

(12) UK Patent

(19) GB

(11) 2545057

(45) Date of B Publication

(13) B

18.05.2022

(54) Title of the Invention: Velocity based control in a non-periodically updated controller

(51) INT CL: **G05B 11/42** (2006.01)    **G05B 17/02** (2006.01)    **G05B 19/418** (2006.01)

(21) Application No: **1617274.4**

(22) Date of Filing: **11.10.2016**

(30) Priority Data:

(31) **62/240322**

(32) **12.10.2015**

(33) **US**

(31) **15/081189**

(32) **25.03.2016**

(33) **US**

(43) Date of A Publication **07.06.2017**

(56) Documents Cited:

WO 2015/143298 A1

US 7620460 B2

US 20130184837 A1

US 20080082180 A1

(58) Field of Search:

As for published application 2545057 A viz:

INT CL **G05B**

Other: **WPI, EPDOC, Patent Fulltext**

updated as appropriate

Additional Fields

Other: **None**

(72) Inventor(s):

Terrence L Blevins

Pete Stoltenow

Mark John Nixon

(73) Proprietor(s):

Fisher-Rosemount Systems, Inc

1100 W. Louis Henna Blvd, Bldg 1,

Round Rock 78681, Texas, United States of America

(74) Agent and/or Address for Service:

Forresters IP LLP

Rutland House, 148 Edmund Street, BIRMINGHAM,

B3 2JA, United Kingdom

GB

2545057 B

FIG. 1

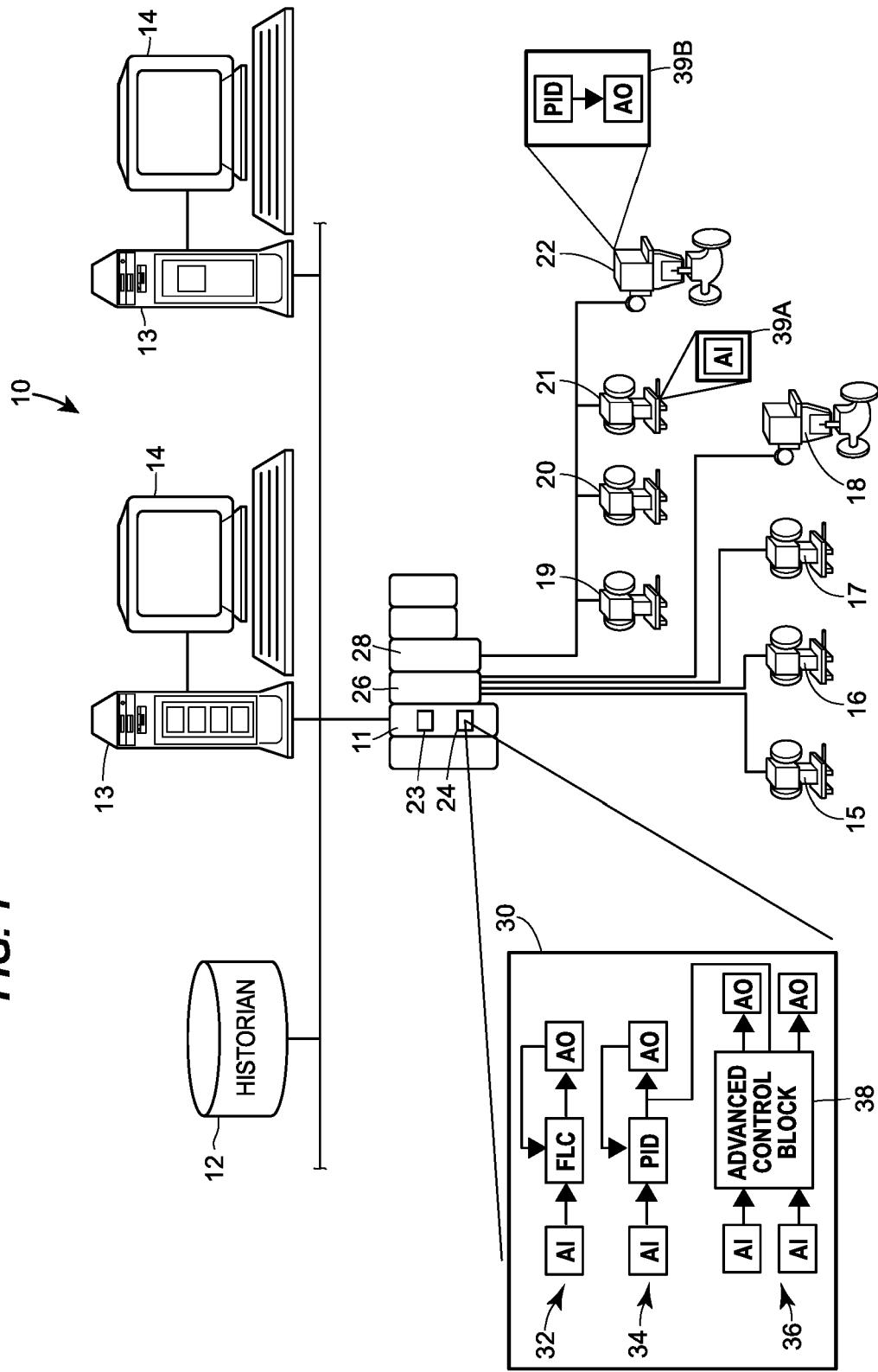
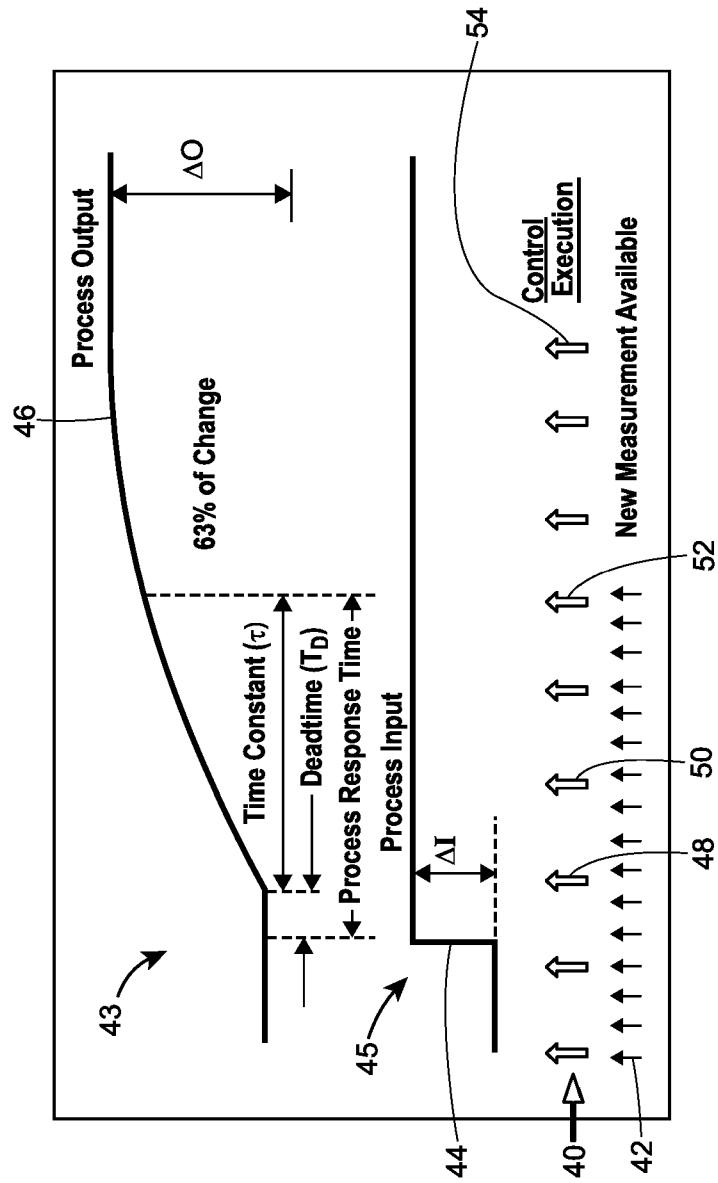


FIG. 2



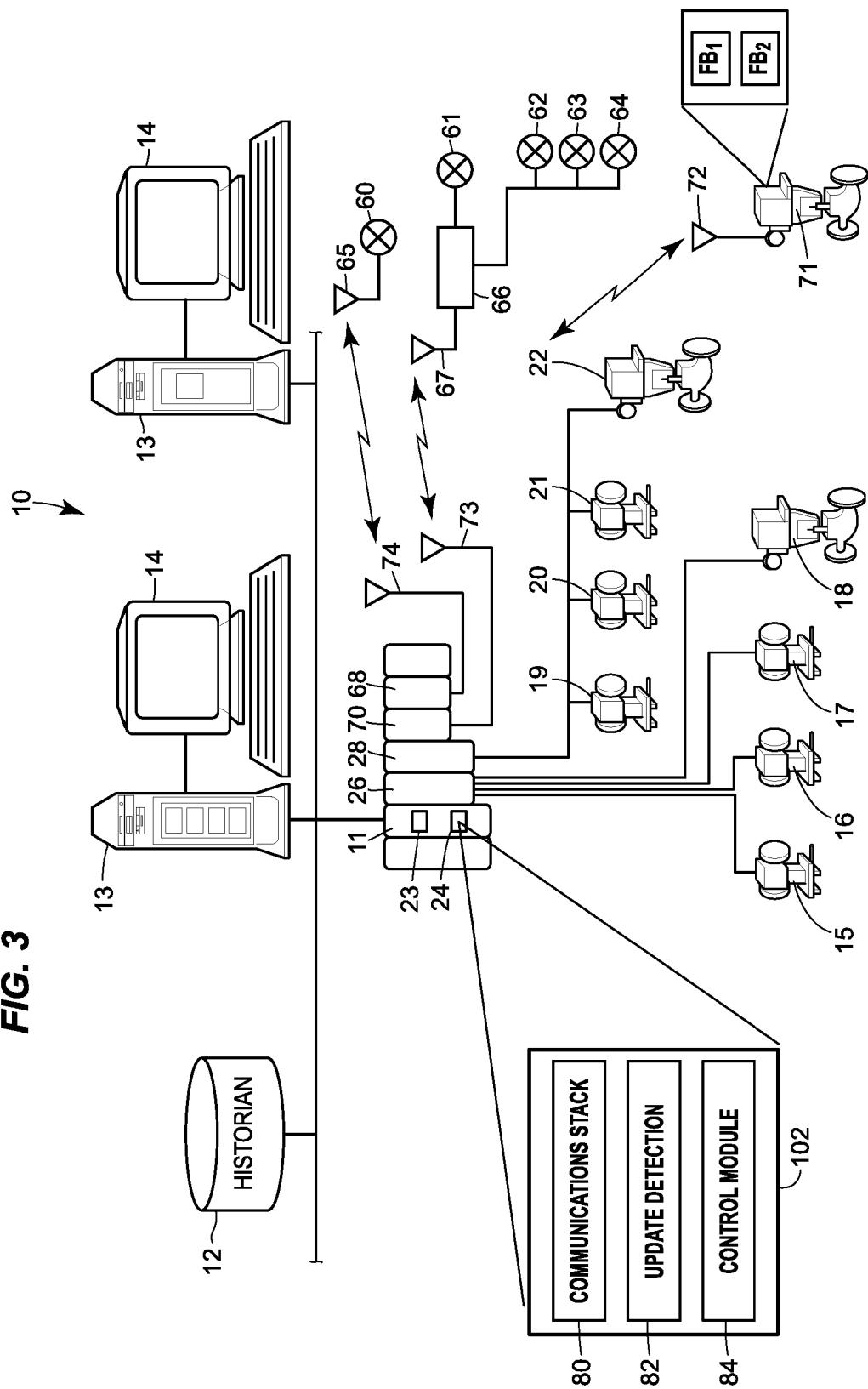
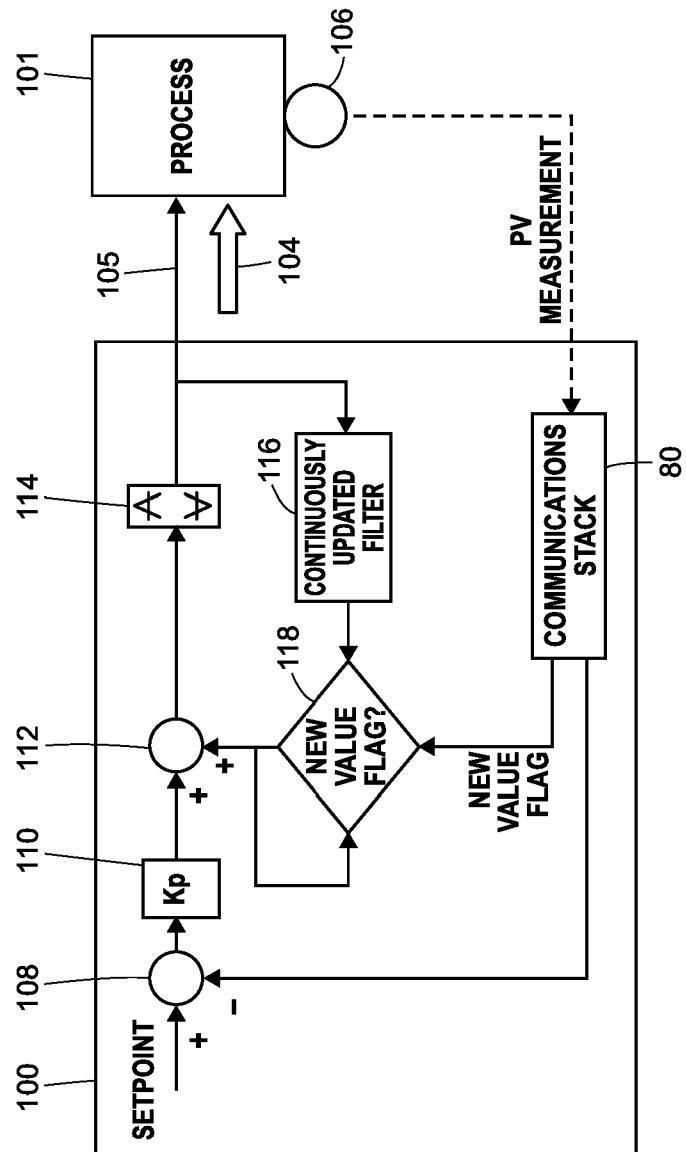
**FIG. 3**

