

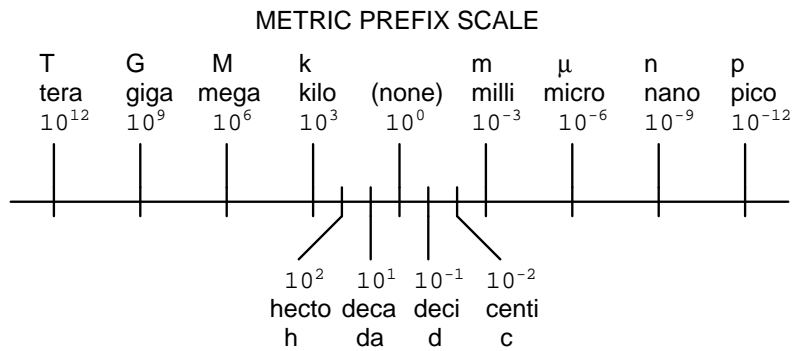
PID loop tuning

This worksheet and all related files are licensed under the Creative Commons Attribution License, version 1.0. To view a copy of this license, visit <http://creativecommons.org/licenses/by/1.0/>, or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA. The terms and conditions of this license allow for free copying, distribution, and/or modification of all licensed works by the general public.

Metric prefixes and conversion constants

- **Metric prefixes**

- Yotta = 10^{24} Symbol: Y
- Zeta = 10^{21} Symbol: Z
- Exa = 10^{18} Symbol: E
- Peta = 10^{15} Symbol: P
- Tera = 10^{12} Symbol: T
- Giga = 10^9 Symbol: G
- Mega = 10^6 Symbol: M
- Kilo = 10^3 Symbol: k
- Hecto = 10^2 Symbol: h
- Deca = 10^1 Symbol: da
- Deci = 10^{-1} Symbol: d
- Centi = 10^{-2} Symbol: c
- Milli = 10^{-3} Symbol: m
- Micro = 10^{-6} Symbol: μ
- Nano = 10^{-9} Symbol: n
- Pico = 10^{-12} Symbol: p
- Femto = 10^{-15} Symbol: f
- Atto = 10^{-18} Symbol: a
- Zepto = 10^{-21} Symbol: z
- Yocto = 10^{-24} Symbol: y



- **Conversion formulae for temperature**

- $^{\circ}\text{F} = (^{\circ}\text{C})(9/5) + 32$
- $^{\circ}\text{C} = (^{\circ}\text{F} - 32)(5/9)$
- $^{\circ}\text{R} = ^{\circ}\text{F} + 459.67$
- $\text{K} = ^{\circ}\text{C} + 273.15$

Conversion equivalencies for distance

- 1 inch (in) = 2.540000 centimeter (cm)
- 1 foot (ft) = 12 inches (in)
- 1 yard (yd) = 3 feet (ft)
- 1 mile (mi) = 5280 feet (ft)

Conversion equivalencies for volume

1 gallon (gal) = 231.0 cubic inches (in³) = 4 quarts (qt) = 8 pints (pt) = 128 fluid ounces (fl. oz.) = 3.7854 liters (l)

1 milliliter (ml) = 1 cubic centimeter (cm³)

Conversion equivalencies for velocity

1 mile per hour (mi/h) = 88 feet per minute (ft/m) = 1.46667 feet per second (ft/s) = 1.60934 kilometer per hour (km/h) = 0.44704 meter per second (m/s) = 0.868976 knot (knot – international)

Conversion equivalencies for mass

1 pound (lbm) = 0.45359 kilogram (kg) = 0.031081 slugs

Conversion equivalencies for force

1 pound-force (lbf) = 4.44822 newton (N)

Conversion equivalencies for area

1 acre = 43560 square feet (ft²) = 4840 square yards (yd²) = 4046.86 square meters (m²)

Conversion equivalencies for common pressure units (either all gauge or all absolute)

1 pound per square inch (PSI) = 2.03602 inches of mercury (in. Hg) = 27.6799 inches of water (in. W.C.) = 6.894757 kilo-pascals (kPa) = 0.06894757 bar

1 bar = 100 kilo-pascals (kPa) = 14.504 pounds per square inch (PSI)

Conversion equivalencies for absolute pressure units (only)

1 atmosphere (Atm) = 14.7 pounds per square inch absolute (PSIA) = 101.325 kilo-pascals absolute (kPaA) = 1.01325 bar (bar) = 760 millimeters of mercury absolute (mmHgA) = 760 torr (torr)

Conversion equivalencies for energy or work

1 british thermal unit (Btu – “International Table”) = 251.996 calories (cal – “International Table”) = 1055.06 joules (J) = 1055.06 watt-seconds (W-s) = 0.293071 watt-hour (W-hr) = 1.05506 x 10¹⁰ ergs (erg) = 778.169 foot-pound-force (ft-lbf)

Conversion equivalencies for power

1 horsepower (hp – 550 ft-lbf/s) = 745.7 watts (W) = 2544.43 british thermal units per hour (Btu/hr) = 0.0760181 boiler horsepower (hp – boiler)

Acceleration of gravity (free fall), Earth standard

9.806650 meters per second per second (m/s²) = 32.1740 feet per second per second (ft/s²)

Physical constants

Speed of light in a vacuum (c) = 2.9979×10^8 meters per second (m/s) = 186,281 miles per second (mi/s)

Avogadro's number (N_A) = 6.022×10^{23} per mole (mol^{-1})

Electronic charge (e) = 1.602×10^{-19} Coulomb (C)

Boltzmann's constant (k) = 1.38×10^{-23} Joules per Kelvin (J/K)

Stefan-Boltzmann constant (σ) = 5.67×10^{-8} Watts per square meter-Kelvin⁴ ($\text{W}/\text{m}^2 \cdot \text{K}^4$)

Molar gas constant (R) = 8.314 Joules per mole-Kelvin (J/mol-K)

Properties of Water

Freezing point at sea level = $32^\circ\text{F} = 0^\circ\text{C}$

Boiling point at sea level = $212^\circ\text{F} = 100^\circ\text{C}$

Density of water at $4^\circ\text{C} = 1000 \text{ kg}/\text{m}^3 = 1 \text{ g}/\text{cm}^3 = 1 \text{ kg}/\text{liter} = 62.428 \text{ lb}/\text{ft}^3 = 1.94 \text{ slugs}/\text{ft}^3$

