Another category of control strategies involves the use of selector relays or function blocks with the ability to switch between different signal values, or re-direct signals to new pathways. Such functions are useful when we need a control system to choose between multiple signals of differing value in order to make the best control decisions.

The “building blocks” of such control strategies are specified relays (or function blocks in a digital control system) shown here:

*High-selector* functions output whichever input signal has the greatest value. *Low-selector* functions do exactly the opposite: output whichever input signal has the least value. “Greater-than” and “Less than” symbols are used in these two selector functions, respectively, and each type may be equipped to receive more than two input signals.

Sometimes you will see these relays represented in P&IDs simply by an inequality sign in the middle of a large bubble, rather than off to the side in a square. You should bear in mind that the location of the inequality lines has no relationship at all to the direction of the inequality symbol — e.g., it is not as though a high-selector

relay looks for the input on the left side to be greater than the input on the right. Note the examples shown below, complete with sample signal values:

High-limit and low-limit functions are similar to high- and low-select functions, but they only receive input each, and the limit value is a parameter programmed into the function rather than received from another source. The purpose of these functions is to place a set limit on how high or how low a signal value is allowed to go before being passed on to another portion of the control system. If the signal value lies within the limit imposed by the function, the input signal value is simply passed on to the output with no modification.

Like the select functions, limit functions may appear in diagrams with nothing more than the limit symbol inside the bubble, rather than being drawn in a box off to the side:

Rate limit functions place a maximum rate-of-change limit on the input signal, such that the output signal will follow the input signal precisely until and unless the input signal’s rate-of-change over time \( \frac{dx}{dt} \) exceeds the pre-configured limit value. In that case, the relay still produces a ramping output value, but rate of that ramp remains fixed at the limit \( \frac{dx}{dt} \) value no matter how fast the input keeps changing. After output value “catches up” with the input value, the function once again will output a value precisely matching the input unless the input begins to rise or fall at too fast a rate again.

**Limit controls**

A common application for select and limit functions is in cascade control strategies, where the output of one controller becomes the setpoint for another. It is entirely possible for the primary (master) controller to set a setpoint that is unreasonable or unsafe for the secondary (slave) to attain. If this possibility exists, it is wise to place a limit function between the two controllers to limit the cascaded setpoint signal.
In the following example, a cascade control system regulates the temperature of molten metal in a furnace. The output of the master (metal temperature) controller becomes the setpoint of the slave (air temperature) controller. A high limit function limits the maximum value this cascaded setpoint can attain, thereby protecting the refractory brick of the furnace from being exposed to excessive air temperatures:

![Diagram of cascade control system](image)

It should be noted that although the different functions are drawn as separate bubbles in the P&ID, it is possible for multiple functions to exist within one physical control device. In this example, it is possible to find a controller able to perform the functions of both PID control blocks (master and slave) and the high limit function as well. It is also possible to use a distributed technology such as FOUNDATION Fieldbus to place all control functions inside field instruments, so only three field instruments exist in the loop: the air temperature transmitter, the metal temperature transmitter, and the control valve (with a Fieldbus positioner).

This same control strategy could have been implemented using a low select function block rather than a high limit:
Here, the low-select function selects whichever signal value is lesser: the setpoint value sent by the main temperature controller, or the maximum air temperature limit value sent by the hand indicating control (HIC – sometimes referred to as a manual loading station).

An advantage of this latter approach over the former might be ease of limit value changes. With a configured limit value residing in a high-limit function, it might be that only qualified maintenance personnel have access to changing that value. If the decision of the operations department is to have the temperature limit value easily adjusted by anyone, the latter control strategy’s use of a manual load station would be better suited.

Another detail to note in this system is the possibility of integral windup in the master controller in the event that the high setpoint limit takes effect. Once the high-limit (or low-select) function secures the slave controller’s remote setpoint at a fixed value, the master controller’s output is no longer controlling anything it has become decoupled from the process. If, when in this state of affairs, the metal temperature is still below setpoint, the master controller’s integral action will “wind up” the output value over time with absolutely no effect, since the slave controller is no longer following its output signal. If and when the metal temperature reaches setpoint, the master controller’s output will likely be saturated at 100% due to the time it spent winding up. This will cause the metal temperature to overshoot setpoint, as a positive error will be required for the master controller’s integral action to wind back down from saturation.