FIG. 4A



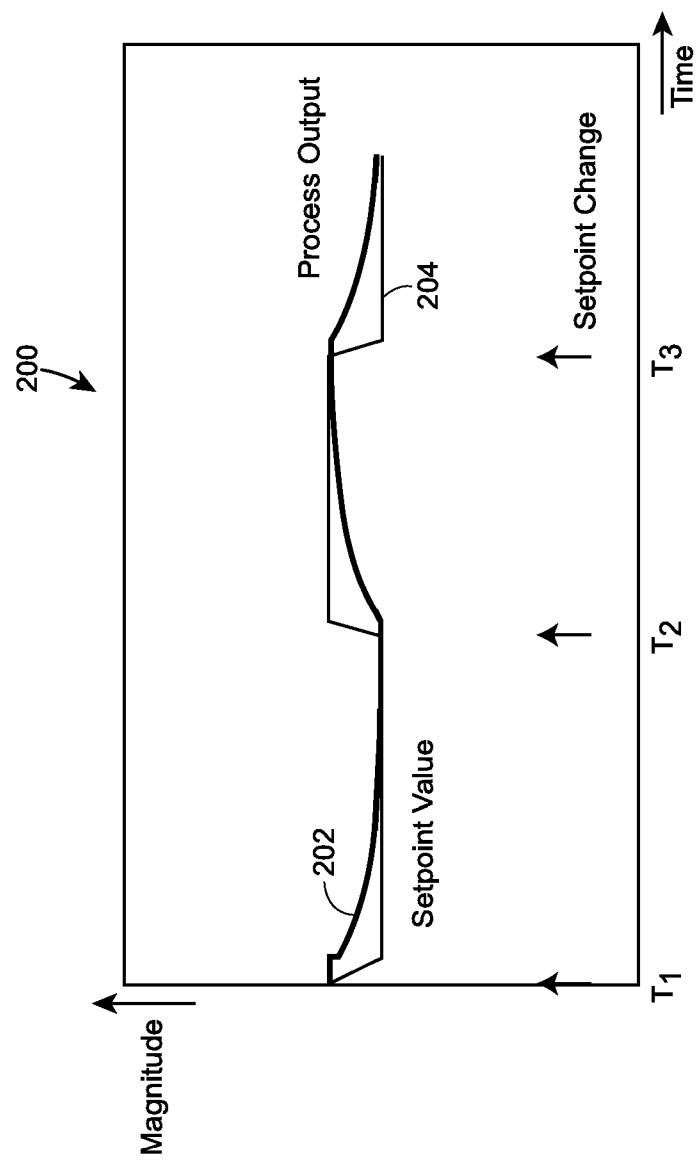


FIG. 4B

FIG. 5

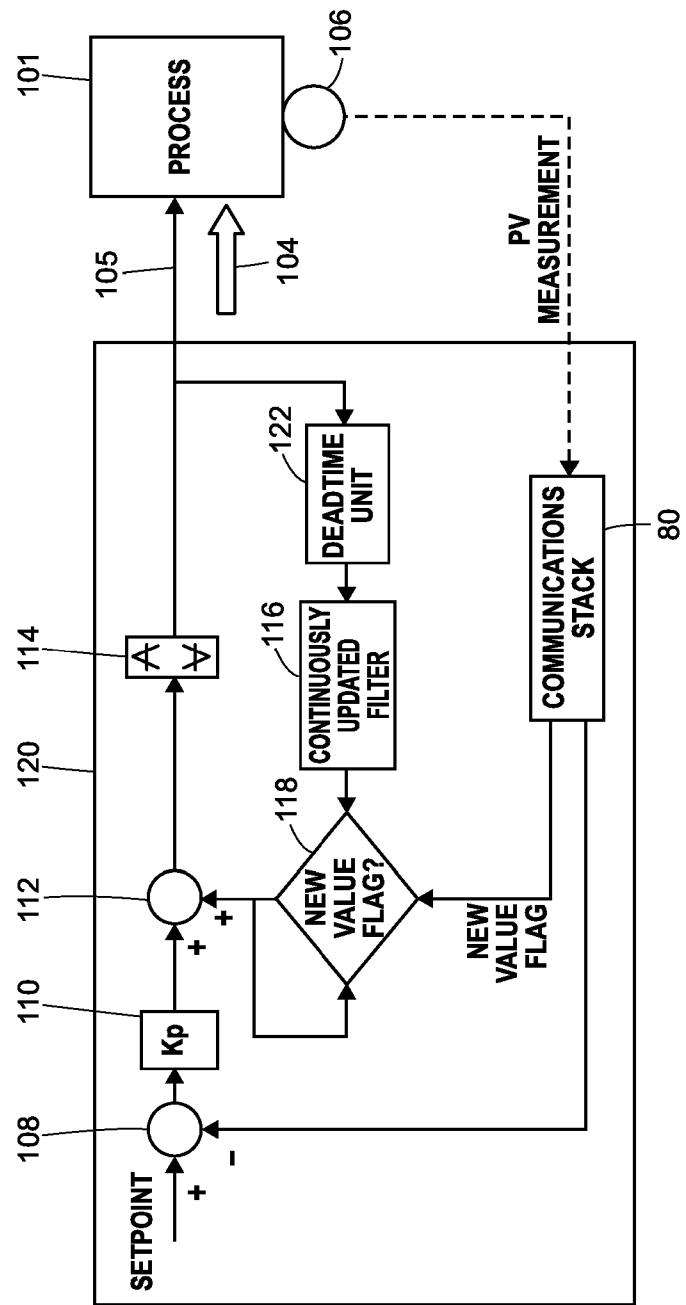


FIG. 6

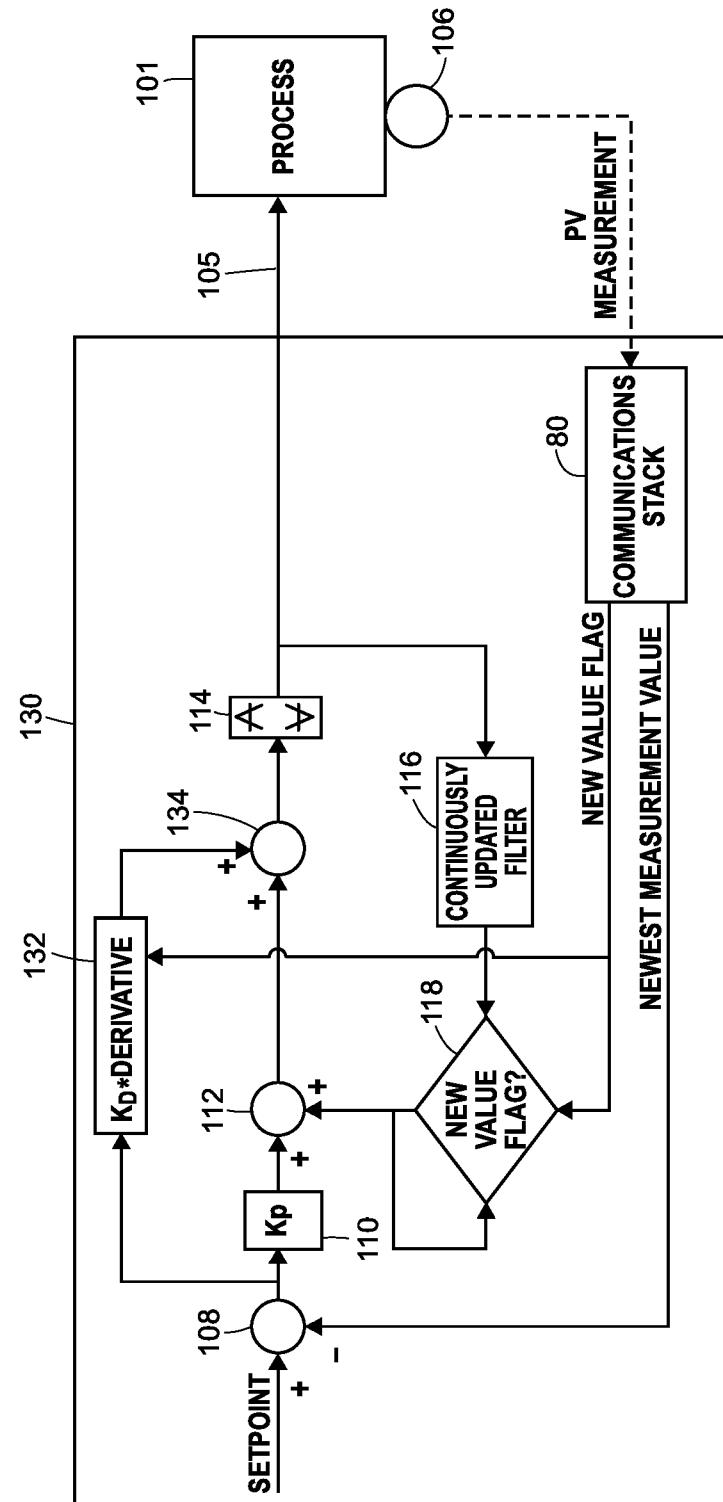


FIG. 7

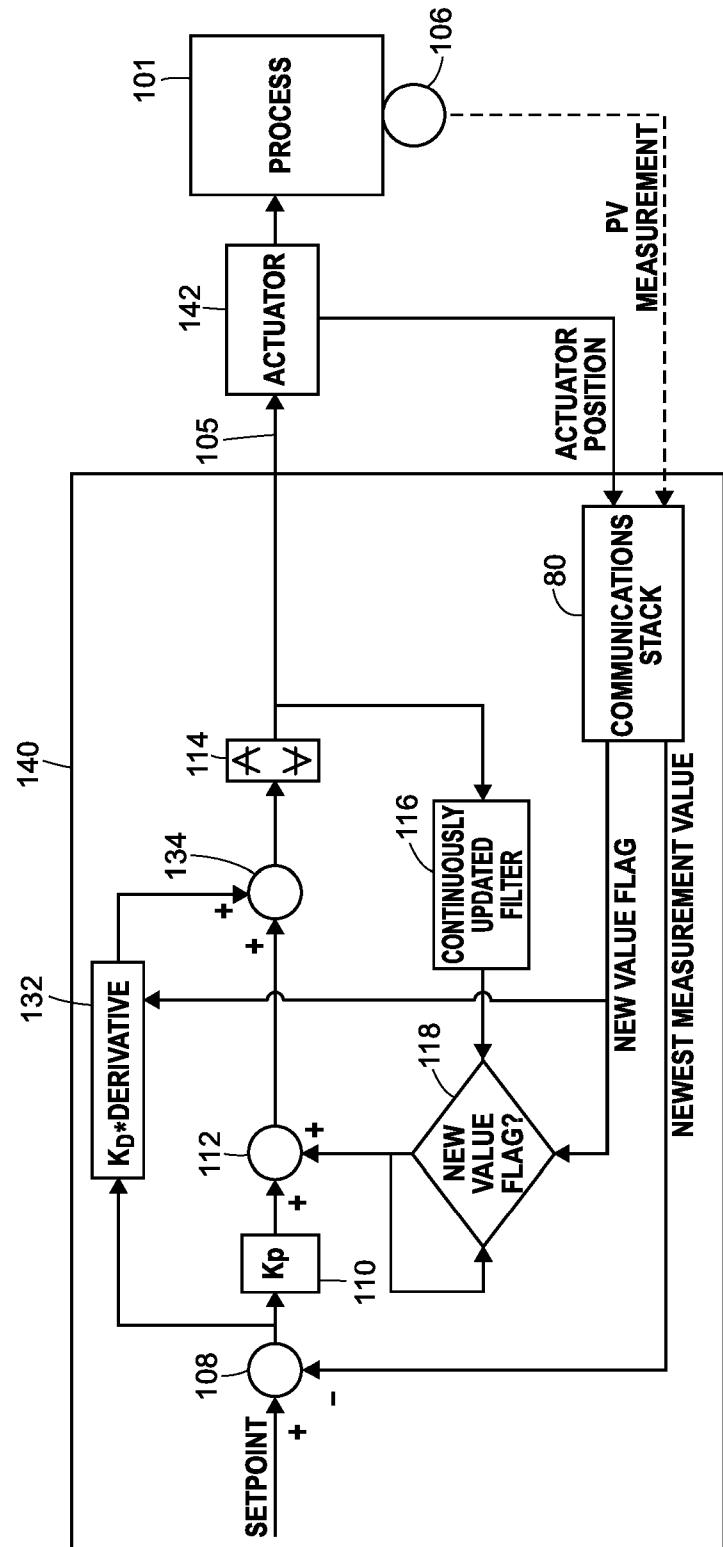


FIG. 8

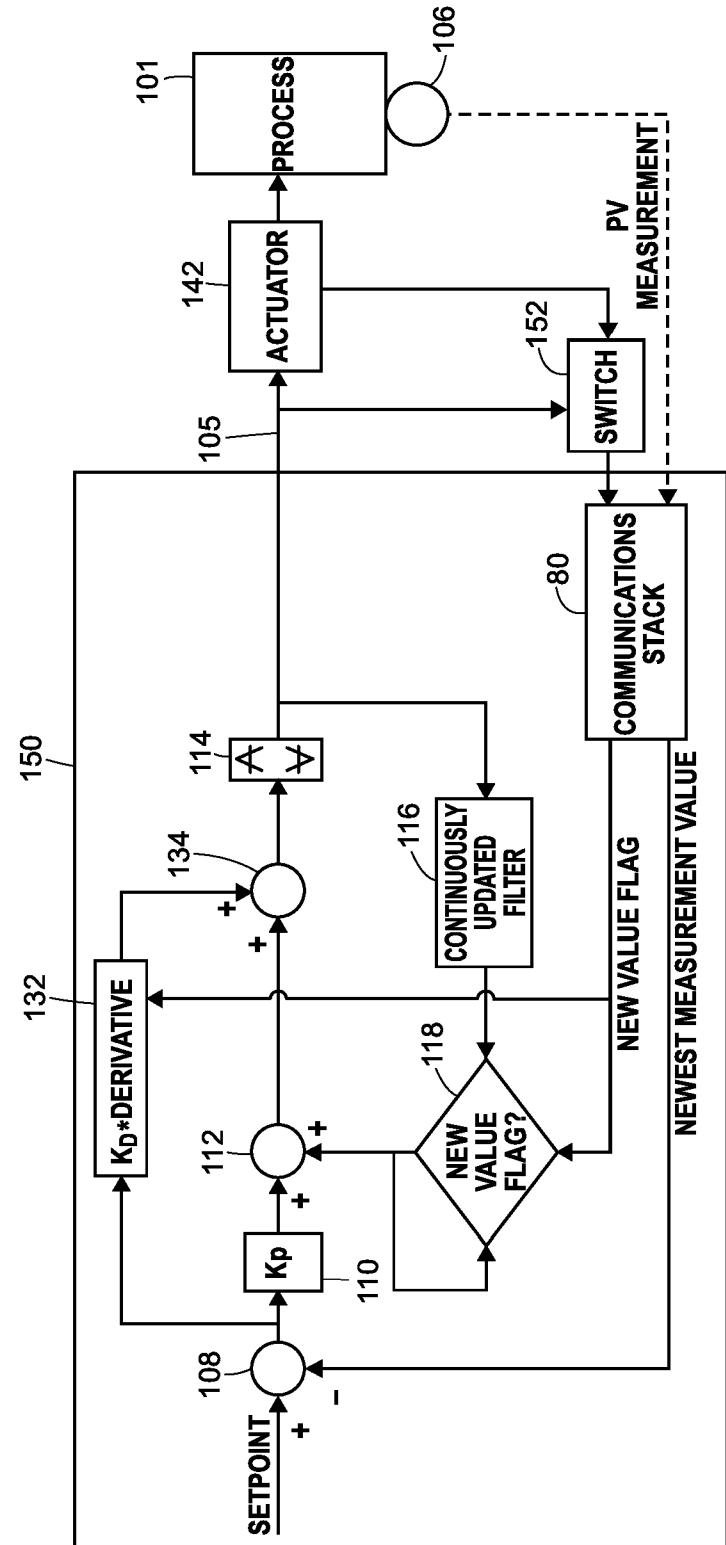


FIG. 9

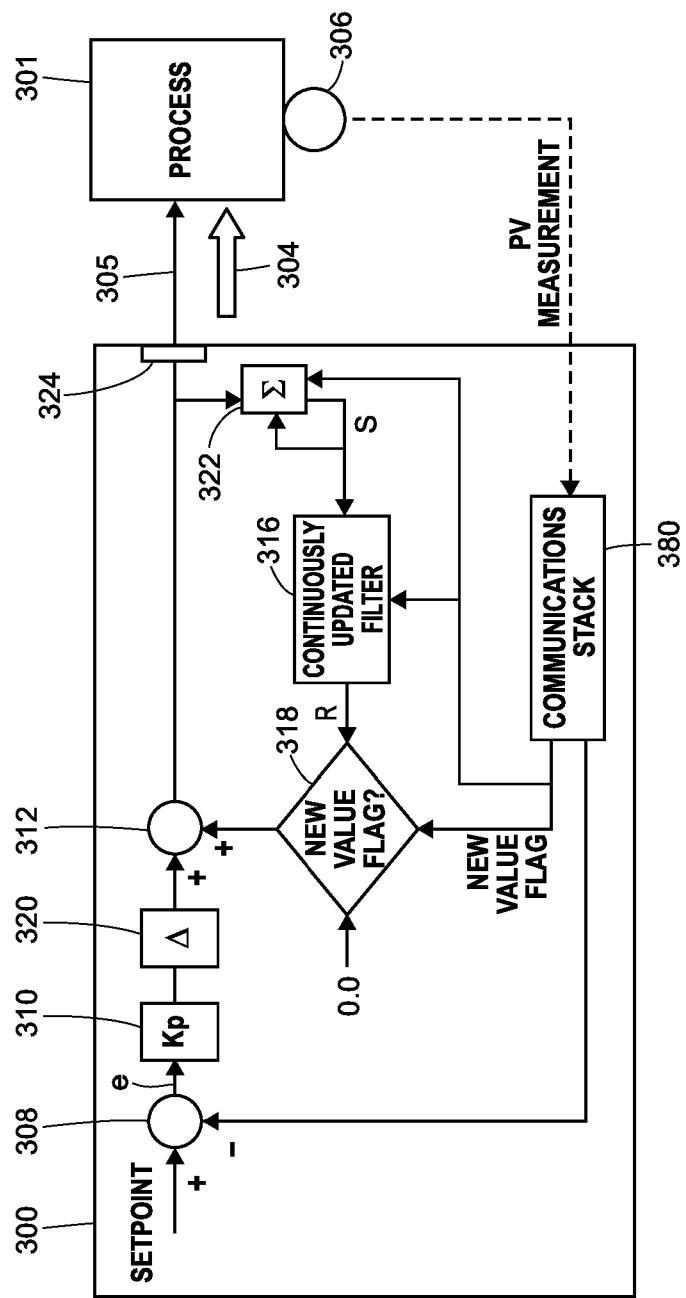


FIG. 10

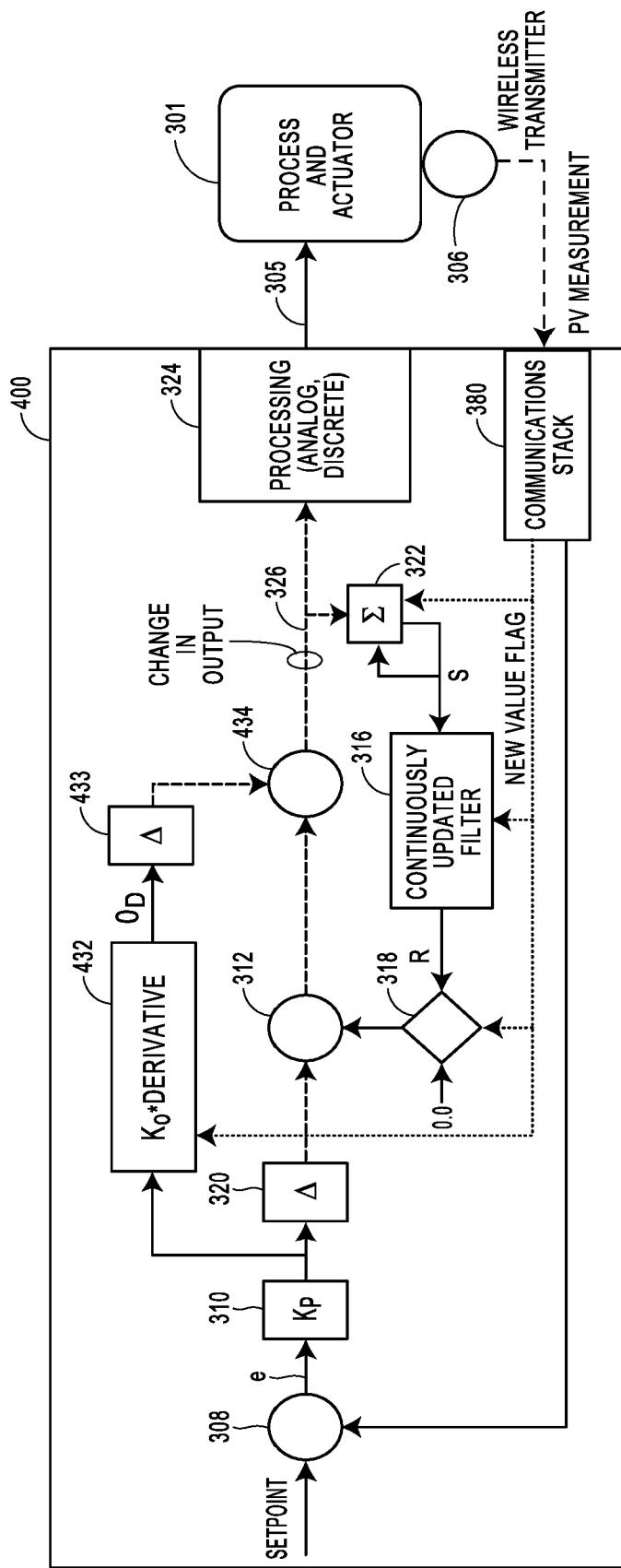


FIG. 11A

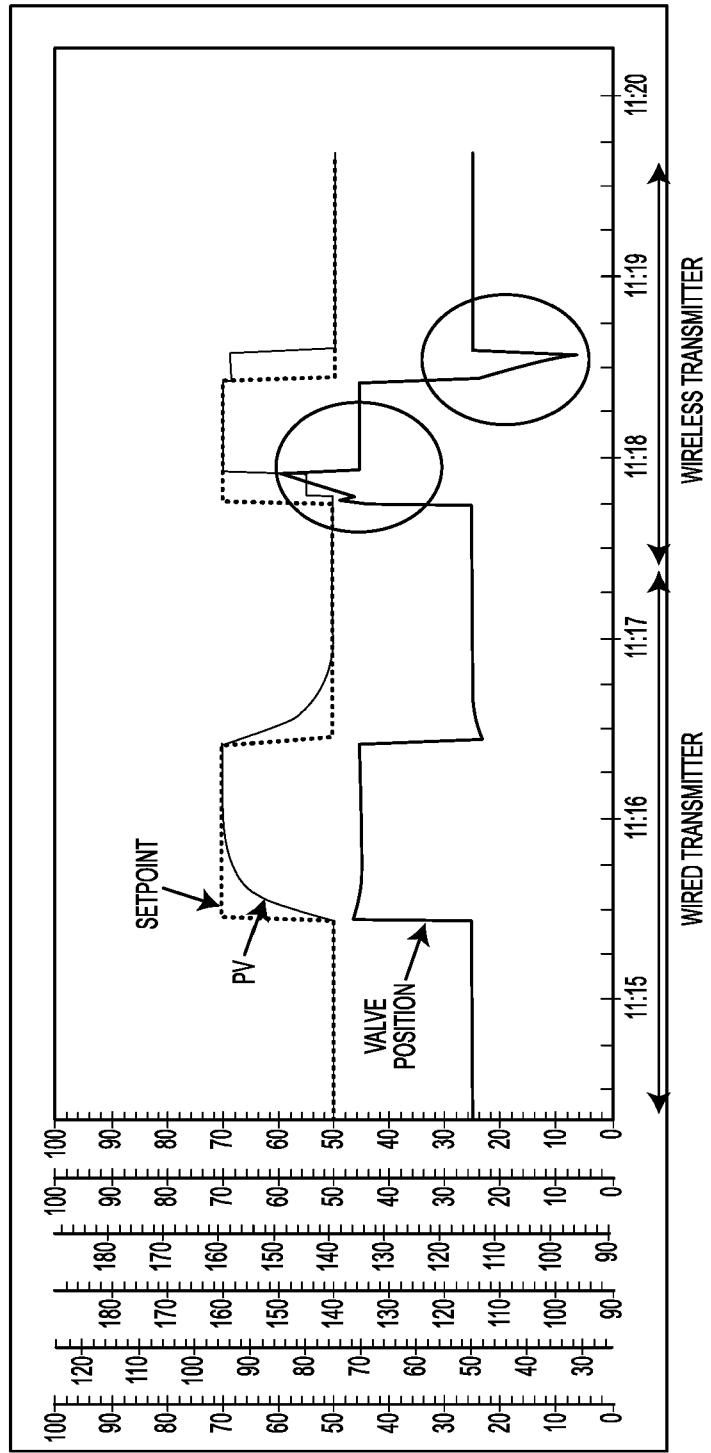


FIG. 11B

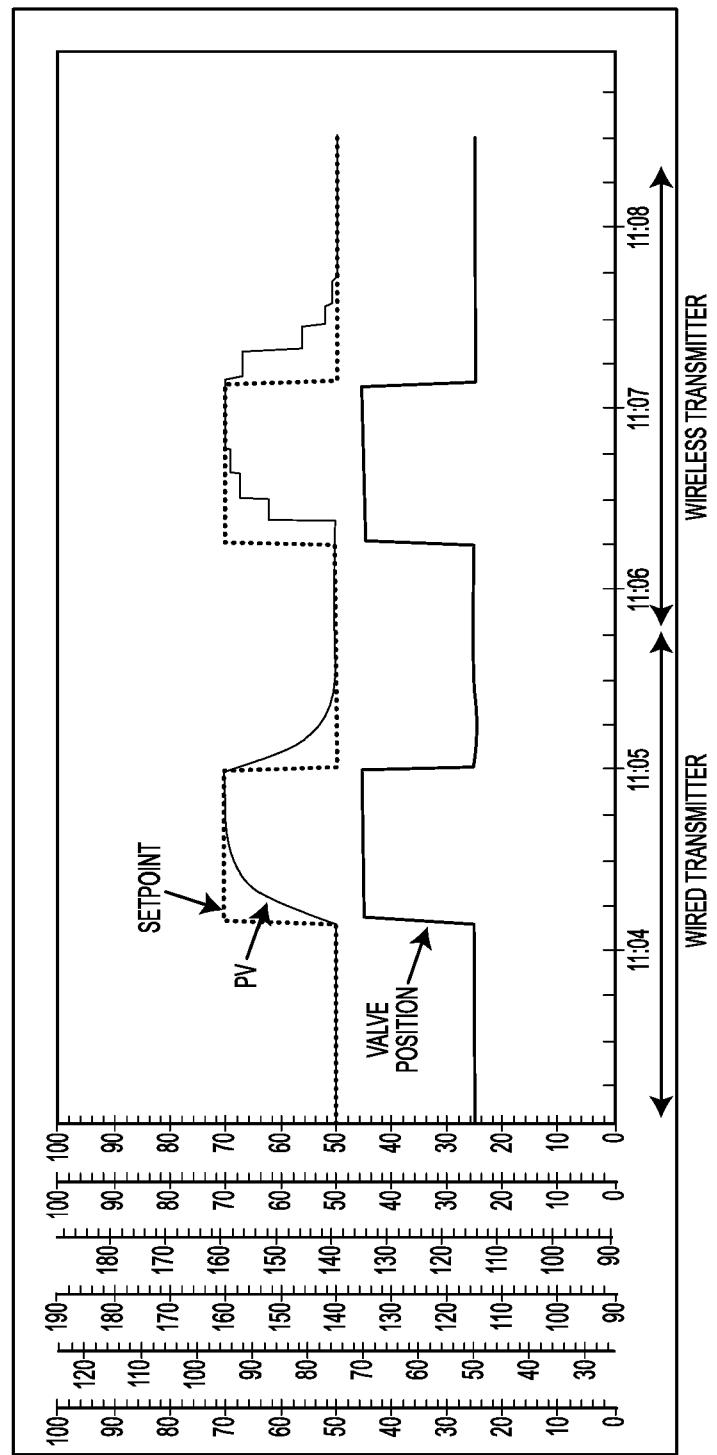


FIG. 12A

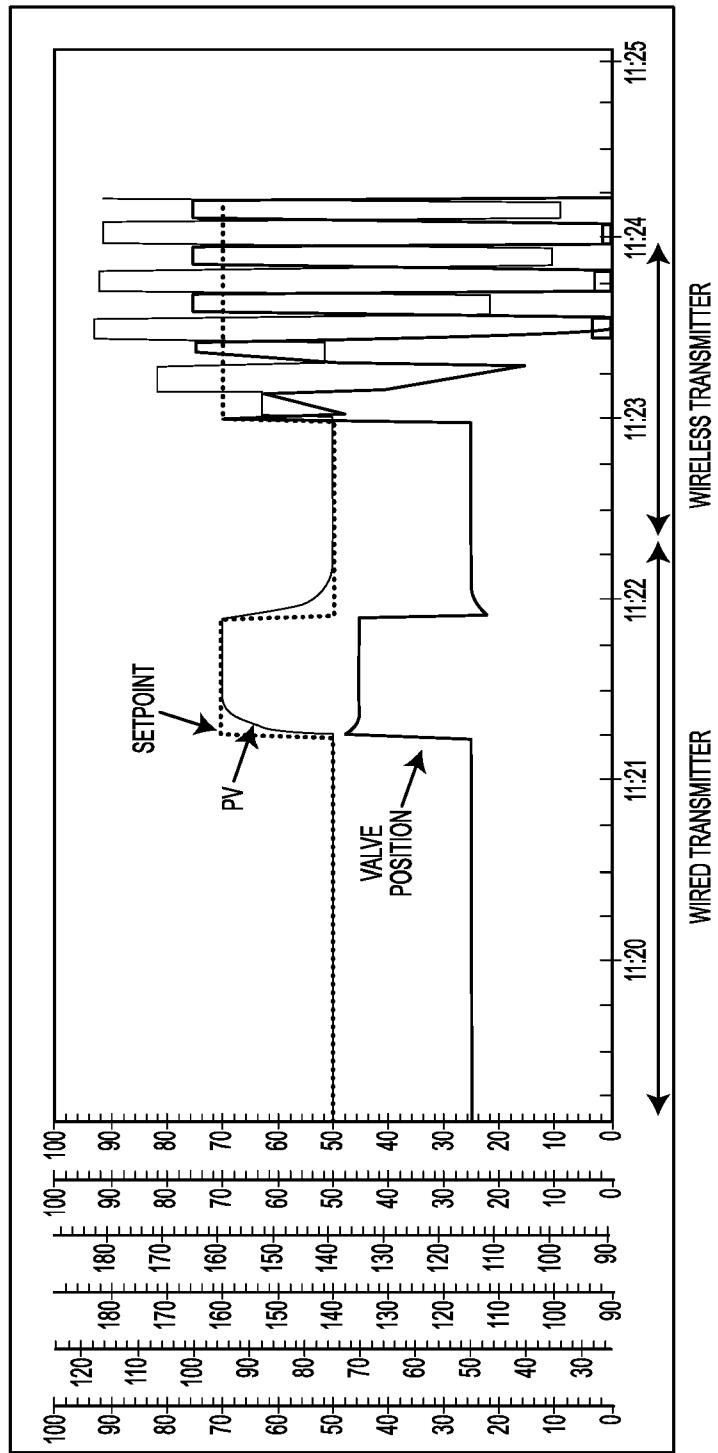


FIG. 12B

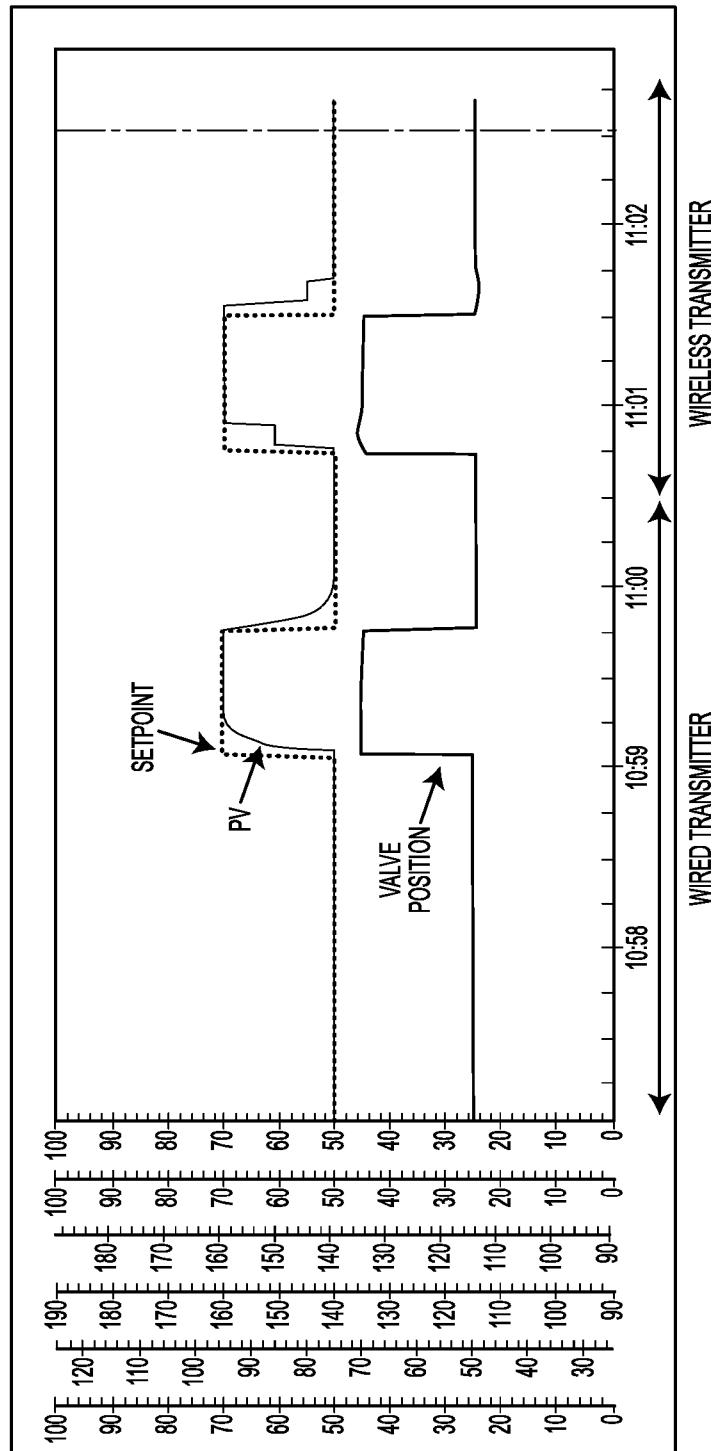


FIG. 13A

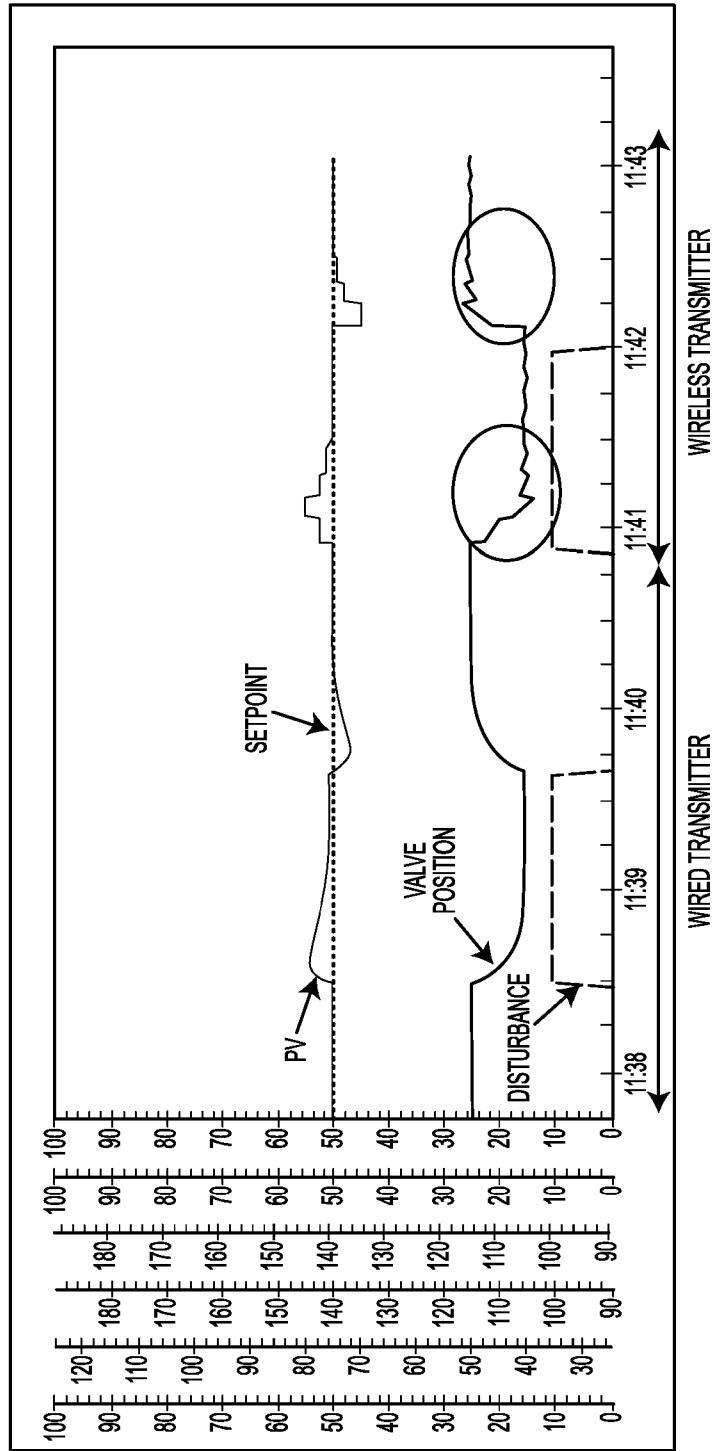


FIG. 13B

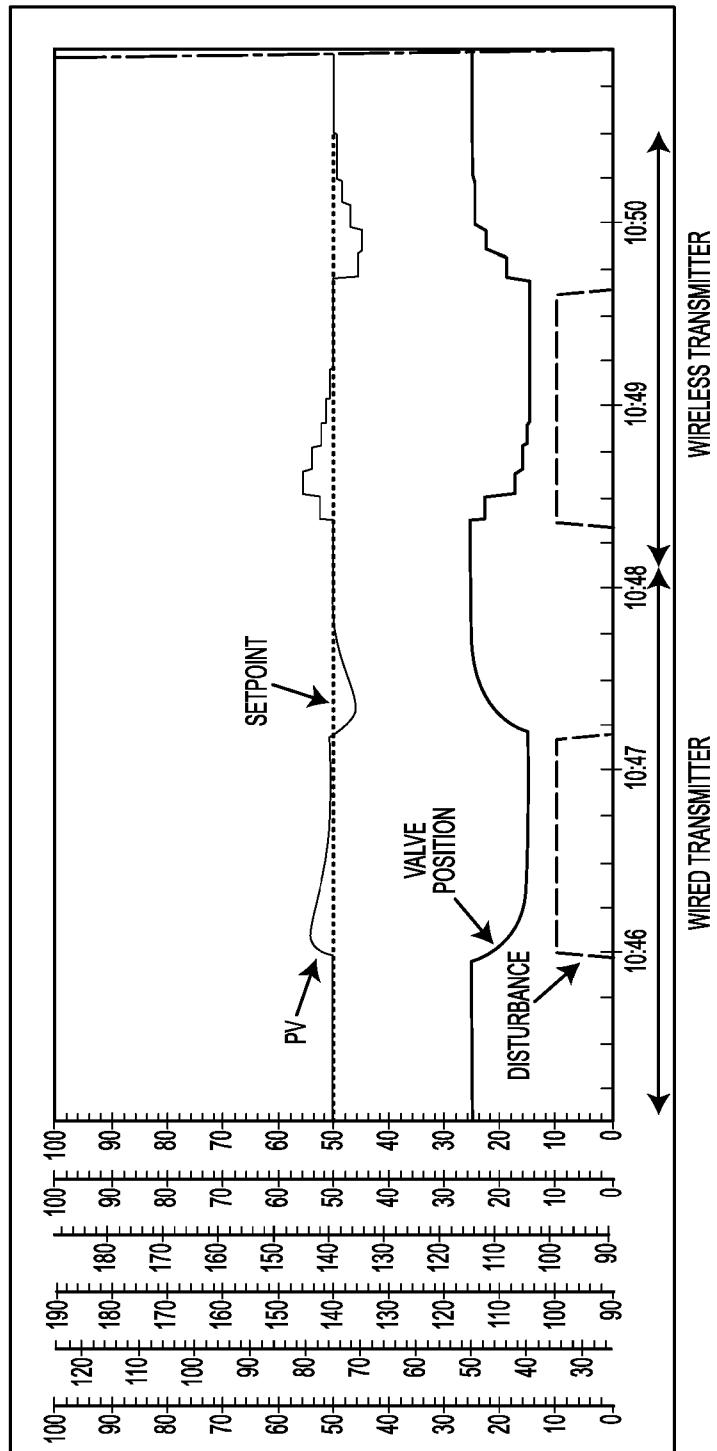


FIG. 14A

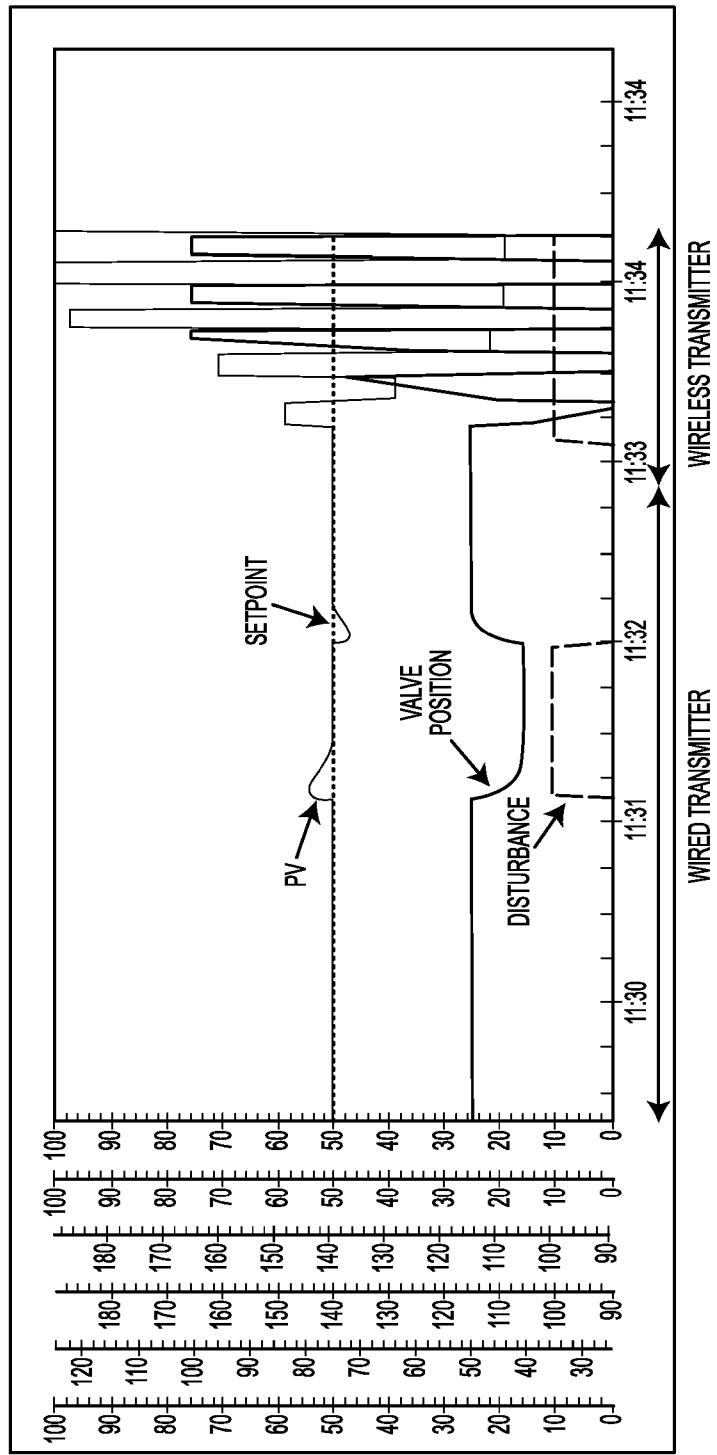
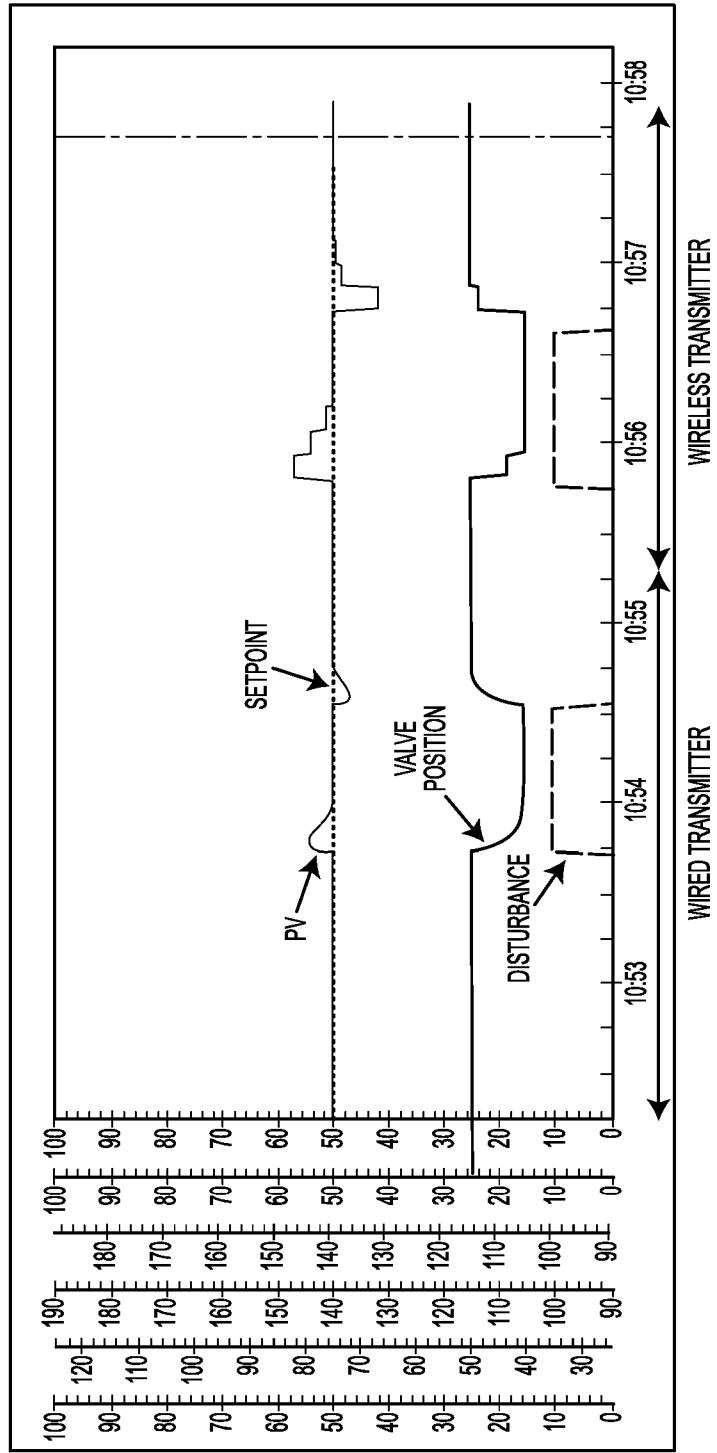


FIG. 14B



21 02 22

## VELOCITY BASED CONTROL IN A NON-PERIODICALLY UPDATED CONTROLLER

**[0001]** This patent relates to compensating a method and system for providing velocity based control in a process control system that uses non-periodic control or slow feedback process variable communications and, more particularly, to a device and method configured to robustly control a process when implementing control when receiving process variable feedback at a rate that is slow compared to the process dynamics.

**[0002]** Process control systems, such as distributed or scalable process control systems like those used in chemical, petroleum or other processes, typically include one or more process controllers communicatively coupled to each other, to at least one host or operator workstation and to one or more field devices via analog, digital or combined analog/digital buses. The field devices, which may be, for example, valves, valve positioners, switches and transmitters (e.g., temperature, pressure and flow rate sensors), perform functions within the process such as opening or closing valves and measuring process parameters. The process controller receives signals indicative of process measurements made by the field devices and/or other information pertaining to the field devices, and uses this information to implement a control routine to generate control signals which are sent over the buses to the field devices to control the operation of the process. Information from the field devices and the controller is typically made available to one or more applications executed by the operator workstation to enable an operator to perform any desired function with respect to the process, such as viewing the current state of the process, modifying the operation of the process, etc.

**[0003]** Some process control systems, such as the DeltaV® system sold by Emerson Process Management, use function blocks or groups of function blocks referred to as modules located in the controller or in different field devices to perform control and/or monitoring operations. In these cases, the controller or other device is capable of including and executing one or more function blocks or modules, each of which receives inputs from and/or provides outputs to other function blocks (either within the same device or within different devices), and performs some process operation, such as measuring or detecting a process parameter, monitoring a device, controlling a device, or performing a control operation, such as the implementation of a proportional-derivative-integral (PID) control routine. The different function blocks and modules within a process control system are generally configured to communicate with each other (e.g., over a bus) to form one or more process control loops.

**[0004]** Process controllers are typically programmed to execute a different algorithm, sub-routine or control loop (which are all control routines) for each of a number of different loops defined for, or contained within a process, such as flow control loops, temperature control loops, pressure control loops, etc. Generally speaking, each such control loop includes one or more input blocks, such as an analog input (AI) function block, a single-output control block, such as a proportional-integral-derivative (PID) or a fuzzy logic control function block, and an output block, such as an analog output (AO) function block. Control

routines, and the function blocks that implement such routines, have been configured in accordance with a number of control techniques, including PID control, fuzzy logic control, and model-based techniques such as a Smith Predictor or Model Predictive Control (MPC).