Specific heat of water at $14^\circ\text{C} = 1.00002 \text{ calories}/\text{g} \cdot ^\circ\text{C} = 1 \text{ BTU}/\text{lb} \cdot ^\circ\text{F} = 4.1869 \text{ Joules}/\text{g} \cdot ^\circ\text{C}$

Specific heat of ice $\approx 0.5 \text{ calories}/\text{g} \cdot ^\circ\text{C}$

Specific heat of steam $\approx 0.48 \text{ calories}/\text{g} \cdot ^\circ\text{C}$

Absolute viscosity of water at $20^\circ\text{C} = 1.0019 \text{ centipoise (cp)} = 0.0010019 \text{ Pascal-seconds (Pa}\cdot\text{s)}$

Surface tension of water (in contact with air) at $18^\circ\text{C} = 73.05 \text{ dynes}/\text{cm}$

pH of pure water at $25^\circ\text{C} = 7.0$ (*pH scale = 0 to 14*)

Properties of Dry Air at sea level

Density of dry air at 20°C and 760 torr = $1.204 \text{ mg}/\text{cm}^3 = 1.204 \text{ kg}/\text{m}^3 = 0.075 \text{ lb}/\text{ft}^3 = 0.00235 \text{ slugs}/\text{ft}^3$

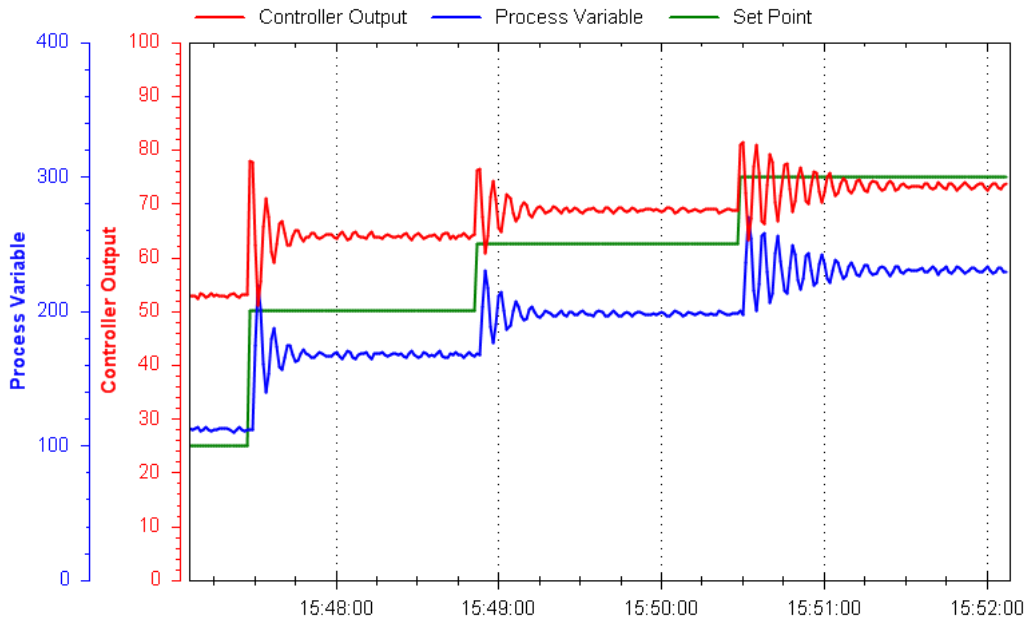
Absolute viscosity of dry air at 20°C and 760 torr = $0.018 \text{ centipoise (cp)} = 1.8 \times 10^{-5} \text{ Pascal-seconds (Pa}\cdot\text{s)}$

file conversion_constants

Questions

Question 1

Examine this process trend showing the PV, SP, and Output of a loop controller:



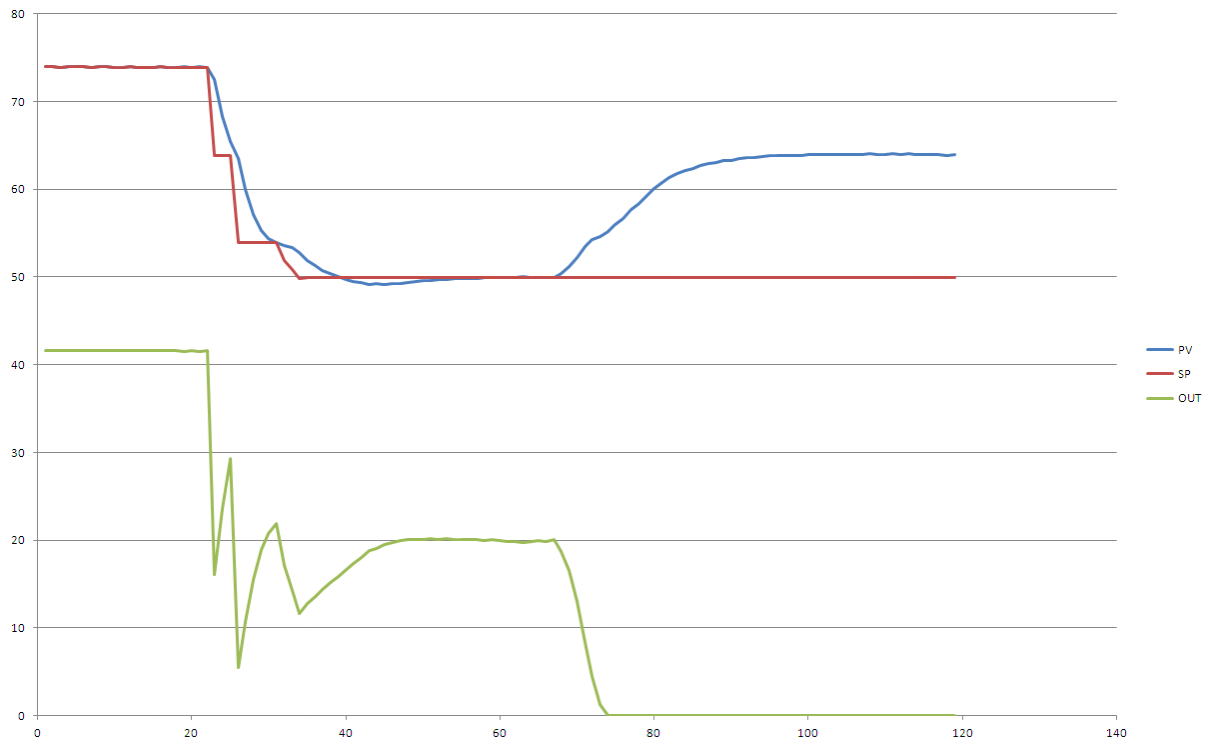
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i02632](#)

