascade Architecture for the Jacketed Reactor



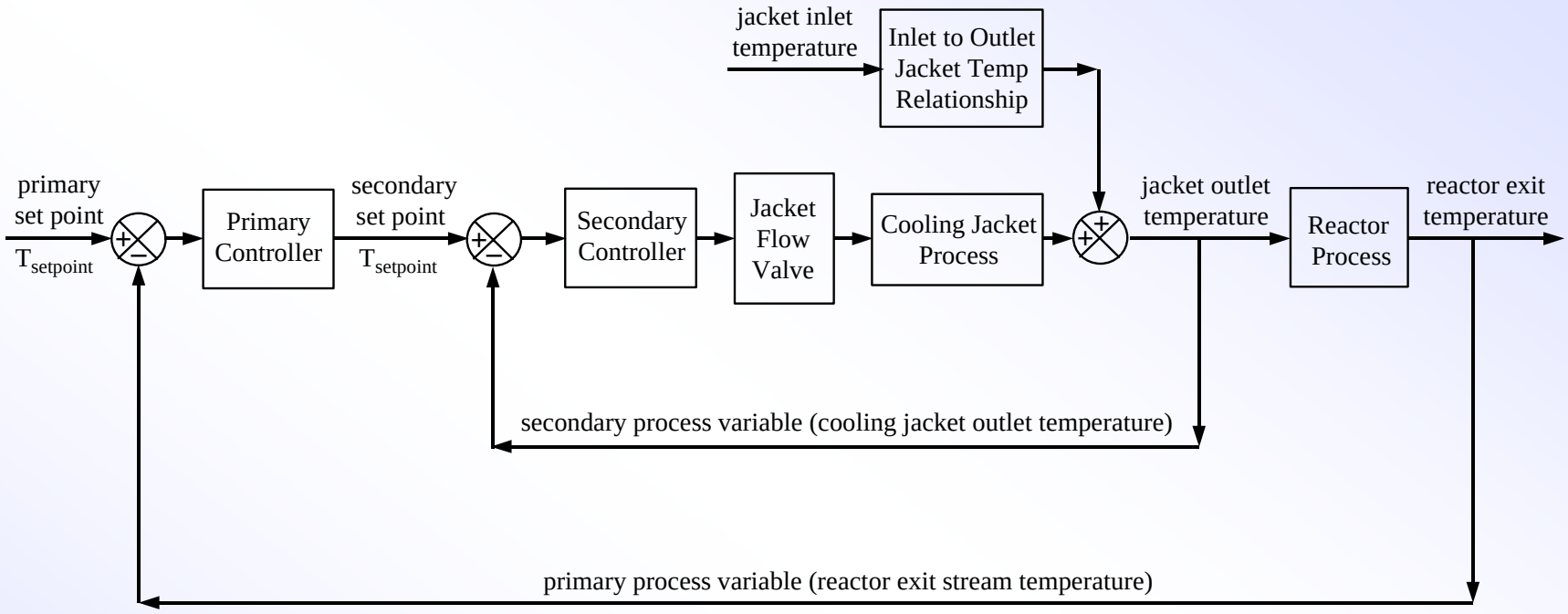
- Outer primary variable remains reactor exit stream temperature
- Inner secondary variable is cooling jacket outlet temperature

control station

The Reactor Cascade Architecture

- Cooling jacket outlet temp is a proper secondary variable
  - it is measurable with a sensor
  - valve used to manipulate reactor exit stream temperature (PV1) also manipulates cooling jacket outlet temp (PV2)
  - changes in cooling jacket inlet temperature that disturb reactor exit stream temp also disturb the cooling jacket outlet temp
  - the cooling jacket outlet temp is *inside* the reactor exit temp in that it responds first to changes in valve position and changes in the cooling jacket inlet temperature

The Reactor Cascade Architecture

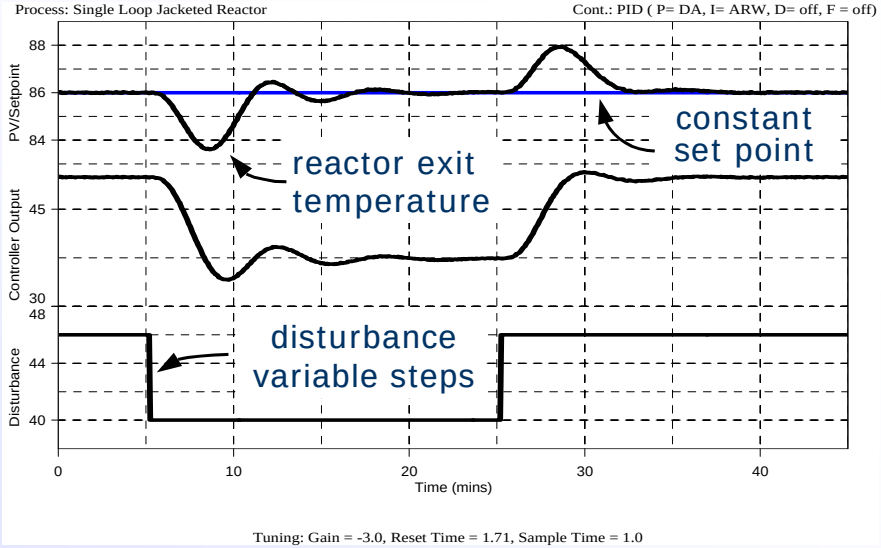


- Outer primary process (PV1) is reactor exit temperature
  - measured variable is reactor exit stream temperature
  - controller output is set point of secondary controller
- Inner secondary process (PV2) is the cooling jacket
  - measured variable is the cooling jacket outlet temperature
  - manipulated variable is the cooling jacket liquid flow rate