**[0005]** To support the execution of the routines, a typical industrial or process plant has a centralized control room communicatively connected with one or more process controllers and process I/O subsystems, which, in turn, are connected to one or more field devices. Traditionally, analog field devices have been connected to the controller by two-wire or four-wire current loops for both signal transmission and the supply of power. An analog field device that transmits a signal to the control room (e.g., a sensor or transmitter) modulates the current running through the current loop, such that the current is proportional to the sensed process variable. On the other hand, analog field devices that perform an action under control of the control room is controlled by the magnitude of the current through the loop.

**[0006]** With the increased amount of data transfer, one particularly important aspect of process control system design involves the manner in which field devices are communicatively coupled to each other, to controllers and to other systems or devices within a process control system or a process plant. In general, the various communication channels, links and paths that enable the field devices to function within the process control system are commonly collectively referred to as an input/output (I/O) communication network.

**[0007]** The communication network topology and physical connections or paths used to implement an I/O communication network can have a substantial impact on the robustness or integrity of field device communications, particularly when the network is subjected to adverse environmental factors or harsh conditions. These factors and conditions can compromise the integrity of communications between one or more field devices, controllers, etc. The communications between the controllers and the field devices are especially sensitive to any such disruptions, inasmuch as the monitoring applications or control routines typically require periodic updates of the process variables for each iteration of the routine. Compromised control communications could therefore result in reduced process control system efficiency and/or profitability, and excessive wear or damage to equipment, as well as any number of potentially harmful failures.

**[0008]** In the interest of assuring robust communications, I/O communication networks used in process control systems have historically been hardwired. Unfortunately, hardwired networks introduce a number of complexities, challenges and limitations. For example, the quality of hardwired networks may degrade over time. Moreover, hardwired I/O communication networks are typically expensive to install, particularly in cases where the I/O communication network is associated with a large industrial plant or facility distributed over a large area, for example, an oil refinery or chemical plant consuming several acres of land. The requisite long wiring runs typically involve substantial amounts of labor, material and expense, and may introduce signal degradation arising from wiring impedances and electromagnetic interference. For these and other reasons, hardwired I/O communication networks are generally difficult to reconfigure, modify or update.

**[0009]** It has been suggested to use wireless I/O communication networks to alleviate some of the difficulties associated with hardwired I/O networks. For example, U.S. Patent Application Publication

No. 2003/0043052, entitled "Apparatus for Providing Redundant Wireless Access to Field Devices in a Distributed Control System," the entire disclosure of which is hereby expressly incorporated by reference herein, discloses a system utilizing wireless communications to augment or supplement the use of hardwired communications.