Question 2

Examine this process trend showing the PV, SP, and Output of a loop controller:



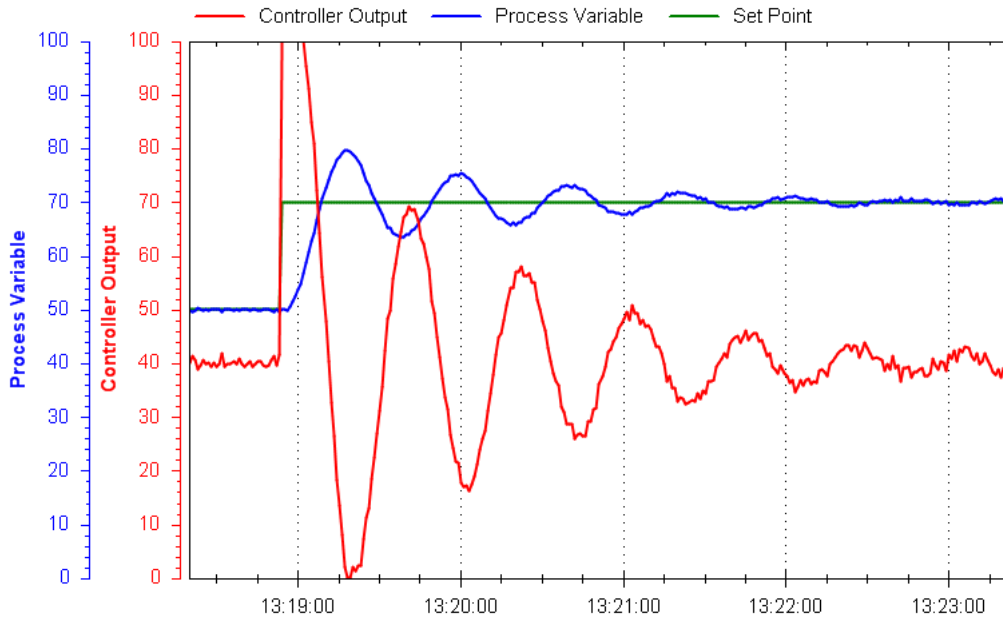
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i02866](#)

Question 3

Examine this process trend showing the PV, SP, and Output of a loop controller:



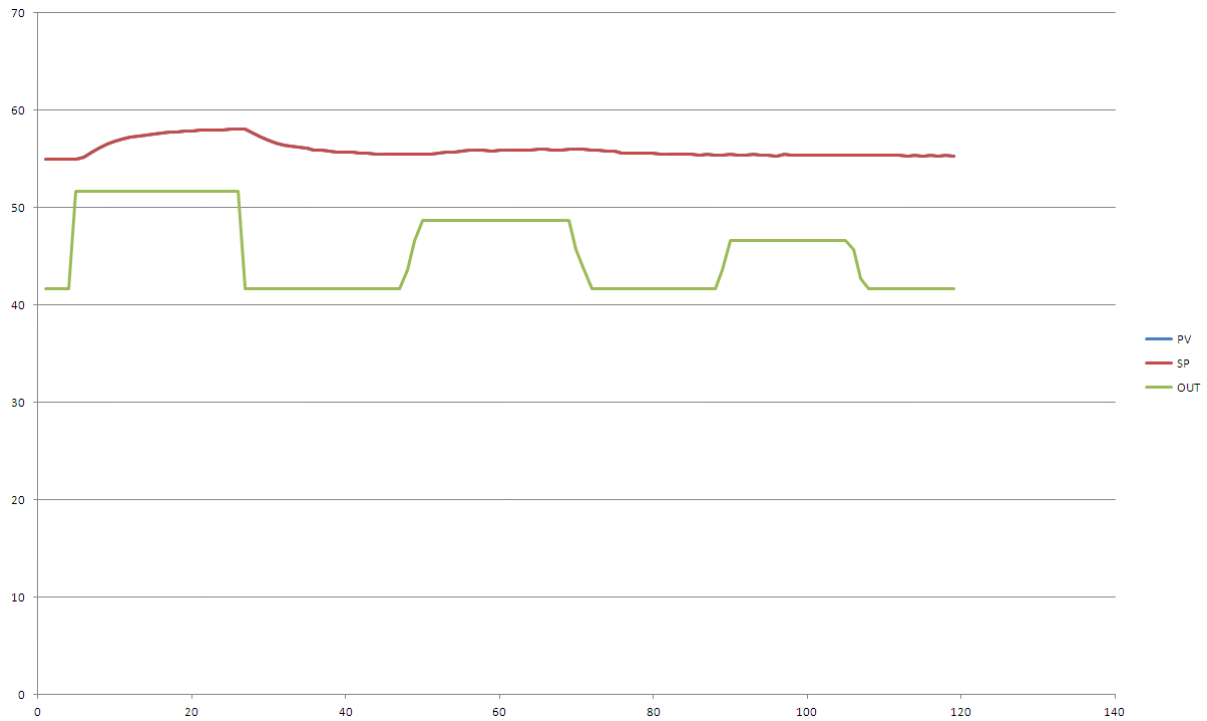
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02633

Question 4

Examine this process trend showing the PV, SP, and Output of a loop controller:



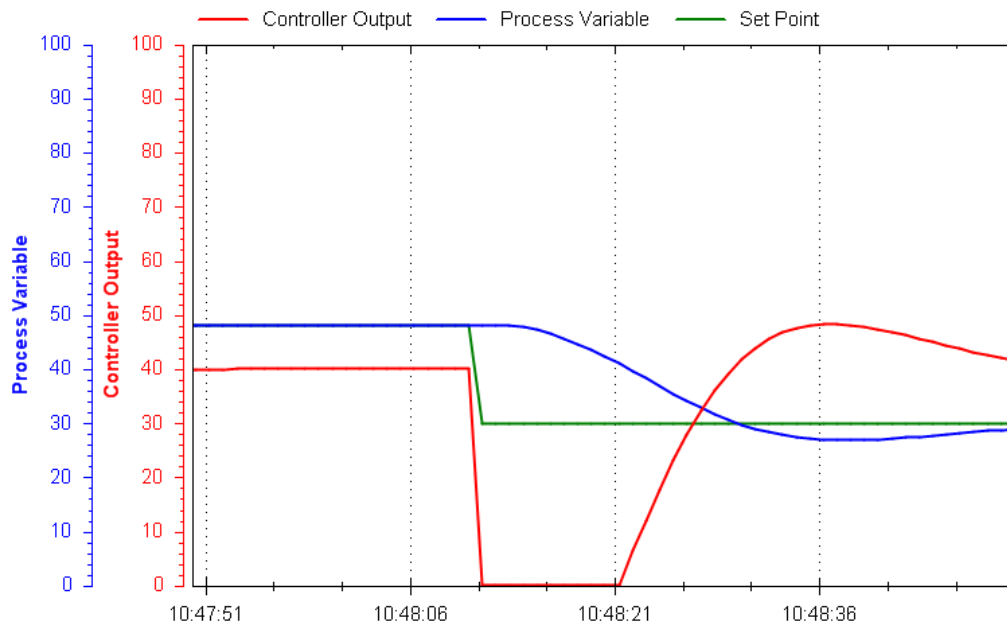
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i02571](#)