A relatively easy solution to this problem is to configure the master controller to stop integral action whenever the high limit relay engages. This is easiest to do if the master PID and high limit functions both reside in the same physical controller. Many digital limit function blocks generate a bit representing the state of that bit (whether it is passing the input signal to the output or limiting the signal at the pre-configured value), some PID function blocks have a boolean input used to disable integral action. If this is the case with function blocks comprising the high-limit control strategy, it may be implemented like this:
Another method used to prevent integral windup is to make use of the feedback input available on some function blocks. This is an input used to calculate the integral term of the PID equation. In the days of pneumatic PID controllers, this option used to be called external reset. Normally connected to the output of the PID block, if connected to the output of the high-limit function it will let the controller know whether any attempt to wind up the output is having an effect. If the output has been de-selected by the high-limit block, integral windup will cease:

Limit control strategies implemented in FOUNDATION Fieldbus instruments use the same principle, except that the concept of a “feedback” signal sending information backwards up the function block chain is aggressively-applied design philosophy throughout the FOUNDATION Fieldbus standard. Nearly every function block in the Fieldbus suite provides a “back calculation” output, and nearly every function block accepts a “back calculation” input from a downstream block. The “Control Selector” (CS) function block specified in the FOUNDATION Fieldbus standard provides the limiting function we need between the master and slave controllers. The BKCAL OUT signal of this selector block connects to the master controller’s BKCAL IN input, making the master controller aware of its selection status. If ever the Control Selector function block de-selects the master controller’s output, the controller will immediately know to halt interaction:
Selector controls

In the broadest sense, a “selector” control strategy is one where one signal gets selected from multiple signals in a system to perform a measurement control function. In the context of this book and this chapter, I use the term “selector” to categorize the automatic selection of a measurement or setpoint signal. Selection of a controller output signal will be explored in the next subsection.

Perhaps one of the simplest examples of a selector control strategy is where we must select a process variable signal from multiple transmitters. For example, consider this chemical reactor, where the control system must throttle the flow of coolant to keep the hottest measured temperature at setpoint, since the reaction happens to be exothermic (heat-releasing): 2

The high-select relay (TY-24) sends only the highest temperature signal from the three transmitters to controller. The other two temperature transmitter signals are simply ignored. Another use of selector relays (or function blocks) is for the determination of a median process measurement. This sort of strategy is often used on triple-redundant measurement systems, where three transmitters are installed to measure the same process variable, providing a valid measurement even in the event of transmitter failure.

The median select function may be implemented one of two ways using high- and low-select function block
The left-hand selector strategy selects the highest value from each pair of signals (A and B, B and C, A and C), then selects the lowest value of those three primary selections. The right-hand strategy is exactly opposite – first selecting the lowest value from each input pair, then selecting the highest of those values – but it accomplishes the same function. Either strategy outputs the middle value of the three input signals.

Although either of these methods of obtaining a median measurement requires four signal selector functions, it is quite common to find function blocks available in control systems ready to perform the median selection all in a single block. The median-select function is so common to redundant sensor control systems that many control system manufacturers provide it as a standard function unto itself.

This is certainly true in the FOUNDATION Fieldbus standard, where two standardized function blocks capable of this function, the CS (Control Selector) and the ISEL (Input Selector) blocks:

Of these two Fieldbus function blocks, the latter (ISEL) is expressly designed for selecting transmitter signals whereas the former (CS) is best suited for selecting controller outputs with its “back calculation” facility designed to modify the response of all de-selected controllers. Using the terminology of this book section, ISEL function block is best suited for selector strategies, while the CS function block is ideal for override strategies (discussed in the next section). If receiving three “good” inputs, the ISEL function block will output the middle (median) value of the three. If one of the inputs carries a “bad” status, the ISEL block output...
the averaged value of the remaining two (good) inputs. Note how this function block also possesses individual "disable" inputs, giving external boolean (on/off) signals the ability to disable any one of the transmitter inputs to this block. Thus, the ISEL function block may be configured to de-select a particular transmitter input based on some programmed condition other than internal diagnostics.

If receiving four "good" inputs, the ISEL function block normally outputs the average value of the two middle (median) signal values. If one of the four inputs becomes "bad" is disabled, the block behaves as a normal three-input median select.

A general design principle for redundant transmitters is that you **never** install exactly two transmitters to measure the same process variable. Instead, you should install three (minimum). The problem with having just two transmitters is a lack of information for "voting" if the two transmitters happen to disagree. In a transmitter system, the function blocks may select the median signal value, or average the "best 2 out of 4" if there are just two transmitters installed, and they do not substantially agree with one another, i.e., anyone’s guess which one should be trusted.

**Override controls**

An "override" control strategy involves a selection between two or more controller output signals, where only one controller at a time gets the opportunity to exert control over a process. All other "de-select" controllers are thus overridden by the selected controller.