**[0010]** Reliance on wireless communications for control-related transmissions has traditionally been limited due to, among other things, reliability concerns. As described above, modern monitoring applications and process control relies on reliable data communication between the controller and the field devices to achieve optimum control levels. Moreover, typical controllers execute control algorithms at fast rates to quickly correct unwanted deviations in the process. Undesirable environmental factors or other adverse conditions may create intermittent interferences that impede or prevent the fast communications necessary to support such execution of monitoring and control algorithms. Fortunately, wireless networks have become much more robust over the last two decades, enabling the reliable use of wireless communications in some types of process control systems.

**[0011]** However, power consumption is still a complicating factor when using wireless communications in process control applications. Because wireless field devices are physically disconnected from the I/O network, the field devices typically need to provide their own power source. Accordingly, field devices may be battery powered, draw solar power, or pilfer ambient energy such as vibration, heat, pressure, etc. For these devices, energy consumed for data transmission may constitute a significant portion of total energy consumption. In fact, more power may be consumed during the process of establishing and maintaining a wireless communication connection than during other important operations performed by the field device, such as the steps taken to sense or detect the process variable being measured. To reduce power consumption in wireless process control systems and thus prolong battery life, it has been suggested to implement a wireless process control system in which the field devices, such as sensors, communicate with the controller in a non-periodic, slow or intermittent manner. In one case, the field devices may communicate with or send process variable measurements to the controller only when a significant change in a process variable has been detected, leading to non-periodic communications with the controller.

**[0012]** One control technique that has been developed to handle non-periodic process variable measurement updates uses a control system that provides and maintains an indication of an expected process response to the control signal produced by the controller between the infrequent or non-periodic measurement updates. An expected process response may be developed by a mathematical model that calculates an expected process response to a control signal for a given measurement update. One example of this technique is described in U.S. Patent No. 7,587,252, entitled, "Non-Periodic Control Communications in Wireless and Other Process Control Systems," the entire disclosure of which is hereby expressly incorporated by reference herein. In particular, this patent discloses a control system having a filter that generates an indication of an expected process response to a control signal upon the receipt of a non-periodic process variable measurement update and that maintains the generated indication of the expected process response until the arrival of the next non-periodic process variable measurement update. As another example, U.S. Patent No. 7,620,460, entitled "Process Control With Unreliable

Communications," the entire disclosure of which is hereby expressly incorporated by reference herein, discloses a system that includes a filter that provides an indication of an expected response to the control signal but further modifies the filter to incorporate a measurement of the time that has elapsed since a last non-periodic measurement update to generate a more accurate indication of the expected process response.