Question 5

Examine this process trend showing the PV, SP, and Output of a loop controller:



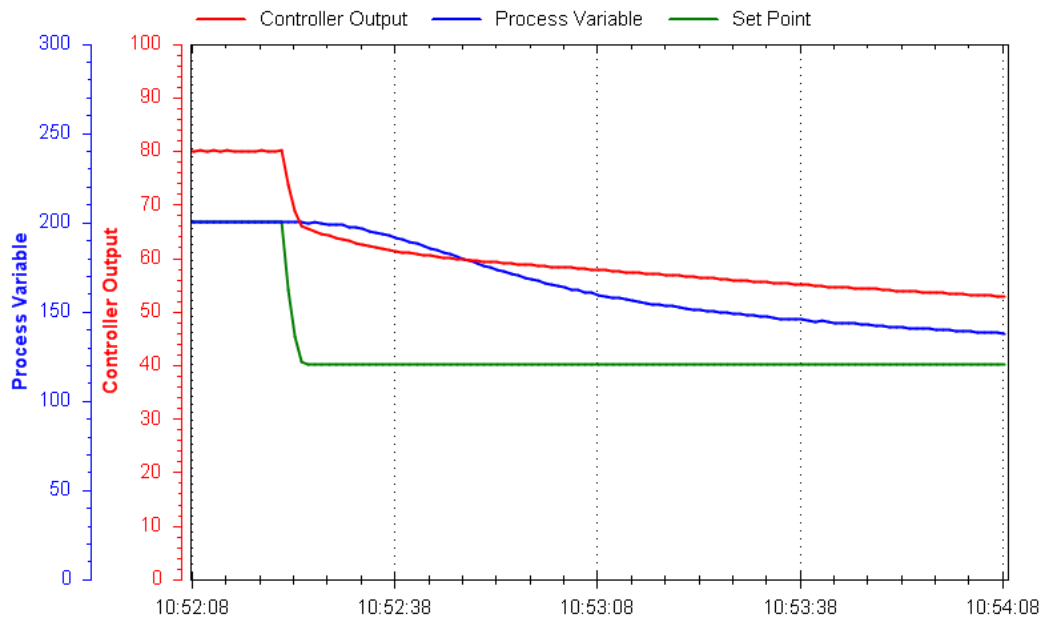
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02634

Question 6

Examine this process trend showing the PV, SP, and Output of a loop controller:



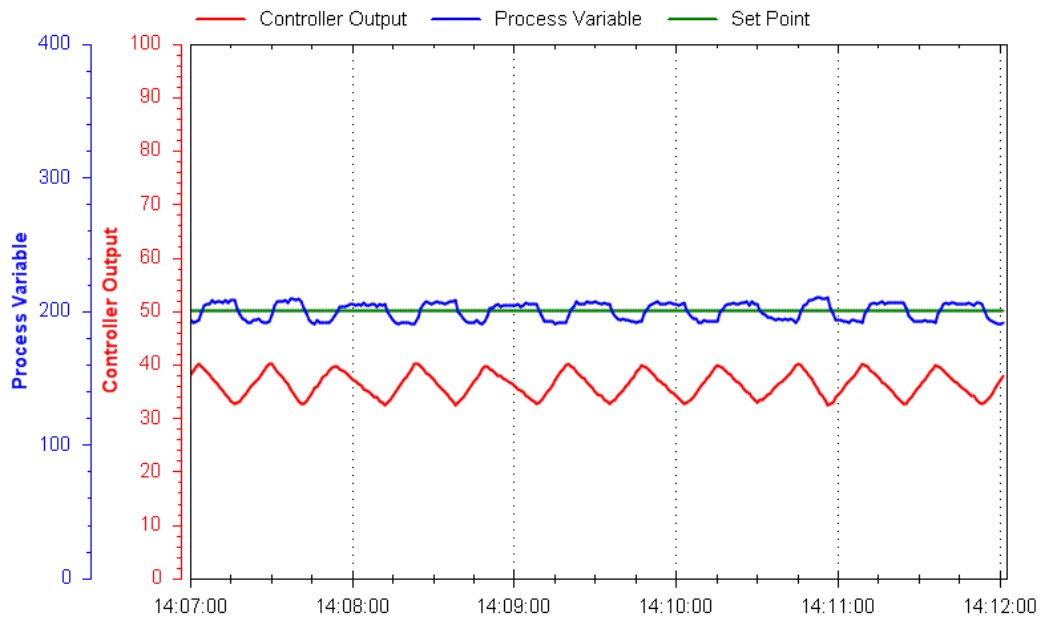
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02635

Question 7

Examine this process trend showing the PV, SP, and Output of a loop controller:



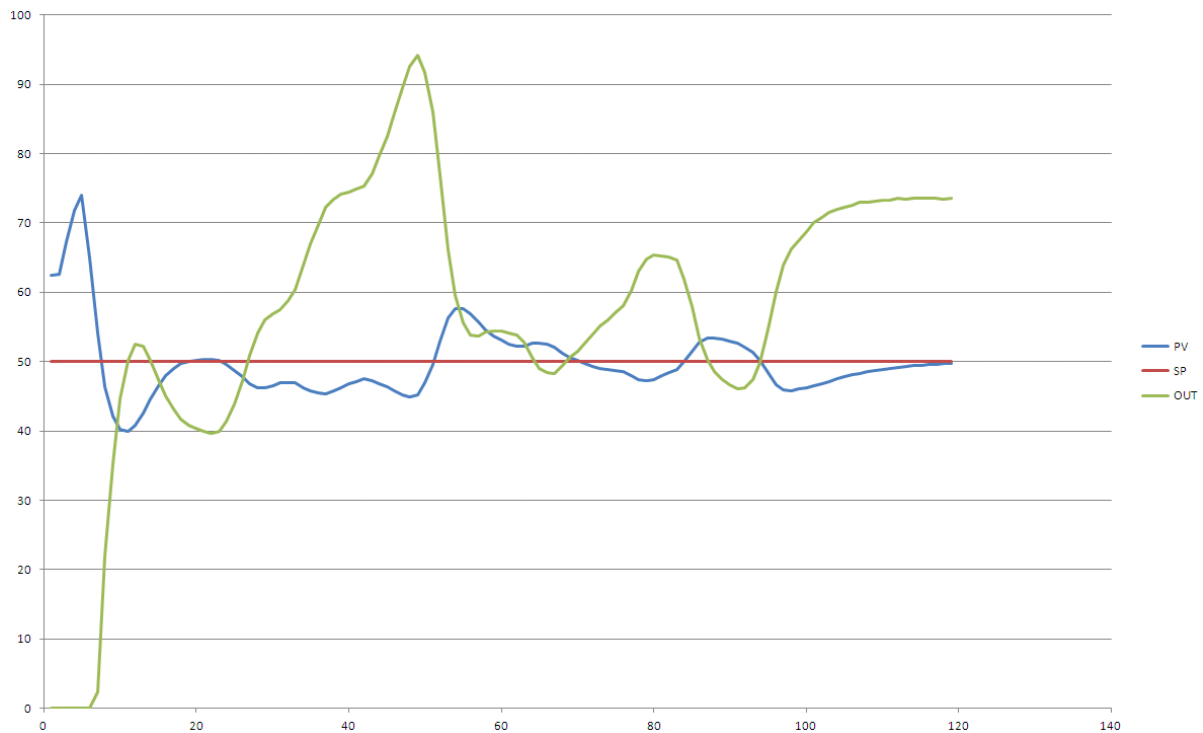
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i02636](#)

Question 8

Examine this process trend showing the PV, SP, and Output of a loop controller:



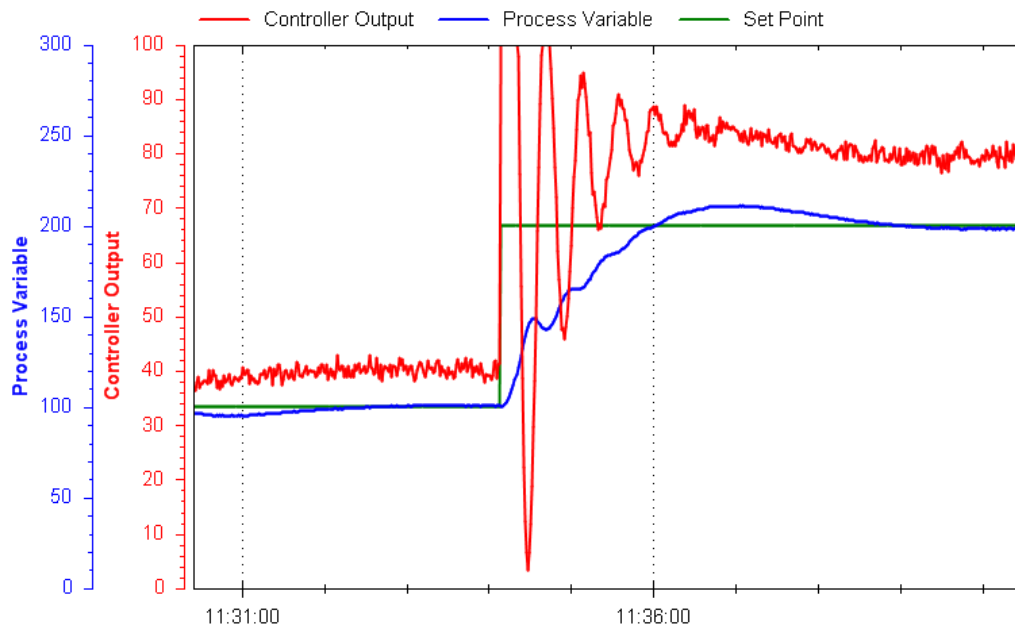
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02572

Question 9

Examine this process trend showing the PV, SP, and Output of a loop controller:



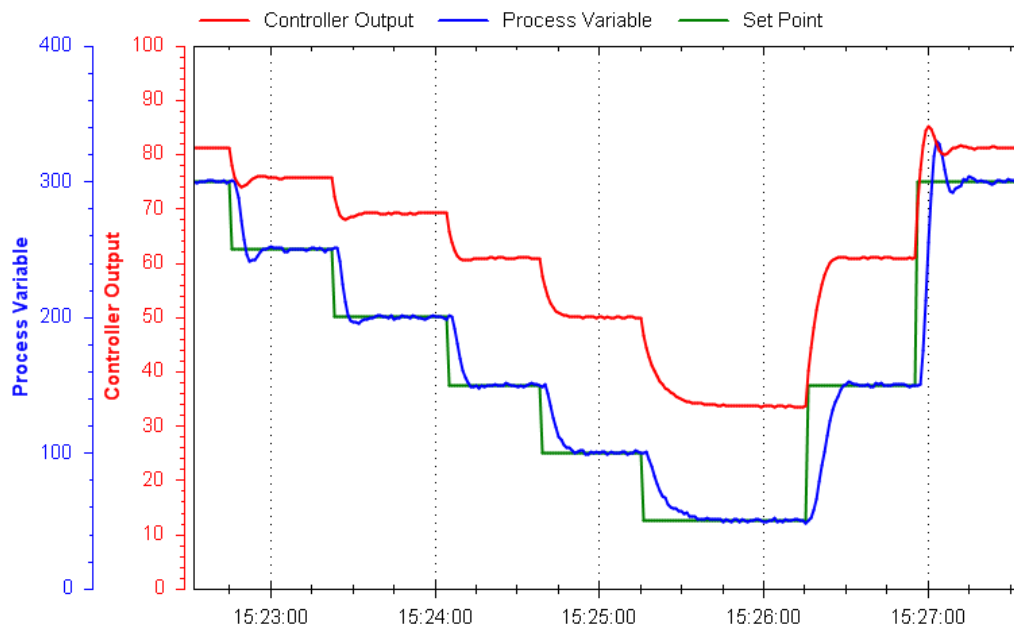
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02637

Question 10

Examine this process trend showing the PV, SP, and Output of a loop controller:



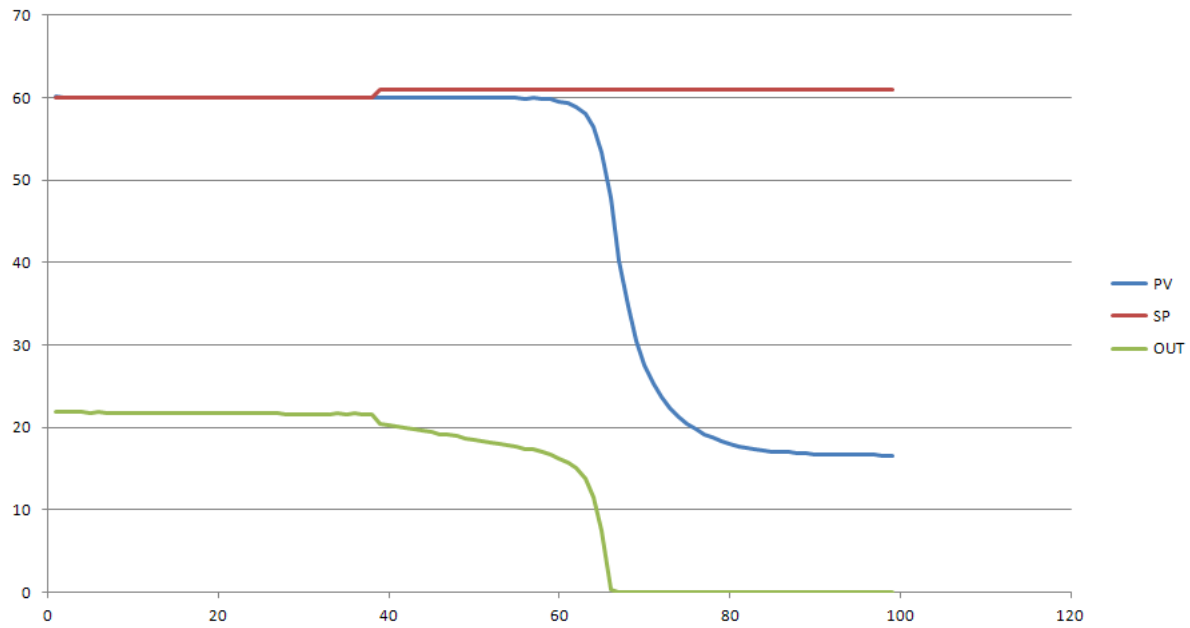
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02638

Question 11

Examine this process trend showing the PV, SP, and Output of a loop controller:



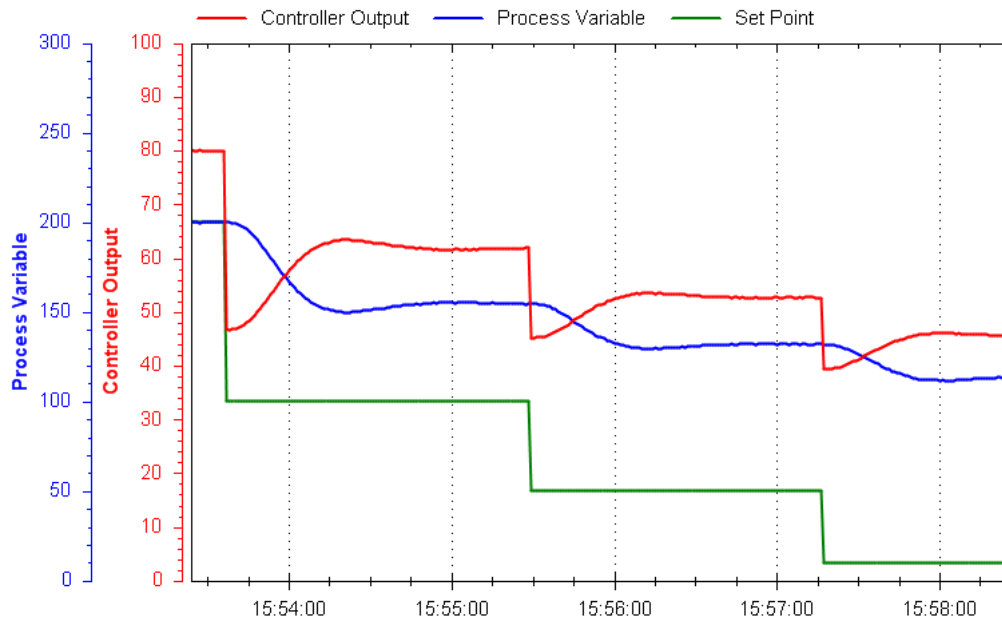
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i01280](#)

Question 12

Examine this process trend showing the PV, SP, and Output of a loop controller:



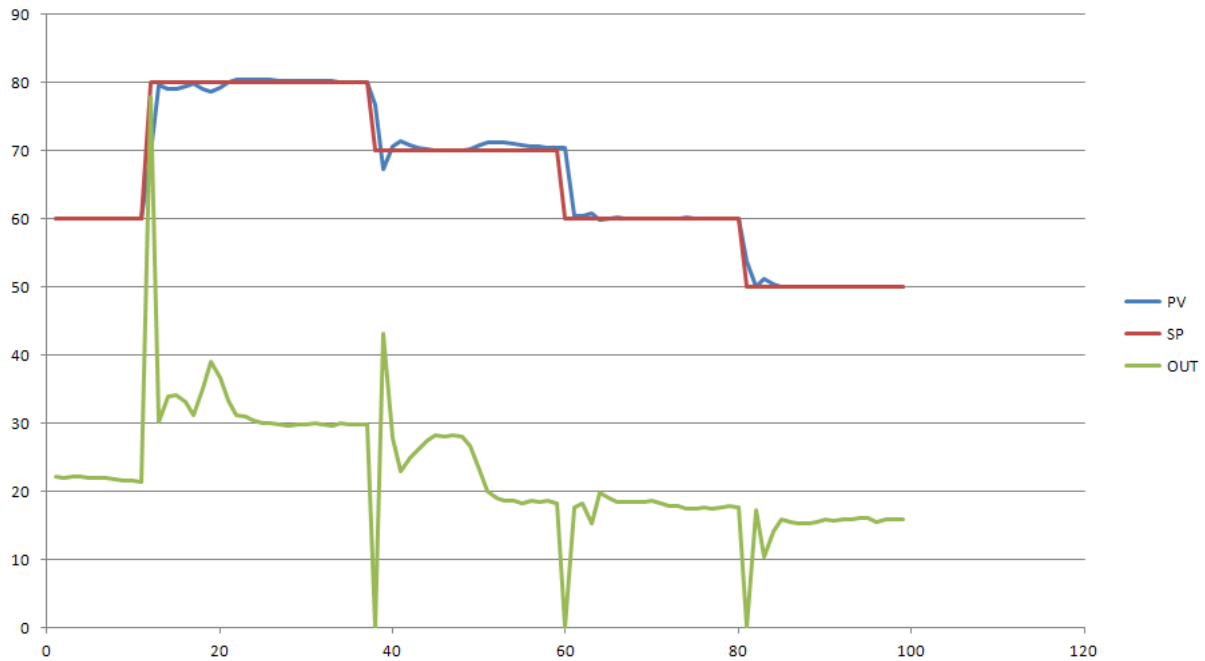
Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