Consider this water pumping system, where a water pump is driven by a variable-speed electric motor to draw water from a well and provide constant water pressure to a customer:

![Diagram of water pumping system]

Incidentally, this is an excellent application for a variable-speed motor as the final control element rather than a control valve. Reducing pump speed in low-flow conditions will save a lot of energy over time compared to the energy that would be wasted by a constant-speed pump and control valve. A potential problem with this system is the pump running "dry" if the water level in the well gets too low, as might happen during summer months when rainfall is low and customer demand is high. If the pump runs for long with no water passing through it, the seals will become damaged. This will necessitate complete shutdown and costly rebuild of the pump, right at the time customers need it the most.

One solution to this problem would be to install a level switch in the well, sensing water level and shutting off the electric motor driving the pump if the water level ever gets too low.
This may be considered a kind of “override” strategy, because the low-level switch over-rides the pressure controller’s command for the pump to turn. It is also a crude solution to the problem, for while it protects the pump from damage, it does so at the cost of completely shutting off water to customers. One way to describe this control strategy would be to call it a hard override system, suggesting the uncompromising action it will take to protect the pump.

A better solution to the dilemma would be to have the pump merely slow down as the well water level approaches a low-level condition. This way at least the pump could be kept running (and some amount of pressure maintained), decreasing demand on the well while maintaining curtailed service to customers still protecting the pump from dry-running. This would be termed a soft override system.

We may create just such a control strategy by replacing the well water level switch with a level transmitter connecting the level transmitter to a level controller, and using a low-select relay or function block to select the lowest-valued output between the pressure and level controllers. The level controller’s setpoint will be at some low level above the acceptable limit for continuous pump operation:

If ever the well’s water level goes below this setpoint, the level controller will command the pump to slow down, even if the pressure controller is calling for a higher speed. The level controller will have overridden the pressure controller, prioritizing pump longevity over customer demand. Bear in mind that the concept of low-level switch completely shutting off the pump is not an entirely bad idea. In fact, it might be prudent to integrate such a “hard” shutdown control in the override control system, just in case something goes wrong with the level controller (e.g. an improperly adjusted setpoint or poor tuning) or the low-select function.

With two layers of safety control for the pump, this system provides both a “soft constraint” provoking moderated action and a “hard constraint” providing aggressive action to protect the pump from operation:
In order that these two levels of pump protection work in the proper order, the level controller’s setpoint needs to be set to a higher value than the low level alarm’s (LAL) trip point. A very important consideration for any override control strategy is how to manage integral windup. Any time a controller or any integral (reset) action at all is de-selected by the selector function, the integral term of the controller have the tendency to wind up (or wind down) over time. With the output of that controller de-coupled from the final control element, it can have no effect on the process variable. Thus, integral control action—purpose of which being to constantly drive the output signal in the direction necessary to achieve zero error—will work in vain to eliminate an error it cannot influence. If when control is handed back to that controller, the integral action will have to spend time “winding” the other way to undo what it did while it was de-selected.

Thus, override controls demand some form of integral windup limits that engage when a controller is selected. Methods of accomplishing this function are discussed in an earlier section on limit controls.

I generally suggest keeping such limit values inaccessible to low-level operations personnel. This is especially true in cases where the presence of a high temperature setpoint limit is intended for the longevity of the equipment. There is a strong tendency in manufacturing environments to “push the limits” of production beyond values considered safe or expedient by engineers who designed the equipment. Limits are there for a reason, and should not be altered except by people with understanding of and full responsibility over the consequences!

Only the coolant flow control instruments and piping are shown in this diagram, for simplicity. In a real P&ID, there would many more pipes, valves, and other apparatus shown surrounding this process vessel.

In order to understand how this works, I advise you try a “thought experiment” for each function block network whereby arbitrarily assign three different numerical values for A, B, and C, then see for yourself which of those three values becomes output value.

In FOUNDATION Fieldbus, each and every signal path not only carries the signal value, but also a “status” flag declaring whether “Good,” “Bad,” or “Uncertain.” This status value gets propagated down the entire chain of connected function blocks, to dependent blocks of a possible signal integrity problem if one were to occur.

This principle holds true even for systems with no function blocks “voting” between the redundant transmitters. Perhaps installation consists of two transmitters with remote indications for a human operator to view. If the two displays substantially disagree, which one should the operator trust? A set of three indicators would be much better, providing the operator enough information to make an intelligent decision on which display(s) to trust.

Go Back to Lessons in Instrumentation Table of Contents