**[0013]** However, in many control applications, a process control system may receive a setpoint change during process operation. Generally, when a setpoint is changed during the execution of a periodically updated control system (e.g. a hard-wired control communication system), a controller that is designed to take proportional action on the error between the setpoint and the measured process variable will immediately change the controller output to drive the process variable towards the new steady state value. However, in a wireless control system that receives infrequent, non-periodic measurement updates that operates as described in both of the examples above, the measured process response reflected by each new measurement update reflects changes made in the controller output taken on account of the last measurement update in addition to changes in the output resulting from the setpoint change made sometime after the last measurement update was received. In this case, the calculation of the controller reset component based on the controller output and the time since the last measurement update (as described in U.S. Patent No. 7,620,460) may over compensate for changes made after the last measurement update. The process response to a setpoint change may therefore differ based on when the setpoint change was made after the last measurement update. As a result, this system does not respond as quickly or as robustly to a setpoint change because the controller continues to rely on a previously generated (and now outdated) indication of the expected response when developing the control signal after a change in the setpoint. To overcome this problem, U.S. Patent Application Publication No. 2013/0184837, entitled "Compensating for Setpoint Changes in a Non-Periodically Updated Controller," the entire disclosure of which is hereby expressly incorporated by reference herein, discloses a system that uses a continuously updated filter in a feedback loop within the controller to track the operation of the controlled variable during the times at which no new process variable measurement is received and uses the output of this controller when a new process variable measurement is received at the controller, but otherwise uses the output of the filter from the most recent time that a measurement of the process variable has been received to generate a control signal. In this system, the control signal generated is more responsive to and operates better when there are changes in the setpoint of the process variable between times at which process variable measurements are received.

**[0014]** Moreover, when using battery powered transmitters in wireless control systems, it is desirable to set up the system to maintain a long battery life. For example, to obtain a battery life of 3-5 years using current transmitter and battery technology, it is generally necessary to use a communication update rate of 8 seconds or greater. Using such a slow update rate, however, restricts using PID (proportional-integral-derivative) based wireless control to processes that have a process response time of 30 seconds or more because it is important to still have process feedback measurements received at a rate that is at least four times the rate associated with the response time of the process (i.e., the inverse of the process response time) to maintain control of the process.

**[0015]** Still further, various types of PID algorithms exist that may be used to address wireless control, including those using producing a positional output, e.g., a 4-20ma signal or digital signal that is provided to the valve or other controlled element telling the element the ultimate position to which to move. However, other PID algorithms exist that provide velocity based control signals which instruct the valve or other controlled element to move a certain amount in a certain direction, for example, by energizing the moveable element for a certain amount of time. Such velocity based control signals are commonly used with electric motors and provide control signal in the form of pulsed signal (having a pulse width that is modulated to indicate the amount of time the valve should energize the movable element to move over a particular period of time). Velocity based controllers tend to produce a change in position signal, as opposed to a signal indicating an actual position to be obtained by the moveable element. Thus, velocity based control algorithms tend to be used to provide an incremental (increase/decrease) output to an actuator and so may be used in controlling actuators that may not provide position feedback.

#### SUMMARY

**[0016]** A control technique allows for the robust control of a process or control loop having fast dynamics with respect to the rate at which measured values of a controlled process variable are provided as feedback to the process controller. In particular, the control technique uses a positional or a velocity form of a PID algorithm to be able to control processes in which a process variable measurements or feedback signals are provided to the controller at a time interval that is equal to or even greater than the process response time. In particular, the control technique may be used to provide robust control in processes having a response time that is two to four times less than the feedback time interval. Such situations can arise, for example, when using wireless control in which the process variable feedback measurements are provided to the controller wirelessly, intermittently, or at a time interval that is less than, close to or even greater than the response time of the process.

**[0017]** The disclosed velocity PID control routine may be used in many different situations, such as to perform control using a wireless measurement when the actuator requires an incremental input and no position feedback is available to the controller, to perform control with wired measurements when interfacing to an actuator that requires a positional or incremental input, and to address traditional installations as well as newer installations where wireless measurements are used in control. Moreover, tuning of the velocity form of the PID algorithm may be performed based on the process gain and dynamics and is independent of the wireless communication rate. Still further, the velocity form of the PID control routine automatically holds the last output position on loss of communications and provides bumpless recovery when communications are re-established.

**[0018]** In one case, the controller that implements the new control technique generally includes differential construction in that differential proportional, integral, and derivative control signal components are generated and used to create a differential or movement based control signal which is then sent to the controlled device, e.g., a valve, to control the operation of the controlled device and thus the process. This differential or velocity based form of control generates a control signal that operates better than standard PID control in the presence of slow process variable feedback measurements. In particular, the controller using this control technique generates, during each controller iteration, a differential proportional value

21 02 22

representing the difference between a previous proportional control signal component and a newly calculated proportional control signal component and this differential proportional value is used as the basis for each new control signal from the controller. However, various other control signal components, e.g., a derivative and/or an integral control signal component, may be added to or combined with the differential proportion control signal component when a new process variable measurement signal is available at the controller. These two control signal components may also be based on differentials between newly calculated values and previously calculated values. In particular, a new derivative component may be calculated during the controller iterations at which a newly received value of a process variable measurement is available at the controller. Likewise, a new integral component may be developed using a continuously updated filter that generates a new indication of an expected response of the process for each control routine iteration of the controller. However, the output of the continuously updated filter is used to generate a new integral component only when a new value of a process variable measurement is received. At other times, the integral control signal component is set to zero.

**[0019]** The velocity based PID control technique disclosed herein uses the differential signal form to produce a control signal that adapts quickly to changes in the setpoint (even between times at which a process variable feedback signal is present at the input of the controller) while still providing robust and stable control in the presence of slowly received (e.g., intermittent) feedback signals, including feedback signals received at a rate that is less than (e.g., even two to four times less than), close to or greater than the inverse of the response time of the process being controlled.

**[0020]** Accordingly, an aspect of the present invention provides a method of controlling a process, comprising: implementing, on a computer processing device, multiple iterations of a control routine to generate a control signal including, during each iteration of the control routine; generating, using a computer processing device, an integral feedback contribution for use in producing the control signal, including using, during each of the multiple iterations, including a current iteration, a continuously updated an iterative filter to determine a current integral feedback contribution value for the current iteration of the control routine from an integral feedback contribution value of a preceding iteration of the control routine and a summed control signal produced by summing the control signal generated during each iteration of the control routine since the last iteration of the control routine at which a new process response indication was received; and using, as determined by a switch coupled to the continuously updated filter, the current integral feedback contribution to generate the control signal for the current iteration of the control routine during each controller iteration at which a new process response indication is received, and not using, as determined by the switch, the current integral feedback contribution to generate the control signal during controller iterations at which a new process response indication to the control signal is not received; and using the control signal to control the process.

**[0021]** The method may further include generating a proportional contribution during each iteration and using the proportional contribution during each iteration to generate the control signal.

**[0022]** The new process response indication may be a measurement of a process parameter being affected by the control signal.

**[0023]** The process parameter may be a process variable controlled by a field device that is responsive to the control signal.

**[0024]** Determining the integral feedback contribution may include generating the integral feedback contribution value based on the difference between the control signal for the current iteration of the control routine and the integral feedback contribution value of the preceding iteration of the control routine multiplied by a factor dependent on a reset time and a controller execution period.

**[0025]** Implementing multiple iterations of a control routine to generate a control signal may include, during each iteration of the control routine, generating the control signal based on a setpoint value, a most recent measurement of a process variable and the integral feedback contribution.

**[0026]** Implementing multiple iterations of a control routine to generate a control signal may include, during each iteration of the control routine, generating a proportional component from a setpoint value, a most recently received process response indication, and a proportional gain value, and generating the control signal using a difference between the proportional component generated during the current iteration and the proportional component generated during a previous iteration.

**[0027]** Another aspect provides a process controller that generates a control signal for controlling a process variable during each of a multiplicity of controller iterations of the process controller, comprising: a communications unit that receives a new value of the process variable at less than each of the multiplicity of iterations of the process controller; a proportional control component that generates a proportional control signal value during each of the iterations of the process controller including, a first summer that determines a difference between a set point value for the process variable and a received value of a process variable, and a proportional gain unit coupled to the summer; an integral control component that generates an integral control signal value during each of the iterations of the process controller including, a continuously updated iterative filter that determines a preliminary integral control component during each iteration of the process controller based on a previous value of the preliminary integral control component generated during a previous iteration of the process controller and based on a summed control signal produced by summing the control signal generated during each iteration of the process controller since the last iteration of the process controller at which a new value of the process variable was received; a switch coupled to the continuously updated iterative filter that receives the preliminary integral control component and a further value, wherein the switch operates to provide the preliminary integral control component generated by the continuously updated iterative filter during process controller iterations associated with the receipt of a new value of the process variable at the communications unit as the integral control signal value and that provides the further value as the integral control signal value during controller iterations that are not associated with the receipt of a new value of the process variable at the communications unit; and a second summer that sums the proportional control signal value and the integral control signal value during each process controller iteration to produce the control signal.

**[0028]** The summed control signal may be reset upon receipt of a new value of the process variable at the communications unit.

**[0029]** The summed control signal may be reset to zero upon receipt of a new value of the process variable at the communications unit.

**[0030]** The further value may be zero.

**[0031]** The proportional control component may include a difference unit coupled to the proportional gain unit.

**[0032]** The difference unit may be coupled between the proportional gain unit and the second summer and may determine a difference value between the output of the proportional gain unit from a previous controller iteration and the output of the proportional gain unit at the current controller iteration and may provide the difference value as the proportional control signal value to the second summer.

**[0033]** The process controller may further include a control signal conversion unit coupled to the second summer that converts the control signal to an output control signal to be sent to control a device within a process.

**[0034]** The control signal may represent a change in the output control signal to be sent to the process.

**[0035]** The process controller may further include a derivative control component that determines a derivative control signal value, the derivative control component including a derivative gain unit coupled to the first summer, and a second difference unit coupled to the derivative gain unit that determines a further difference value between the output of the derivative gain unit from a previous controller iteration and the output of the derivative gain unit at the current controller iteration and provides the further difference to the second summer as the derivative control signal value, and wherein the second summer sums the derivative control signal value with the integral control signal value and the proportional control signal value to produce the control signal.

**[0036]** The further value may be an integral control signal value output by the switch in a previous controller iteration.

**[0037]** Another aspect provides a method of generating a velocity based process control signal during each of a multiplicity of iterations of a control routine, comprising: receiving, via a computer processing device, a new value of the process variable being controlled at less than each the multiplicity of iterations of the control routine; generating, using a computer processing device, during each of the multiplicity of iterations of the control routine, a difference based integral feedback contribution for use in producing the control signal, including summing the control signal generated during each iteration of the control routine since the last iteration of the control routine at which a new process variable value was received to produce a summed control signal, providing the summed control signal to an iterative filter, and using, during each of the multiplicity of iterations, the iterative filter to determine a current integral feedback contribution value for the current iteration of the control routine from an integral feedback contribution value of a preceding iteration of the control routine and the summed control signal; and using the current integral feedback contribution value to generate the control signal for the current iteration of the control routine during each iteration at which a new process variable value is received, and not using the current integral

feedback contribution value to generate the control signal during iterations at which a new process variable value is not received from the process; and using the control signal to control the process variable.

**[0038]** The method may further include generating, using a computer processing device, a difference based proportional contribution during each of the iterations of the control routine and using the proportional contribution during each of the iterations of the control routine to generate the control signal.

**[0039]** The new process variable value may be a measurement of a process parameter being affected by the control signal.

**[0040]** Determining the integral feedback contribution may include generating the current integral feedback contribution value based on the difference between the summed control signal for the current iteration of the control routine and the integral feedback contribution value of the preceding iteration of the control routine multiplied by a factor dependent on a reset time and a controller execution period.

**[0041]** Not using the current integral feedback contribution value to generate the control signal during iterations at which a new process variable value is not received from the process may include using a fixed value as the current integral feedback contribution to generate the control signal during iterations at which a new process variable value is not received from the process.

**[0042]** The fixed value may be zero.

**[0043]** Another aspect provides a velocity based process controller that generates a control signal for controlling a process variable during each of a multiplicity of controller iterations of the process controller, comprising: a communications unit that receives a new value of the process variable at less than each of the multiplicity of iterations of the process controller; an integral control component that generates an integral control signal value during each of the iterations of the process controller including, a continuously updated iterative filter that determines a preliminary integral control component during each iteration of the process controller based on a previous value of the preliminary integral control component generated during a previous iteration of the process controller and based on a summed control signal produced by summing the control signal generated during each iteration of the process controller since the last iteration of the process controller at which a new value of the process variable was received; and a switch coupled to the continuously updated iterative filter that receives the preliminary integral control component and a further value, wherein the switch operates to provide the preliminary integral control component generated by the continuously updated iterative filter during process controller iterations associated with the receipt of a new value of the process variable at the communications unit as the integral control signal value and that provides the further value as the integral control signal value during controller iterations that are not associated with the receipt of a new value of the process variable at the communications unit; and a control signal generator coupled to receive the integral control signal value to generate a control signal for use in controlling the process during each of the controller iterations.

**[0044]** The velocity based process controller may further comprise a proportional control component that generates a proportional control signal value during each of the iterations of the process controller including, a first summer that determines a difference between a set point value for the process variable and a received value of the process variable, and a proportional gain unit coupled to the summer;

and wherein the control signal generator includes a second summer that sums the proportional control signal value and the integral control signal value during each process controller iteration to produce the control signal.

**[0045]** The proportional control component may include a difference unit coupled to the proportional gain unit.

**[0046]** The difference unit may be coupled between the proportional gain unit and the second summer and may determine a difference value between the output of the proportional gain unit from a previous controller iteration and the output of the proportional gain unit at the current controller iteration and may provide the difference value as the proportional control signal value to the second summer.

**[0047]** The process controller may further include a derivative control component that determines a derivative control signal value, the derivative control component including a derivative gain unit coupled to the first summer, and a second difference unit coupled to the derivative gain unit that determines a further difference value between the output of the derivative gain unit from a previous controller iteration and the output of the derivative gain unit at the current controller iteration and provides the further difference to the second summer as the derivative control signal value, and wherein the second summer sums the derivative control signal value with the integral control signal value and the proportional control signal value to produce the control signal.

**[0048]** The summed control signal is reset upon receipt of a new value of the process variable at the communications unit.

**[0049]** The summed control signal is reset to zero upon receipt of a new value of the process variable at the communications unit.

**[0050]** The further value may be zero.

**[0051]** The velocity based process controller may further include a control signal conversion unit coupled to the control signal generator that converts the control signal produced by the control signal generator to an output control signal to be sent to control a device within a process.

**[0052]** The control signal may represent a change in the output control signal to be sent to the process.

**[0053]** Embodiments are described with reference to the accompanying drawings, by way of example only, in which:

**[0054]** Fig. 1 is a block diagram of an example, periodically updated, hard-wired process control system;

**[0055]** Fig. 2 is graph illustrating a process output response to a process input for an example periodically updated, hard-wired process control system including the process response time;

**[0056]** Fig. 3 is a block diagram illustrating an example of a wireless process control system having a controller that receives slow or non-periodic feedback input;

**[0057]** Fig. 4A is a block diagram of a controller that allows for robust compensation of setpoint changes or feed-forward disturbances in a non-periodically updated, wireless process control system;

**[0058]** Fig. 4B is a graph illustrating the process output response of an example controller of Fig. 4A as the controller responds to several setpoint changes;

**[0059]** Fig. 5 is a block diagram of a controller that performs setpoint change compensation in a non-periodically updated process control system in which the controller compensates for a process and/or measurement delay in the feedback signal;

**[0060]** Fig. 6 is a block diagram of a controller that performs setpoint change compensation in a non-periodically updated process control system in which the process controller uses a derivative, or a rate, contribution to determine a control signal;

**[0061]** Fig. 7 is a block diagram of a controller that performs setpoint change compensation in a non-periodically updated process control system in which the process controller receives additional controller-input data provided from a field device, a control element, or other downstream device to affect a response in the operation of the process;

**[0062]** Fig. 8 is a block diagram of a controller that performs setpoint change compensation in a non-periodically updated process control system in which the process controller accommodates the use of either actual or implied controller-input data for a field device;

**[0063]** Fig. 9 is a block diagram of an example velocity based PI controller that allows for robust compensation of setpoint changes or feed-forward disturbances in a process control system in response to slowly received process variable measurement signals;

**[0064]** Fig. 10 is a block diagram of an example velocity based PID controller that allows for robust compensation of setpoint changes or feed-forward disturbances in a process control system in response to slowly received process variable measurement signals;

**[0065]** Fig. 11A is a graph illustrating the simulated process response of an example prior art velocity based PID controller in both a wired and a wireless configuration in response to a setpoint change in the primary control variable and with a process response time of 8 seconds;

**[0066]** Fig. 11B is a graph illustrating the simulated process response of an example velocity based PID controller according to the present invention in both a wired and a wireless configuration in response to a setpoint change in the primary control variable and with a process response time of 8 seconds;

**[0067]** Fig. 12A is a graph illustrating the simulated process response of an example prior art velocity based PID controller in both a wired and a wireless configuration in response to a setpoint change in the primary control variable and with a process response time of 3 seconds;

**[0068]** Fig. 12B is a graph illustrating the simulated process response of an example velocity based PID controller according to the present invention in both a wired and a wireless configuration in response to a setpoint change in the primary control variable and with a process response time of 3 seconds;

**[0069]** Fig. 13A is a graph illustrating the simulated process response of an example prior art velocity based PID controller in both a wired and a wireless configuration in response to a disturbance change and with a process response time of 8 seconds;

**[0070]** Fig. 13B is a graph illustrating the simulated process response of an example velocity based PID controller according to the present invention in both a wired and a wireless configuration in response to a disturbance change and with a process response time of 8 seconds;

**[0071]** Fig. 14A is a graph illustrating the simulated process response of an example prior art velocity based PID controller in both a wired and a wireless configuration in response to a disturbance change and with a process response time of 3 seconds; and

**[0072]** Fig. 14B is a graph illustrating the simulated process response of an example velocity based PID controller according to the present invention in both a wired and a wireless configuration in response to a disturbance change and with a process response time of 3 seconds.