file i02639

Question 13

Examine this process trend showing the PV, SP, and Output of a loop controller:



Based on what you see here, determine the following:

- Whether this is an open-loop or a closed-loop response
- Whether the controller is (or needs to be) *direct-acting* or *reverse-acting*
- If possible, identify any problems with the field instrumentation
- If possible, identify any problems with the controller PID tuning
- Qualitatively identify the kind of PID tuning we will need for robust control

[file i01047](#)

Answer 1

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, but it appears as though the process is very quick to respond with no discernable dead time or other lags.

The controller tuning is clearly inappropriate for this process. Note the large offset between PV and SP (i.e. how the process variable never settles at the setpoint value, even though it's clearly a fast-responding process). This tells us the controller is configured only for proportional action, and this process needs integral! We can also tell this from the 180° phase shift between PV and output during the oscillations: this is the classic response of a reverse-acting proportional-only controller with excessive gain.

Aggressive integral action with a minimum of proportional gain should work very well in this process, which is probably a liquid flow-control process.

Answer 2

This is a **closed-loop** (automatic-mode) response, seeing how the output reacts to the changes in setpoint and process variable.

The control action here is definitely **reverse**. We can tell this by examining the output's immediate down-step as the setpoint is decreased, as well as the output's decrease as the process variable rises later on.

The control system's initial response to the large setpoint changes looks very good. No problems, either with the field instruments or the controller itself, are evident from this portion of the trend recording. However, something strange happens around the "75" time index, where the process variable begins to rise and the controller output falls to zero in response. It would appear that something has happened to cause the process variable to rise uncontrollably, and the controller is helpless to correct the situation.

We may speculate as to the cause of this uncontrollable situation, but there is not enough information to make a definitive judgment. One possible cause is a large load change in the process that overwhelms the controller. Another possible cause is a final control element failure, going to the 100% condition as if the controller signal had increased to saturation.

Answer 3

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, but it appears as though the process is very quick to respond with no discernable dead time or other lags.

The controller tuning is too heavy on proportional action. We can tell this from the phase shift between PV and output during the oscillations, which is nearly 180° . Excessive integral action would shift the phase of the output wave further to the right (i.e. so that each peak of the output waveform coincided with the zero-crossing of the PV waveform, or very nearly). The fact that the inverse peaks of the PV and output waves are very nearly aligned tells us that excessive gain (proportional action) is the culprit here. Another clue is the magnification of noise we see in the output trend compared to the PV trend – only proportional action or derivative action can cause this, and since we see no sign of excessive derivative action (e.g. output wave leading the PV wave), we can safely say the problem is too much gain.

The process response time (dead time, lag time) seems to be very short, which is a good thing for process control. We can tell, however, that this is an *integrating* process by the way it was able to achieve a new SP value with the old output value. This means it will exhibit some overshoot with SP changes if there is any integral action. We may have to do most of the control through proportional action (albeit much less gain than we are using now!), with just enough integral action to handle load changes.

Answer 4

This is an **open-loop** (manual-mode) response, seeing how the output is being moved up and down in step-change fashion which suggests a human operator making changes in manual. The process variable, by comparison, is responding in a smooth-changing manner to those output step-changes.

The process variable is increasing as the output signal is made to increase. This tells us the control action in automatic mode will need to be **reverse**. Put in other terms, if we know the process variable is driven by the output signal in direct fashion (i.e. increasing the output makes the process variable increase, and vice-versa), then we know the controller is going to have to do the opposite to regulate this process. For example, if the process variable happens to increase due to a load change, the controller's output will need to decrease in order to drive that process variable back down where it belongs.

We can tell there is definitely a significant amount of hysteresis in the final control element because the process variable fails to respond to the last (smallest) step-change in controller output. This, incidentally, is a good way to test for hysteresis: step the output up and down in progressively smaller increments until no response is seen in the process variable. It is at that point you may interpret that last output step-change as the amount of hysteresis in the final control element. In this case, that quantity appears to be about 5%.

Since this is an open-loop test we cannot tell how well the controller is tuned. However, we can discern from the response of the process variable to output step-changes that this is a **self-regulating** process with little dead time and first-order lag characteristics, which means it should respond well to fairly aggressive proportional action and will also require integral action to avoid offset.

Answer 5

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

This process does exhibit some dead time as well as lag time, which explains the setpoint overshoot. A field check of the control element (valve) might be good to do, so see that it is not sticking and causing dead time.

The controller tuning actually looks pretty good here. The only problem is the slight overshoot of setpoint, which may or may not be significant depending on the specific process and the needs of operations personnel. If this overshoot is deemed excessive, we might wish to turn down the proportional action (gain), based on the fact the PV and output waves seem to hit their respective peaks at nearly the same time (characteristic of proportional-dominant action).

Given the existence of dead time and lag time together, we must be careful not to use too much proportional action lest the loop oscillate. Derivative action could be very useful in taming the effects of lag time.

Answer 6

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, and it appears as though the process possesses a multiple-order lag, but the time scale of this lag seems modest. It might not be a bad idea to look around for sources of lag time (e.g. thermowell mass, improperly inserted temperature probe), just to see if the response time might be improved a bit. Another possibility explaining the lag time would be a transmitter (or controller input block) configured with too much filtering (damping), adding a single-order lag to whatever lag(s) the process itself already possesses.

The controller tuning is clearly inappropriate for this process, and should be much more aggressive than it is right now. Note how the PV is taking a long time to reach the new SP, and how the controller output is ramping down at a very leisurely pace.

This process appears to be self-regulating, and so we know we must have some integral action in the controller. The existence of multiple-order lag makes this loop a good candidate for moderate derivative action.