control station

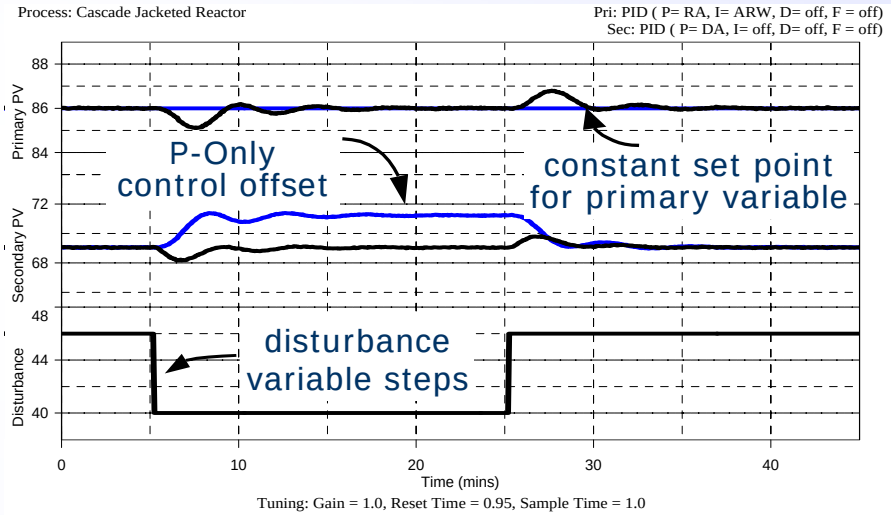
Disturbance Rejection Comparison

Disturbance Rejection Performance of Single Loop PI Controller



Tuning: Gain = -3.0, Reset Time = 1.71, Sample Time = 1.0

Disturbance Rejection Performance of Cascade Architecture



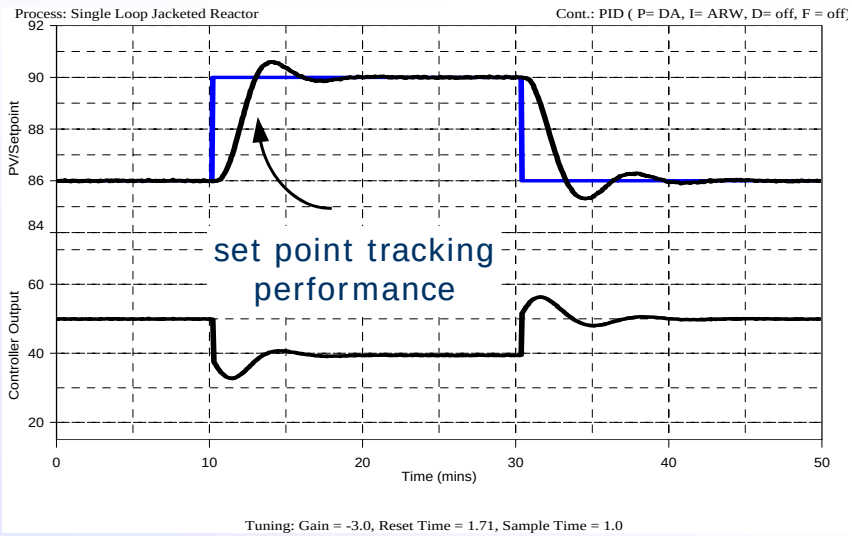
Tuning: Gain = 1.0, Reset Time = 0.95, Sample Time = 1.0

Tuning: Gain = -5.8, Sample Time = 1.0

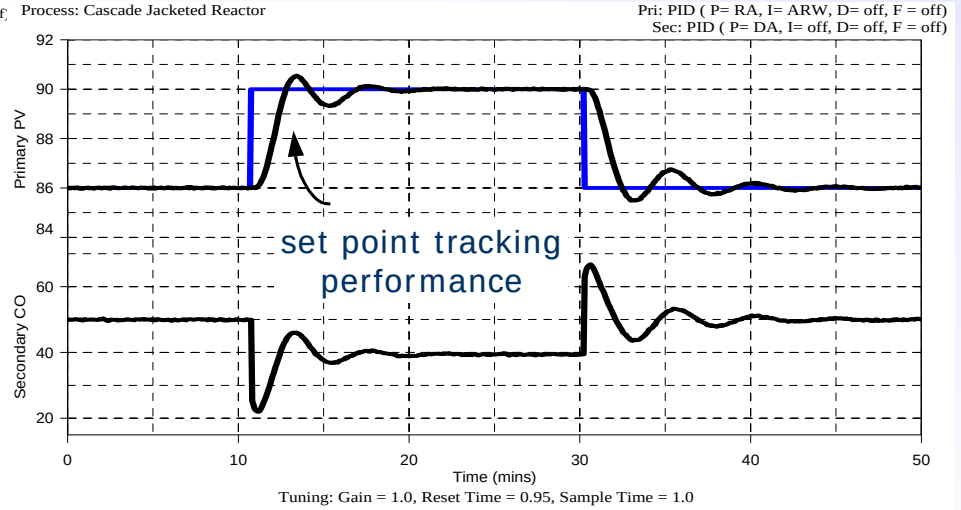
control station

Set Point Tracking Comparison

Set Point Tracking Performance Under PI Control



Set Point Tracking Performance Under Cascade Control



- Cascade does not provide benefit in tracking set point changes