**[0073]** A control technique is usable to perform control in a process loop in which process measurement feedback signals are slow or intermittently received and is particularly useful when the process measurement feedback signals are received at a rate that is slower than, is similar to, or only slightly greater than the rate associated with the process dynamics being controlled, such as the inverse of the process response time. This controller may be used, for example, in controllers that receive process measurement signals as feedback signals in a slow and/or non-periodic manner and in particular at a rate that is less than, on the same order of magnitude or otherwise similar to the process response rate of the process dynamics being controlled (i.e., the inverse of the process response time for the process variable being controlled). In one case, the control technique produces a control signal for use in controlling a process device, such as a valve, by combining a proportional contribution signal with one or more of a derivative contribution signal and an integral contribution signal. In one case, in which a velocity based PID algorithm is used, the controller produces a proportional contribution value from a difference between a setpoint and a most recently received process variable measurement feedback signal. This error signal is then multiplied by a gain signal and provided to a difference unit that determines the change in this signal since the last execution cycle of the controller. Moreover, a derivative unit may receive the error signal and performs a derivative calculation on the error signal essentially taking a derivative of the error signal over time since the last measurement signal has been received at the controller. The output of the derivative module or calculation is also provided to a change detection unit or a difference unit that determines the difference between the current output of the derivative calculation and the previous value used in the calculation of a control signal. Likewise, an integral calculation unit receives and integrates (e.g., sums) the difference of the control signal as produced by the proportional and derivative units. The output of the summer is provided to a filter that operates to filter the summer output to produce an integral contribution signal. However this integral contribution is provided to a summer that sums this contribution with the output of the control signal only when a new feedback values is provided to the controller. That is, the integral contribution is set to zero for all controller iterations except those at which a new feedback signal is available to the controller.

**[0074]** Generally speaking, the continuously updated filter of the integral contribution unit within the controller generates an indication of an expected process response (also called a feedback contribution) during each control routine iteration of the controller, despite receiving process variable measurement value updates slowly or non-periodically from field devices. The continuously updated filter uses, in part, a previously generated indication of an expected response from the last control routine iteration and the control routine execution period to generate the indication of an expected response during each control routine iteration. In addition, an integral output switch within the controller provides an output of the continuously updated filter as a feedback contribution, such as an integral (also known as reset) contribution, based on the newest measurement indication, to the control signal. Generally speaking, the integral output switch provides an expected process response as generated by the continuously updated filter at the time that the last measurement value update was received by the controller or a zero value in a velocity form of the PID controller as the integral or reset contribution to the control signal during each iteration of the controller at which a new measurement signal is not available. When a new measurement value update is available, the integral output switch clamps onto a new indication of the expected process response generated by the continuously updated filter (based on an indication of the new measurement value update), and provides the new expected process response as the integral contribution of the control signal. As a result, the controller using the continuously updated filter determines a new expected response of the process during each controller iteration, wherein each new expected process response reflects the impact of a setpoint change or a feed-forward change that was made in the time between measurement updates and that affects the controller output during development of the control signal, even though the integral or reset component of control signal is changed only when a new measurement value is available at the controller.

**[0075]** A process control system 10 is illustrated in Fig. 1 that may be used to implement the control methodologies described. The process control system 10 includes a process controller 11 connected to a data historian 12 and to one or more host workstations or computers 13 (which may be any type of personal computers, workstations, etc.), each having a display screen 14. The controller 11 is also connected to field devices 15-22 via input/output (I/O) cards 26 and 28. The data historian 12 may be any desired type of data collection unit having any desired type of memory and any desired or known software, hardware or firmware for storing data. The controller 11 is, in Fig. 1, communicatively connected to the field devices 15-22 using a hardwired communication network and communication scheme.

**[0076]** Generally, the field devices 15-22 may be any types of devices, such as sensors, valves, transmitters, positioners, etc., while the I/O cards 26 and 28 may be any types of I/O devices conforming to any desired communication or controller protocol. The controller 11 includes a processor 23 that implements or oversees one or more process control routines (or any module, block, or sub-routine thereof) stored in a memory 24. Generally speaking, the controller 11 communicates with the devices 15-22, the host computers 13 and the data historian 12 to control a process in any desired manner. Moreover, the controller 11 implements a control strategy or scheme using what are commonly referred to as function blocks, wherein each function block is an object or other part (e.g., a subroutine) of an overall control routine that operates in conjunction with other function blocks (via communications called links) to implement process control loops within the process control system 10. Function blocks typically perform

one of an input function, such as that associated with a transmitter, a sensor or other process parameter measurement device, a control function, such as that associated with a control routine that performs PID, fuzzy logic, etc. control, or an output function which controls the operation of some device, such as a valve, to perform some physical function within the process control system 10. Of course, hybrid and other types of function blocks exist and may be utilized herein. The function blocks may be stored in and executed by the controller 11 or other devices as described below.

**[0077]** As illustrated by the exploded block 30 of Fig. 1, the controller 11 may include a number of single-loop control routines, illustrated as control routines 32 and 34, and, if desired, may implement one or more advanced control loops, illustrated as a control loop 36. Each such control loop is typically referred to as a control module. The single-loop control routines 32 and 34 are illustrated as performing single loop control using a single-input/single-output fuzzy logic control block and a single-input/single-output PID control block, respectively, connected to appropriate analog input (AI) and analog output (AO) function blocks, which may be associated with process control devices such as valves, with measurement devices such as temperature and pressure transmitters, or with any other device within the process control system 10. The advanced control loop 36 is illustrated as including an advanced control block 38 having inputs communicatively connected to one or more AI function blocks and outputs communicatively connected to one or more AO function blocks, although the inputs and outputs of the advanced control block 38 may be connected to any other desired function blocks or control elements to receive other types of inputs and to provide other types of control outputs. The advanced control block 38 may implement any type of multiple-input, multiple-output control scheme, and may constitute or include a model predictive control (MPC) block, a neural network modeling or control block, a multi-variable fuzzy logic control block, a real-time-optimizer block, etc. It will be understood that the function blocks illustrated in Fig. 1, including the advanced control block 38, can be executed by the stand-alone controller 11 or, alternatively, can be located in and executed by any other processing device or control element of the process control system, such as one of the workstations 13 or one of the field devices 19-22. As an example, the field devices 21 and 22, which may be a transmitter and a valve, respectively, may execute control elements for implementing a control routine and, as such, include processing and other components for executing parts of the control routine, such as one or more function blocks. More specifically, the field device 21 may have a memory 39A for storing logic and data associated with an analog input block, while the field device 22 may include an actuator having a memory 39B for storing logic and data associated with a PID or other control block in communication with an analog output (AO) block, as illustrated in Fig. 1.

**[0078]** The graph of Fig. 2 generally illustrates a process output developed in response to a process input for a process control system based on the implementation of one or more of the control loops 32, 34 and 36 (and/or any control loop incorporating the function blocks residing in the field devices 21 and 22 or other devices). The control routine being implemented generally executes in a periodic manner over a number of controller iterations with the times of the control routine execution being indicated in Fig. 2 along the time axis by the thick arrows 40. In a conventional case, each control routine iteration 40 is supported by an updated process measurement indicated by the thin arrows 42 provided by, for instance, a transmitter or other field device. As illustrated in Fig. 2, there are typically multiple periodic process measurements 42 made and received by the control routine between each of the periodic control routine

execution times 40. To avoid the restrictions associated with synchronizing the measurement value with control execution, many known process control systems (or control loops) are designed to over-sample the process variable measurement by a factor of 2-10 times. Such over-sampling helps to ensure that the process variable measurement is current for use in the control scheme during each control routine execution or iteration. Also, to minimize control variation, conventional designs specify that feedback based control should be executed 4-10 times faster than the process response time and that a new process variable measurement be available at each controller iteration. The process response time is depicted in a process output response curve 43 of the graph of Fig. 2 as being the time associated with a process time constant ( $\tau$ ) (e.g., 63% of the process variable change) plus a process delay or deadtime ( $T_D$ ) after an implementation of a step change 44 in a process input (shown in the lower line 45 of Fig. 2). In any event, to satisfy these conventional design requirements, the process measurement value updates (indicated by the arrows 42 of Fig. 2) have been sampled and provided to the controller at a much faster rate than the control routine execution rate (indicated by the arrows 40 of Fig. 2), which in turn is much faster or higher than the process response time.

**[0079]** However, obtaining frequent and periodic measurement samples from the process may not be practical or even possible when a controller is operating in process control environment in which, for example, the controller receives measurements wirelessly from one or more field devices. In particular, in these cases, the controller may only be able to receive slow process variable measurements (to conserve battery life of the wireless sensors/transmitters) or non-periodic process variable measurements. Moreover, in these cases, the time between the non-periodic or even periodic process variable measurements may be greater than the control routine execution rate (indicated by the arrows 40 of Fig. 2). Fig. 3 depicts an exemplary wireless process control system 10 that may implement the use of slow and/or non-periodic wireless communications of process control data or process variable measurements at a controller 11.

**[0080]** The control system 10 of Fig. 3 is similar in nature to the control system 10 of Fig. 1, with like elements being numbered the same. However, the control system 10 of Fig. 3 includes a number of field devices 60-64 and 71 which are wirelessly communicatively coupled to the controller 11 and potentially to one another. As illustrated in Fig. 3, the wirelessly connected field device 60 is connected to an antenna 65 and cooperates to communicate wirelessly with an antenna 74 which is, in turn, coupled to a wireless I/O device 68. Moreover, the field devices 61-64 are connected to a wired-to-wireless conversion unit 66 which is, in turn, connected to an antenna 67. The field devices 61-64 communicate wirelessly through the antenna 67 with an antenna 73 connected to a further wireless I/O device 70. As also illustrated in Fig. 3, the field device 71 includes an antenna 72 which communicates with one or both of the antennas 73 and 74 to thereby communicate with the I/O devices 68 and/or 70. The I/O devices 68 and 70 are, in turn, communicatively connected to the controller 11 via a wired backplane connection (not shown in Fig. 3). In this case, the field devices 15-22 remain hardwired to the controller 11 via the I/O devices 26 and 28.

**[0081]** The process control system 10 of Fig. 3 generally uses the wireless transmission of data measured, sensed by or computed by the transmitters 60-64 or other control elements, such as the field device 71, as described below. In the control system 10 of Fig. 3, it will be assumed that new process

variable measurements or other signal values are transmitted to the controller 11 by the devices 60-64 and 71 on a slow or non-periodic basis, such as when certain conditions are satisfied. For example, a new process variable measurement value may be sent to the controller 11 when the process variable value changes by a predetermined amount with respect to the last process variable measurement value sent by the device to the controller 11. These signals may also be sent periodically, but at a much slower rate than is normal for typical process control systems, such as wired process control signals. For example, the slow periodic feedback rate may be less than the controller execution rate (the rate at which the controller generates new control signals for use in creating a control signal) and may, using the control techniques described herein, be at a rate that is less than, on par with, or similar to the process response rate or response time, such as a rate that is less than 2-4 times the response rate of the process dynamic being controlled. Here, the process response rate is the inverse of the process response time. Of course, other manners of determining when to send process variable measurement values in a periodic or non-periodic manner may be implemented as well or instead.

**[0082]** As will be understood, each of the transmitters 60-64 of Fig. 3 may transmit a signal indicative of a respective process variable (e.g., a flow, a pressure, a temperature or a level signal) to the controller 11 for use in one or more control loops or routines or for use in a monitoring routine. Other wireless devices, such as the field device 71, may receive process control signals wirelessly, and/or be configured to transmit other signals indicative of any other process parameter. Generally speaking, as illustrated in Fig. 3, the controller 11 includes a communications stack 80 that executes on a processor to process the incoming signals, a module or a routine 82 that executes on a processor to detect when an incoming signal includes a measurement update, and one or more control modules 84 which execute on a processor to perform control based on the measurement updates. The detection routine 82 may generate a flag or other signal to denote that data being provided via the communications stack 80 includes a new process variable measurement or other type of update. The new data and the update flag may then be provided to one or more of the control modules 84 (which may be function blocks) which are then executed by the controller 11 at a predetermined periodic execution rate, as described in further detail below. Alternatively, or in addition, the new data and the update flags may be provided to one or more monitoring modules or applications executed in the controller 11 or elsewhere in the control system 10.

**[0083]** The wireless (or other) transmitters of Fig. 3 generally result in slow or non-periodic, which includes irregular or otherwise less frequent data transmissions, between the field devices 60-64 and 71 and the controller 11. As noted above, however, the communication of measurement values from the field devices 15-22 to the controller 11 has traditionally been structured to be performed in a periodic manner at a rate much faster than the execution rate of the controller or at least much faster than the dynamic rate of the process, i.e., the inverse of the process response time (for the phenomena of the process being controlled). As a result, the control routines in the controller 11 are generally designed for periodic updates of the process variable measurement values used in the feedback loops of the controller 11.

**[0084]** To accommodate the slow, non-periodic, or otherwise unavailable measurement updates (and other unavailable communication transmissions) introduced by, for example, the wireless

communications between some of the field devices and the controller 11, the control and monitoring routine(s) of the controller 11 may be restructured or modified as described below to enable the process control system 10 to function properly when using slow including non-periodic or non-periodic or intermittent updates, and especially when these updates occur less frequently than the execution rate of the controller 11 and even when those updates are received at a rate that is similar to (e.g., less than 2-4 times or the same order of magnitude as, etc.) the process response rate (e.g., the inverse of the process response time of the process variable being controlled). Exemplary control schemes configured to operate using slow and/or non-periodic control-related communications is illustrated in more detail in Figs. 4-10. For example, Fig. 4A schematically illustrates a positional type of process controller 100 coupled to a process 101. The control scheme implemented by the controller 100 (which may be the controller 11 of Figs. 1 and 3 or a control element of a field device, e.g., one of the wireless field devices of Fig. 3, etc.) generally includes the functionality of the communications stack 80, the update detection module 82 and one or more of the control modules 84 illustrated and described in connection with Fig. 3, and produces a control signal the indicates the position to which a moveable element of a control device is to move.

**[0085]** In the exemplary system of Fig. 4A, the controller 100 receives a setpoint signal from, for example, one of the workstations 13 (Figs. 1 and 3) or from any other source within or in communication with the process control system 10 and operates to generate one or more control signals 105 which are provided from an output of the controller 100 to the process 101. Besides receiving the control signal 105, the process 101 may be subjected to measured or unmeasured disturbances schematically shown by the arrow 104. Depending on the type of process control application, the setpoint signal may be changed at any time during control of the process 101, such as by a user, a tuning routine, etc. Of course, the process control signals 105 may control an actuator associated with a valve or may control any other field device to affect a response in the operation of the process 101. The response of the process 101 to changes in the process control signals 105 is measured or sensed by a transmitter, sensor or other field device 106, which may, for example, correspond to any one of the transmitters 60-64 illustrated in Fig. 3. The communication link between the transmitter 106 and the controller 100 may include a wireless connection and is illustrated in Fig. 4A using a dashed line.

**[0086]** In a simple version, the controller 100 may implement a single/input, single/output closed-loop control routine, such as a PI control routine, which is one form of a PID control routine. Accordingly, the controller 100 includes several standard PI controller elements, including a communications stack 80, and a control signal generation unit including a summing block 108, a proportional gain element 110, a further summing block 112 and a high-low limiter 114. The control routine 100 also includes a direct feedback path including a filter 116 and an integral output switch that includes a selection block 118. The filter 116 is coupled to the output of the high-low limiter 114 and the block 118 of the switch is coupled to the output of the filter 116 and provides the integral or reset contribution or component of the control signal being generated by the controller 100 to the summing block 112.

**[0087]** During operation of the controller 100, the summing block 108 compares the setpoint signal with the most recently received process variable measurement value provided from the communications stack 80 within the controller 100 to produce an error signal. The proportional gain

element 110 operates on the error signal by, for example, multiplying the error signal by a proportional gain value  $K_p$  to produce a proportional contribution or component of the control signal. The summing block 112 then combines the output of the gain element 110 (i.e., the proportional contribution) with the integral or reset contribution or component of the control signal produced by the feedback path to produce a control signal that is not limited in nature. The limiter block 114 then performs high-low limiting on the output of the summer 112 to produce the control signal 105 to be sent to control the process 101.

**[0088]** Importantly, the filter 116 and the block or switch 118 within the feedback path of the controller 100 operate to produce the integral or reset contribution component of the control signal in the following manner. The filter 116, which is coupled to receive the output of the limiter 114, produces an indication of the expected process response to the control signal 105 based on the output value of the limiter 114 and the execution period or time of the control algorithm 100. The filter 116 provides this expected process response signal to the switch or block 118. The switch or block 118 samples and clamps the output of the filter 116 at the output of the switch or block 118 whenever a new process variable measurement value has been received and maintains this value until the next process variable output is received at the communications stack 80. As such, the output of the switch 118 remains the output of the filter 116 that was sampled at the last measurement update.

**[0089]** The expected process response to changes in the output of the summer 108, as produced by the filter 116, may be approximated using a first order model as described in more detail below. More generally, however, the expected process response may be produced using any appropriate model of the process 101, and is not limited to a model incorporated in a feedback path of the controller 100, or to a filter or a model associated with determining an integral or reset contribution for a control signal. For example, controllers utilizing a model to provide the expected process response may incorporate a derivative contribution such that the control routine 100 implements a PID control scheme. Several examples that incorporate exemplary types of derivative contributions are described below in connection with Figs. 6-8.