Answer 7

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output ramps up whenever the PV is below SP, and the output ramps down whenever the PV is above SP.

This loop definitely has hysteresis in the final control element (e.g. sticky valve), because this trend is a classic *slip-stick cycle* in a self-regulating process: the PV exhibits a square-wave shape while the output ramps up and down like a sawtooth wave.

There probably isn't anything wrong with the controller's tuning, and no tuning adjustments will fundamentally address the problem of valve stiction.

The process appears to be self-regulating with a fast response time (note how quickly the PV settles at a new value following each "slip" of the control valve), which means it should control very well with aggressive integral action. However, final control element hysteresis is the bane of integral action, as it causes repeated reset windup as we see here.

Answer 8

This is a **closed-loop** (automatic-mode) response based on the smoothly-curved shape of the output trend in response to the process variable changing over time.

This controller is **reverse-acting**, as evidenced by the output trend's inverted shape in comparison to the process variable's trend.

No problems are evident in the field instrumentation, although it appears this process is subjected to large load changes. Note how far the output of the controller must move in order to hold the process variable relatively close to setpoint.

With all the severe load changes happening in this trend, it's difficult to tell whether any instability is being caused by the controller (i.e. overtuning) or if the load is simply that unstable. Several small oscillations do appear between the large surges, which may indicate an overly-aggressive controller response. Based on the phase shift between process variable and output trends being very nearly 180°, it would appear that the dominant action in this reverse-acting controller is **proportional**. Perhaps the gain needs to be diminished. However, decreasing the gain too much will compromise this controller's ability to quickly respond to severe load changes, which this process seems to suffer from.

A recommended tuning strategy would be to try some derivative action to manage load changes, while decreasing proportional action a bit. An even more important recommendation is to identify the source of the load swings, and look for ways to either minimize those variations or incorporate a different control strategy (e.g. cascade or feedforward) to better manage the variations.

Answer 9

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, but it appears as though the process is very quick to respond with little dead time or other lags.

The controller tuning is clearly too aggressive for this process. Note the “porpoising” action of the PV as it approaches SP following the SP step-change. Only two types of controller action can cause this to occur: *proportional*, or *derivative*. Porpoising is when an oscillation occurs in the PV prior to it crossing setpoint, which explains why integral action cannot ever be to blame for porpoising: the only way a loop oscillation can occur is when the final control element oscillates as well (i.e. changes direction), and since integral action will never change direction until PV crosses SP, oscillations that occur on one side of SP cannot be caused by integral action. Looking at the phase shift between PV and output during the oscillations, it appears the output peaks may slightly lead the PV peaks, but only slightly. This suggests that proportional is the action that is too aggressive (if it were derivative, there would be more of a leading phase shift).

This is definitely a self-regulating process, as revealed by the fact a new output value is required to achieve a new setpoint value. This means integral control action will definitely be necessary. Good control will require less gain and perhaps a bit more derivative action to help cancel the lag. Integral action looks just fine where it is right now, with just a little SP overshoot.

Answer 10

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

The only problem here is that the process exhibits a varying gain. This is why the control is over-sensitive (oscillatory) at high setpoint values and sluggish at low setpoint values. This could be a function of process dynamics, or of a control valve with the wrong characteristic (e.g. an equal-percentage valve in an application better suited for a linear valve).

The controller tuning looks really good when the process variable is maintaining around 40%. This mid-range is where the tuning seems to be optimized.

This process appears to be self-regulating with short dead and lag times. It is clear from examining the phase shifts of output versus PV that there is some derivative action at work here, since the output actually leads the PV when there is oscillation. The key will be linearizing the process gain so that one set of tuning parameters will work robustly across the control range.

Answer 11

This is a controller that has been configured for the wrong action! The configured action is *direct*, as evidenced both by the output's step down as the setpoint steps up, and also the output's downward fall as the process variable falls.

The process clearly demands a controller with *reverse* action in order to be regulated.

We can also see evidence of some hysteresis in the final control element. Note the small setpoint step-change made just prior to the "40" mark on the time axis, causing the controller's integral action to slowly wind in the downward direction. However, the process variable does not begin to move until somewhere around the "48" time index. This suggests a final control element that is unresponsive to the change in controller output during that span of 8 time units.

The fact that the process variable plummets simultaneously with the plummeting controller output around "53" on the time axis tells us that 8-unit period of unresponsiveness is not dead time due to transport delay or compounded process lags, but that it must be due to hysteresis in the final control element. If the 8-unit delay were due to an inherent dead-time response in the process itself, we would see the process variable's plummeting happening 8 time units after the output signal's downward dive.

Answer 12

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, but it appears as though the process is very quick to respond with no discernable dead time or other lags.

The controller tuning is clearly lacking integral action. Note the large offset between PV and SP (i.e. how the process variable never settles at the setpoint value). This tells us the controller is configured only for proportional action, and this process needs integral! We can also tell this from the 180° phase shift between PV and output during the oscillations: this is the classic response of a reverse-acting proportional-only controller. The actual amount of gain appears to be appropriate for the loop, since the oscillations are not excessive.

We desperately need to apply some integral action to this loop, because self-regulating loops absolutely need integral action to handle load changes and achieve new setpoint values.

This is a *closed-loop test*, based on the fact the output signal responds dynamically to the changing process variable, as well as to the step-change in setpoint.

This is a *reverse-acting* controller: the output steps up when the setpoint steps up (implying the output would step down if the process variable stepped up).

There do not appear to be any field instrumentation problems revealed in this trend. A manual-mode (open-loop) test would be more informative in that regard, but it appears as though the process is very quick to respond with no discernible dead time or other lags.

The controller tuning is too aggressive, as evidenced by the process variable's oscillations following every setpoint change. Furthermore, we can tell this oscillation is a form of "porpoising" because it occurs prior to crossing setpoint. Integral mode is incapable of causing a control loop to porpoise, and so the culprit must either be excessive gain (P) or excessive derivative action (D). Judging by the phase shift between process variable and setpoint (it looks to be exactly 180 degrees, with every positive PV peak matched by a negative Output peak, and vice-versa) it would appear that excessive gain (too much P action) is the cause.

This process appears to be self-regulating (as evidenced by different output values being necessary to stabilize the process variable at every new setpoint value), and there seems to be negligible time lags or dead time. This suggests a controller action that is predominantly based on integral (reset, or I) should work well, with just enough proportional action (P) to give fast response to setpoint and load changes.