**[0090]** Prior to discussing the operation of the filter 116 of Fig. 4A in more detail, it is useful to note that a traditional PI controller may be implemented using a positive feedback network to determine the integral or reset contribution. Mathematically, it can be shown that the transfer function for a traditional PI implementation is equivalent to the standard formulation for unconstrained control, i.e., where the output is not limited. In particular:

$$\frac{O(s)}{E(s)} = K_p \left( 1 + \frac{1}{sT_{Reset}} \right)$$

where  $K_p$  = Proportional Gain

$T_{Reset}$  = Reset, seconds

$O(s)$  = Control Output

$E(s)$  = Control Error

One advantage of using the positive feedback path within the controller 100 as illustrated in Fig. 4A is that the reset contribution is automatically prevented from winding up when the controller output is high or low limited, i.e., by the limiter 114.

**[0091]** In any event, the control technique described below enables using the positive feedback path for determining the reset or integral contribution when the controller receives non-periodic updates of the process variable, while still enabling a robust controller response in the event of setpoint changes or feed-forward changes that occur between the receipt of new process variable measurements. Specifically, to provide robust setpoint change controller operation, the filter 116 is configured to calculate a new indication or value of an expected process response during each or every execution of the controller 100 regardless of whether this output of the filter is ever provided to the summing block 112. As a result, the output of the filter 116 is regenerated anew during each execution cycle of the controller routine, even though only the output of the filter 116 generated immediately after the controller 100 receives a new process measurement update from communications stack 80 is used as the integral or reset contribution in the summer 112.

**[0092]** In particular, the new indication of the expected response as produced by the filter 116 is calculated during each controller execution cycle from the current controller output (i.e., the control signal after the limiter 114), the indication of an expected response produced by the filter 116 produced during the last (i.e., immediately preceding) controller execution cycle, and the controller execution period. As a result, the filter 116 is described herein as being continuously updated because it is executed to produce a new process response estimation during each controller execution cycle. An example equation that may be implemented by the continuously updated filter 116 to produce a new expected process response or filter during each controller execution cycle is set forth below:

$$F_N = F_{N-1} + (O_{N-1} - F_{N-1}) * \left(1 - e^{\frac{-\Delta T}{T_{Reset}}}\right)$$

where  $F_N$  = New filter output

$F_{N-1}$  = Filter output last execution

$O_{N-1}$  = Controller output last execution

$\Delta T$  = Controller execution period

**[0093]** Here, it will be noted that the new filter output  $F_N$  is iteratively determined as the most previous filter output  $F_{N-1}$  (i.e., the current filter output value) plus a decaying component determined as the difference between the current controller output value  $O_{N-1}$  and the current filter output value  $F_{N-1}$  multiplied by a factor dependent on the reset time  $T_{Reset}$  and the controller execution period  $\Delta T$ . Using a filter that updates continuously in this manner, the control routine 100 is better able to determine the expected process response when calculating the integral control signal input when a new process variable measurement is received, thereby being more reactive to a changes in the setpoint or other feed-forward disturbances that occur between the receipt of two process variable measurements. More particularly, it

will be noted that a change in the setpoint (without the receipt of a new process measurement value) will immediately result in a change in the error signal at the output of the summer 108 which changes the proportional contribution component of the control signal and thus changes the control signal. As a result, the filter 116 will immediately begin producing a new expected response of the process to the changed control signal and will thus update its output prior to the controller 100 receiving a new process measurement value. Then, when the controller 100 receives a new process measurement value, and a sample of the filter output is clamped to the input of the summer 112 by the switch 118 to be used as the integral or reset contribution component of the control signal, the filter 116 has iterated to an expected process response that, to some degree at least, has reacted to or incorporated the response of the process 101 to the change in the setpoint.

**[0094]** In the past, such as in the systems described in U.S. Patent Nos. 7,587,252 and 7,620,460, the reset contribution filter used in a feedback path of a non-periodically updated controller only calculated a new indication of an expected response when a new process variable measurement value was available. As a result, the reset contribution filter did not compensate for setpoint changes or feed-forward disturbances that occurred between receipt of process variable measurements because the setpoint changes or feed-forward disturbances were wholly independent from any measurement value updates. For example, if a setpoint change or feed-forward disturbance occurred between two measurement updates, the expected process response of the controller was likely to be distorted because the calculation of the new indication of the expected response was based on the time since the last measurement update and the current controller output 105. As a result, the filter 116 could not begin to account for the time changes in the process (or the control signal) that resulted from a setpoint change (or other feed-forward disturbance) that occurred between the receipt of two process variable measurement values at the controller.

**[0095]** As will be understood, however, the control routine 100 of Fig. 4A provides an expected process response by basing its calculations on non-periodic measurement values while, in addition, determining the expected response between the receipt of two measurement values to account for changes caused by a change in the setpoint (or any measured disturbance used as a feed-forward input to the controller 100). Thus, the control technique described above is able to accommodate for setpoint changes, feed-forward action on measured disturbances, etc., that may affect the expected process response and thus provide a more robust control response.

**[0096]** As will be understood, the control technique illustrated in Fig. 4A calculates an indication of an expected response via the continuously updated filter 116 (e.g. the reset contribution filter) for each execution of the control block or routine 100. Here, the controller 100 configures the continuously updated filter 116 to calculate a new indication of an expected response for each execution of the control block. However, to determine if the output of the filter 116 should be used as an input to the summing block 112, the communications stack 80 and, in some examples, the update detection module 82 (Fig. 3), process the incoming data from the transmitter 106 to generate a new value flag for the integral output switch 118 when a new process variable measurement value is received. This new value flag informs the switch 118 to sample and clamp the filter output value for this controller iteration to the input of the summer 112.

**[0097]** Regardless of whether a new value flag is communicated, the continuously updated filter 116 continues to calculate an indication of an expected response for each iteration of the control routine. This new indication of an expected response is delivered to the integral output switch 118 each execution of the control block. Depending on the presence of the new value flag, the integral output switch 118 switches between allowing the new indication of the expected response from the continuously updated filter 116 to pass through to the summing block 112 and maintaining the signal that was previously delivered to the summing block 112 during the last execution of the control block. More particularly, when a new value flag is communicated, the integral output switch 118 allows the most recently calculated indication of the expected response from the continuously updated filter 116 to pass to the summing block 112. Conversely, if the new value flag is not present, then the integral output switch 118 resends the indication of the expected response from the last control block iteration to the summing block 112. In other words, the integral output switch 118 clamps onto the new indication of the expected response each time a new value flag is communicated from the stack 80, but does not allow any newly calculated indication of the expected response to reach the summing block 112 if a new value flag is not present.

**[0098]** This control technique allows the continuously updated filter 116 to continue to model the expected process response regardless of whether a new measurement value is communicated. If the control output changes as a result of a setpoint change or a feed-forward action based on a measured disturbance, irrespective of the presence of a new value flag, the continuously updated filter 116 correctly reflects the expected process response by calculating a new indication of an expected response at each control routine iteration. However, the new indication of the expected response (i.e. the reset contribution or integration component) will only be incorporated into the controller calculations when a new value flag is communicated (via the integral output switch 118).

**[0099]** A graph 200 illustrated in Fig. 4B depicts a simulated operation of the controller 100 of Fig. 4A in driving a process output signal 202 to a steady state value as the controller 100 responds to several setpoint changes. In Fig. 4B, a process output signal 202 (illustrated as a bold line) is shown vis-à-vis a setpoint value signal 204 (illustrated as a thinner line) during wireless operation in a process control system. When a setpoint change occurs, as indicated by the arrows along the time axis at the bottom of the graph 200, the controller 102 responds by producing a control signal that drives the process output to react to the new setpoint value (i.e. steady state value). For example, as illustrated in Fig. 4B, a setpoint change occurs at the time  $T_1$  as evidenced by the setpoint value signal 204 significantly changing its magnitude from a higher value to a lower value. In response, the controller 102 drives the process variable associated with the setpoint to the new steady state or setpoint value in a smooth transient curve as exhibited by the output signal 202 between the times  $T_1$  and  $T_2$ . Similarly, in Fig. 4B, a second setpoint change occurs at the time  $T_2$  as evidenced by the magnitude of setpoint value signal 204 significantly changing from a lower value to a higher value. In response, the controller 102 controls the process variable associated with the setpoint change to the new steady state or setpoint value in a smooth transient curve as shown by the output signal 202 between the times  $T_2$  and  $T_3$ . As a result, as can be seen from Fig. 4B, the controller 100 implementing the control routine described above allows for compensation of setpoint changes in a non-periodic wireless control system in a robust manner. Because feed-forward disturbances may be measured and included in the control action, the controller 100 implementing the control routine

described above may also allow for compensation of feed-forward changes in the control output in a non-periodic wireless control system.

**[00100]** It should be noted that the simple PI controller configuration of Fig. 4A uses the output of the filter 116 directly as the reset contribution to the control signal, and in this case the reset contribution of a closed-loop control routine (e.g. the continuously updated filter equation presented above) may provide an accurate representation of the process response in determining whether the process exhibits steady-state behavior. However, other processes, such as deadtime dominant processes, may require the incorporation of additional components in the controller of Fig. 4A in order to model the expected process response. With regard to processes that are well represented by a first-order model, the process time constant may generally be used to determine the reset time for the PI (or PID) controller. More specifically, if the reset time is set equal to the process time constant, the reset contribution generally cancels out the proportional contribution such that, over time, the control routine 100 reflects the expected process response. In the example illustrated in Fig. 4A, the reset contribution may be effected by a positive feedback network having a filter with the same time constant as the process time constant. While other models may be utilized, the positive feedback network, filter, or model provides a convenient mechanism for determining the expected response of a process having a known or approximated process time constant. For those processes that require PID control, the derivative contribution, also known as rate, to the PID output may also be recomputed and updated only when a new measurement is received. In those cases, the derivative calculation may use the elapsed time since the last new measurement. Some examples of controllers that may use other controller components to control more complex processes using non-periodic receipt of process measurements, but that may use the filtering technique of Fig. 4A to provide for robust control in response to setpoint changes will be described below in conjunction with Figs. 5-8.

**[00101]** Referring now to Fig. 5, an alternative controller (or control element) 120 configured in accordance with the control technique, as described above, is similar in many respects to the controller 100 illustrated in Fig. 4A. As a result, elements common to both controllers are identified with like reference numerals. The controller 120, however, incorporates an additional element into the control routine that determines the expected process response between measurement transmissions. In this case, the process 101 may be characterized as having a considerable amount of deadtime and, as a result, a deadtime unit or a block 122 is included in the controller model for deadtime compensation. The incorporation of the deadtime unit 122 generally helps to arrive at a more accurate representation of the process response. More specifically, the deadtime unit 122 may be implemented in any desired fashion and may include or utilize methods common to Smith predictors or other known control routines. However, in this situation, the continuously updated filter 116 and the switch module 118 operate in the same manner as described above with respect to the controller 100 of Fig. 4A to provide for robust control in response to setpoint changes.

**[00102]** Fig. 6 depicts another alternative controller (or control element) 130 that differs from the controller 100 described above in Fig. 4A in that a derivative, or rate, contribution component is incorporated into the controller 130. By incorporating the derivative contribution, the control routine implemented by the controller 130 includes an additional feedback mechanism such that, in some cases, a proportional-integral-derivative (PID) control scheme is implemented.

**[00103]** The control routine or technique of Fig. 6 includes a derivative contribution that is configured in a manner similar to that described above in connection with the integral contribution of Fig. 4A to accommodate non-periodic or otherwise unavailable updates of the process measurement. The derivative contribution may be restructured to be based on the elapsed time since the last measurement update. In this manner, a spike in the derivative contribution (and the resultant output signal) is avoided. More particularly, the derivative contribution of Fig. 6 is determined by a derivative block 132 that receives the error signal from the summing block 108 in parallel with the elements dedicated to the proportional and integral contributions. While other PID configurations may also be utilized (e.g., a serial configuration), the proportional, integral and derivative contributions are combined at a summing block 134 as shown in Fig. 6.

**[00104]** To accommodate unreliable transmissions and, more generally, the unavailability of measurement updates, the derivative contribution is maintained at the last determined value until a measurement update is received, as indicated by the new value flag from the communications stack 80. This technique allows the control routine to continue with periodic execution according to the normal or established execution rate of the control routine. Upon reception of the updated measurement, the derivative block 132, as illustrated in Fig. 6, may determine the derivative contribution in accordance with the following equation:

$$O_D = K_D \cdot \frac{e_N - e_{N-1}}{\Delta T}$$

where  $e_N$  = Current error

$e_{N-1}$  = Last error

$\Delta T$  = Elapsed time since a new value was communicated

$O_D$  = Controller derivative term

$K_D$  = Derivative gain factor

**[00105]** With this technique for determining the derivative contribution, the measurement updates for the process variable (i.e., control input) can be lost for one or more execution periods without the production of output spikes. When the communication is reestablished, the term ( $e_N - e_{N-1}$ ) in the derivative contribution equation may generate the same value as that generated in the standard calculation of the derivative contribution. However, for a standard PID technique, the divisor in determining the derivative contribution is the execution period. In contrast, the control technique utilizes the elapsed time between two successfully received measurements. With an elapsed time greater than the execution period, the control technique produces a smaller derivative contribution, and reduced spiking, than the standard PID technique.

**[00106]** To facilitate the determination of the elapsed time, the communications stack 80 may provide the new value flag described above to the derivative block 132 as shown in Fig. 6. Alternative examples may include or involve detection of a new measurement, or update, based on its value. Also, the process measurement may be used in place of the error in the calculation of the proportional or derivative component. More generally, the communication stack 80 may include or incorporate any software,

hardware or firmware (or any combination thereof) to implement a communications interface with the process 101, including any field devices within the process 101, process control elements external to the controller, etc. However, in the controller 130 of Fig. 6, the continuously updated filter 116 and the switch module 118 operate the same as described above with respect to the controller 100 of Fig. 4A to provide for robust control in response to setpoint changes.

**[00107]** An actuator or other downstream element controlled by the controllers described in connection with Figs. 3, 4A, and 5-6 may still receive a control signal with sudden changes, especially after periods of no communication between the controller or control element to the downstream actuator or other element. The resultant control action may be sufficiently abrupt in some cases to impact plant operations and such abrupt changes may lead to unsuitable levels of instability.

**[00108]** The potential for abrupt control changes because of loss of communications between the controller and the downstream element may be addressed by incorporating actual downstream data in place of the controller output during the last execution period when determining the feedback contribution(s) to the control signal. Generally speaking, such actual downstream data provides a feedback indication of a response to the control signal, and thus may be measured or calculated by a downstream element (e.g., a process control module) or a device (e.g., an actuator) that receives the control signal. Such data is provided in lieu of an implied response to the control signal, such as the controller output from the last execution. As shown in Figs. 4A and 5-6, the continuously updated filter 116 receives the control signal 105 as an implied indication of the downstream response. The use of such implied data effectively assumes that the downstream element, such as an actuator, received the communications of the control signal and, thus, is responding appropriately to the control signal. The actual feedback data also differs from other response indications, such as the measurement of the process variable being controlled.

**[00109]** Fig. 7 depicts an exemplary controller 140 that receives actuator position data from a downstream device or element responsive to the control signal. The downstream device or element often corresponds with an actuator providing a measurement of actuator position. More generally, the downstream device or element may correspond with, or include, a PID control block, control selector, splitter or any other device or element controlled by the control signal. In the exemplary case shown, the actuator position data is provided as an indication of the response to the control signal. As such, the actuator position data is utilized by the controller 140 during periods of continued execution of the control routine despite the absence of measurement updates of the process variable. To this end, a continuously updated filter 116 may receive the actuator position data via a communications stack 146 that establishes an interface for incoming feedback data. In this exemplary case, the feedback data includes two indications of responses to the control signal, the actuator position and the process variable.

**[00110]** As with the previous examples, the continuously updated filter 116 is configured to accommodate situations involving the absence of measurement updates for the process variable. The continuously updated filter 116 similarly recalculates its output during such absences despite the fact that only the filter output generated after receipt of a new measurement value flag is used in the summer 112. However, upon reception of a measurement update, the continuously updated filter 116 no longer relies

upon feedback of the control signal to modify its output. Rather, the actual response data from the actuator is utilized as shown below:

$$F_N = F_{N-1} + (A_{N-1} - F_{N-1}) * \left(1 - e^{\frac{-\Delta T}{T_{Reset}}}\right)$$

where  $F_N$  = New filter output

$F_{N-1}$  = Filter output last execution

$A_{N-1}$  = Controller output last execution

$\Delta T$  = Controller execution period

$T_{Reset}$  = Reset time

**[00111]** The use of an actual indication of the response to the control signal can help improve the accuracy of the control techniques, both during periods of periodic communications and after a period of non-periodic or lost communications from the PID control element to the downstream element, e.g., an actuator. However, the transmission of the actual response indication will typically require additional communications between a field device and the controller, if implemented in different devices. Such communications may be wireless, as described above, and therefore may be susceptible to unreliable transmissions or power constraints. Other reasons may also lead to the unavailability of the feedback data.

**[00112]** As described below, the control techniques discussed herein can also address situations in which such response indications are not communicated in a periodic or timely manner. That is, application of the control techniques need not be limited to the absence of measurement updates for the process variable. Rather, the control techniques may be advantageously utilized to address situations involving the absence of other response indications, such as the position of an actuator or the output of a downstream control element. Still further, the control techniques may be utilized to address situations involving the loss, delay or other unavailability of transmissions from the controller (or control element) to the downstream element, such as a field device (e.g., actuator) or another control element (e.g., cascaded PID control, splitter, etc.).

**[00113]** The wireless or other unreliable transmission of additional data to the controller or control element (i.e., the response indication or downstream element feedback), or from the controller or control element (i.e., the control signal), provides additional potential for communication issues and/or problems. As described above, feedback from the downstream element (e.g., actuator) may be involved in determining the integral contribution (or other control parameter or contribution). In this example, the control routine relies on two feedback signals rather than the single process variable fed back in the examples described above. Moreover, if the control signal never reaches the downstream element, the process will not receive the benefit of the control scheme. Transmissions of either one of these signals may be delayed or lost and, thus, the techniques described herein address either eventuality.

**[00114]** The absence of the response indication involved in the filter or other control computations may be addressed by maintaining the indication of the expected response (or other control signal component) until an update is received.

**[00115]** When the control signal does not reach the downstream element, the response indication (i.e., feedback) from the downstream element will not be changing. In such cases, the lack of a change in value may trigger logic in the controller (or control element) to similarly maintain the indication of the expected response (or other control signal component) until a change in value is received.

**[00116]** The control techniques may also be implemented in scenarios where actual feedback data is either not desired or is unavailable. The former case may be advantageous in those situations where the simplicity of using an implied response to the control signal is beneficial. For instance, the communication of the actual feedback data may be problematic or impractical. The latter case may involve actuators or other devices not configured to provide position measurement data, as described above. Older devices may not have such capabilities.

**[00117]** To accommodate such devices, a switch or other device may be provided to allow either an implied or actual response indication to be used by the control techniques. As illustrated in Fig. 8, a controller 150 is coupled to a switch 152, which in turn receives both the implied and actual response indications. In this case, the controller 150 may be identical to any of the above-described controllers, inasmuch as the implementation of the control schemes is not dependent upon knowing the type of response indication. The switch 152 may be implemented in software, hardware, firmware, or any combination thereof. Control of the switch 152 may be independent of the controller 150 and the implementation of any control routine. Alternatively or additionally, the controller 150 may provide a control signal to configure the switch 152. Furthermore, the switch 152 may be implemented as a part of the controller itself and, in some cases, may be integrated as a part of the communications stack or other portion of the controller.

**[00118]** Practice of the control methods, system and techniques is not limited to any one particular wireless architecture or communication protocol. Suitable exemplary architectures and communication support schemes are described U.S. Patent Application Serial No. 11/156,215 entitled "Wireless Architecture and Support for Process Control Systems," which was filed on June 17, 2005, the entire disclosure of which is hereby incorporated by reference. In fact, the modifications to the control routines are well-suited for any context in which the control routine is implemented in a periodic manner, but without process variable measurement updates for each control iteration. Other exemplary contexts include where a sampled value is provided irregularly or more seldom by, for instance, an analyzer or via lab samples.

**[00119]** Practice of the control technique is not limited to use with single-input, single-output PID control routines (including PI and PD routines), but rather may be applied in a number of different multiple-input and/or multiple-output control schemes and cascaded control schemes. More generally, the control technique may also be applied in the context of any closed-loop model-based control routine involving one or more process variables, one or process inputs or other control signals, such as model predictive control (MPC).

**[00120]** Fig. 9 illustrates a further example control system using the principles described herein but that is configured to have a controller 300 in the form of a velocity based controller. In the exemplary system of Fig. 9, the controller 300 receives a setpoint signal from, for example, one of the workstations 13 (Figs. 1 and 3) or from any other source within or in communication with the process control system and operates to generate one or more control signals 305 which are provided from an output of the controller 300 to the process 301. Besides receiving the control signal 305, the process 301 may be subjected to measured or unmeasured disturbances schematically shown by the arrow 304 in Fig. 4. Depending on the type of process control application, the setpoint signal may be changed at any time during control of the process 301, such as by a user, a tuning routine, etc. Of course, the process control signals 305 may control an actuator associated with a valve or may control any other field device to affect a response in the operation of the process 301. The response of the process 301 to changes in the process control signals 305 is measured or sensed by a transmitter, sensor or other field device 306, which may, for example, correspond to any one of the transmitters 60-64 illustrated in Fig. 3. The communication link between the transmitter 306 and the controller 300 may include a wireless connection and is illustrated in Fig. 9 using a dashed line. However, this link may also be a wired communication link or other type of communication link. It will be assumed, for the purpose of discussion, that the transmitter 306, which is measuring the process variable being controlled (i.e., the controlled variable) or a proxy variable that is correlated with the controlled process variable, at a slow or intermittent update rate. This slow update rate may be periodic or non-periodic and is assumed to be on the same order of magnitude as the process response rate of the process dynamics associated with the controlled process variable. Thus, the process variable measurements are provided once per a time interval that is larger than response time of the process, once per a time interval similar to the process response time or once per a time interval that is slightly less than the process response time. Thus, in some cases, this update rate may be  $\frac{1}{2}$  to  $\frac{1}{4}$  of the process response rate (the inverse of the process response time).

**[00121]** In a simple embodiment illustrated in Fig. 9, the controller 300 may implement a single/input, single/output closed-loop control routine, such as a PI control routine, which is one form of a PID control routine. Accordingly, the controller 300 includes several standard PI controller elements, including a communications stack 380, and a control signal generation unit including a summing block 308, a proportional gain element 310, and a further summing block 312. The control routine 300 also includes a direct integral feedback path including a filter 316 and an integral output switch that includes a selection block 318. However, in this case, the PI controller of Fig. 9 is configured to perform positional or differential control calculations on the proportional and integral components of the control signal. As such, the controller 300 also includes a differential block 320 disposed in the proportional component calculation path, a summer 322 disposed in the integral component calculation path and a block 324 that uses the differentially calculated control component at the input thereto to produce the control signal 305. Generally speaking, the block 324 scales the change in the position (velocity) control signal, as generated within the controller 300, or otherwise converts this signal to an analog or digital signal that is sent to the control device to instruct the control device to move a certain amount in one direction or another over a particular period of time. This block 324 may, for example, send a pulse width modulated signal, a pulsed signal, a

digital single indicative of a turn-on time, or any other signal that indicates a magnitude of a positional change over time to the control device.

**[00122]** As illustrated in Fig. 9, the integral filter 316 is coupled to the output of the summer 322 which, in turn, is coupled to receive the output of the summer 312, while the block 318 of the switch is coupled to the output of the filter 316 and provides the integral or reset contribution or component of the control signal being generated by the controller 300 to the summing block 312.

**[00123]** During each iteration or operation of the controller 300, the summing block 308 compares the setpoint signal with the most recently received process variable measurement value provided from the communications stack 380 within the controller 300 to produce an error signal (e). The proportional gain element or block 310 operates on the error signal e by, for example, multiplying the error signal e by a proportional gain value  $K_p$  to produce a proportional contribution or component of the velocity control signal. The difference block 320 then determines the change in the proportional gain value since the last controller iteration by determining the difference between the current output of the gain block 310 and the most recent value of the gain block 310 (produced during the last or immediately preceding controller iteration). The summing block 312 then combines the output of the change unit 320 (i.e., the velocity based proportional contribution) with the integral or reset contribution or component of the control signal produced by the integral feedback path to produce a velocity control signal 326 which is provided to the output block 324.

**[00124]** Importantly, however, the summer 322, the filter 316, and the block or switch 318 within the integral feedback path of the controller 300 operate to produce the integral or reset contribution component of the control signal in the following manner. Here, the summer 322 is coupled to receive the output of the summer 312 (i.e., the velocity based control signal representing the change in the position of moveable control element) during each controller iteration and sums that value with the previous output S of the summer 322 (generated during the last iteration of the controller 300) to thereby, in effect, integrate or sum up the change in the change of the output signal over a particular period of time. The new output S of the summer 322 is provided to the integral filter 316 which produces an indication of the expected process response to the control signal 305, indicated in Fig. 9 as R. The filter 316 provides this expected process response signal R to the switch or block 318. As indicated in Fig. 9, however, the switch or block 318 samples and clamps the output of the filter 316 at the output of the switch or block 318 whenever a new process variable measurement value has been received but otherwise provides a zero (0.0) value to the summer 312 as the integral control contribution during those control iterations in which no new process variable measurement is available. As such, the output of the switch 318, which is provided as the integral control contribution to generate a new control signal during each controller iteration, is the output of the filter 316 only during controller iterations at which a new process variable measurement is available to be used by the controller 300 and is zero (0.0) otherwise. After each time that the new value flag produced by the communications stack 380 is set (indicating that a new process variable measurement value is available at the controller 300), the summer 322 sets its output to zero and begins summing over a new period of time. Thus, in effect, the summer 322 sums the change in the control output signal over each controller iteration

between process variable measurement updates and resets whenever (during the controller iteration after) a new process variable measurement update is received at the controller 300.

**[00125]** The expected process response to changes in the control signal, as produced by the filter 316, may be approximated using a first order model as described in more detail below. More generally, however, the expected process response may be produced using any appropriate model of the process 301, and is not limited to a model incorporated in a feedback path of the controller 300, or to a filter or a model associated with determining an integral or reset contribution for a control signal. For example, controllers utilizing a model to provide the expected process response may incorporate a derivative contribution such that the control routine 300 implements a PID control scheme. An example that incorporate an exemplary derivative contribution is described below in connection with Fig. 10.

**[00126]** In any event, the control technique described below enables using the positive feedback path for determining the reset or integral contribution when the controller 300 receives slow or non-periodic updates of the process variable, while still enabling a robust controller response in the event of setpoint changes or feed-forward changes that occur between the receipt of new process variable measurements. Specifically, the filter 316 is configured to calculate a new indication or value of an expected process response during each or every execution of the controller 300 regardless of whether this output of the filter 316 is ever provided to the summing block 312 as the integral component of the control signal. As a result, the output of the filter 316 is regenerated anew during each execution cycle of the controller routine, even though only the output of the filter 316 generated immediately after (or during the execution cycle at which) the controller 300 receives a new process measurement update from communications stack 380 is used as the integral or reset contribution in the summer 312.

**[00127]** In particular, the new indication of the expected response  $R$  as produced by the filter 316 is calculated during each controller execution cycle from the current summer output  $S$  (i.e., the summed change in the control signal output by the summer 312 since the last process variable measurement update), the indication of an expected response produced by the filter 316 produced during the last (i.e., immediately preceding) controller execution cycle, and the controller execution period. As a result, the filter 316 is described herein as being continuously updated because it is executed to produce a new process response estimation during each controller execution cycle. An example equation that may be implemented by the continuously updated filter 316 to produce a new expected process response or filter during each controller execution cycle is set forth below:

$$R_N = R_{N-1} + (S_{N-1} - R_{N-1}) * \left(1 - \frac{-\Delta T}{e^{T_{Reset}}}\right)$$

where  $R_N$  = New filter Output

$R_{N-1}$  = Filter output last execution

$S_{N-1}$  = Integral path summer output last execution

$\Delta T$  = Controller execution period

$e^{T_{Reset}}$  = Integral time period

**[00128]** Here, it will be noted that the new filter output  $R_N$  is iteratively determined as the most previous filter output  $R_{N-1}$  (i.e., the current filter output value) plus a decaying component determined as the difference between the summed change in the current controller output value  $S_{N-1}$  from the summer 322 and the current filter output value  $R_{N-1}$  multiplied by a factor dependent on the reset time  $T_{Reset}$  and the controller execution period  $\Delta T$ . Using a filter that updates continuously in this manner, the control routine 300 is better able to determine the expected process response when calculating the integral control signal input when a new process variable measurement is received, thereby being more reactive to changes in the setpoint or other feed-forward disturbances that occur between the receipt of two process variable measurements. However, this integral path calculation prevents the control system from winding up in the presence of slowly received or intermittent process variable feedback measurements. More particularly, it will be noted that a change in the setpoint (without the receipt of a new process measurement value) will immediately result in a change in the error signal at the output of the summer 308 which changes the proportional contribution component of the velocity control signal 326 and thus changes the control signal 305. As a result, the summer 322 will increase its output  $S$  by that amount and the filter 316 will then begin producing a new expected response of the process to the changed control signal and will thus update its output prior to the controller 300 receiving a new process measurement value. Then, when the controller 100 receives a new process measurement value, and a sample of the filter output is clamped to the input of the summer 312 by the switch 318 to be used as the integral or reset contribution component of the control signal, the filter 316 has iterated to an expected process response that, to some degree at least, has reacted to or incorporated the response of the process 301 to the change in the setpoint based on previously sent control signal 305. However, this integral value is only added into the control signal 326 when a new process measurement value is received, to enable the error signal  $e$  produced by the summer 308 to reflect changes in the process variable between times that the process variable measurements are received at the controller 300. At controller iterations between the times at which a process variable measurement is received, the integral component provided to the summer 312 is set to zero. This technique, prevents or helps the control system 300 avoid winding up. In effect, the integral component produced by the filter 316 estimates the process response between the times (controller iterations) at which subsequent process variable feedback is received and, if the actual process variable response is as expected, when a new process variable measurement is received at the controller 300, then the integral component will zero out the value produced in the proportional path. If the expected response of the process is different than the actual process response during this time, then the integral component will cause a change in the control signal 326 to force movement of the actuator to thereby correct the position of the actuator.

**[00129]** In the past, such as in the systems described in U.S. Patent Nos. 7,587,252 and 7,620,460, the reset contribution filter used in a feedback path of a non-periodically updated controller only calculated a new indication of an expected response when a new process variable measurement value was available. As a result, the reset contribution filter did not compensate for setpoint changes or feed-forward disturbances that occurred between receipt of process variable measurements because the setpoint changes or feed-forward disturbances were wholly independent from any measurement value updates. For example, if a setpoint change or feed-forward disturbance occurred between two measurement updates,

the expected process response of the controller was likely to be distorted because the calculation of the new indication of the expected response was based on the time since the last measurement update and the current controller output 305. As a result, the filter 316 could not begin to account for the time changes in the process (or the control signal) that resulted from a setpoint change (or other feed-forward disturbance) that occurred between the receipt of two process variable measurement values at the controller.

**[00130]** As will be understood, however, the control routine 300 of Fig. 9 provides an expected process response by basing the calculations being performed on slow or non-periodic measurement values while, in addition, determining the expected response between the receipt of two measurement values to account for changes caused by a change in the setpoint (or any measured disturbance used as a feed-forward input to the controller 300). Thus, the control technique described above is able to accommodate for setpoint changes, feed-forward action on measured disturbances, etc., that may affect the expected process response and thus provide a more robust control response. Moreover, this control technique, because it avoids windup in the controller, is able to operate effectively when the process variable measurement feedback rate is equal to or even less than the inverse of the process response time (i.e., when the times between feedback measurements being received at the controller is greater than the process response time).

**[00131]** As will be understood, the control technique illustrated in Fig. 9 calculates an indication of an expected response via the continuously updated filter 316 (e.g. the reset contribution filter) for each execution of the control block or routine 300. Here, the controller 300 configures the continuously updated filter 316 to calculate a new indication of an expected response for each execution of the control block. However, to determine if the output of the filter 316 should be used as an input to the summing block 312, the communications stack 380 and, in some examples, the update detection module 82 (Fig. 3), processes the incoming data from the transmitter 306 to generate a new value flag for the integral output switch 318 and for the summer 326 when a new process variable measurement value is received. This new value flag informs the switch 318 to sample and clamp the filter output value for this controller iteration to the input of the summer 312. Otherwise, the switch 318 provides a zero (0.0) value to the summer 312 as the integral contribution value.

**[00132]** Regardless of whether a new value flag is communicated, the continuously updated filter 316 continues to calculate an indication of an expected response for each iteration of the control routine. This new indication of an expected response is delivered to the integral output switch 318 each execution of the control block. Depending on the presence of the new value flag, the integral output switch 318 switches between allowing the new indication of the expected response from the continuously updated filter 316 to pass through to the summing block 312 or maintaining a zero value at the input to the summing block 312. More particularly, when a new value flag is communicated, the integral output switch 318 allows the most recently or currently calculated indication of the expected response from the continuously updated filter 316 to pass to the summing block 312. Conversely, if the new value flag is not present, then the integral output switch 318 provides a zero value to the summer 312.

**[00133]** After or when a new process variable measurement value is received at the process controller 300, and the output R of the continuous filter 316 is used in the summer 312, then the time since

the last communication is set to zero (0) and the continuous filter output R is set to zero. Likewise, the output of the summer 322 is set to zero (0). Moreover, in these situations, the summer 312 may subtract the continuous filter output R from the output of the block 320 to generate the new control signal 326 depending on the manner or order in which the block 320 performs a difference calculation.

**[0134]** This control technique allows the continuously updated filter 316 to continue to model the expected process response regardless of whether a new measurement value is communicated. If the control output changes as a result of a setpoint change or a feed-forward action based on a measured disturbance, irrespective of the presence of a new value flag, the continuously updated filter 316 reflects the expected process response by calculating a new indication of an expected response at each control routine iteration. However, the new indication of the expected response (i.e. the reset contribution or integration component) will only be incorporated into the controller output signal calculations when a new value flag is communicated (via the integral output switch 318), which prevents or lessens controller windup in response to slowly received process variable measurements at the controller 300.

**[0135]** It should be noted that the simple PI controller configuration of Fig. 9 uses the output of the filter 316 directly as the reset contribution to the control signal, and in this case the reset contribution of a closed-loop control routine (e.g. the continuously updated filter equation presented above) may provide an accurate representation of the process response in determining whether the process exhibits steady-state behavior. However, other processes, such as deadtime dominant processes, may require the incorporation of additional components in the controller of Fig. 9, such as those illustrated in Figs. 5 and 6, by including a deadtime unit in the integral calculation path, in order to model the expected process response. With regard to processes that are well represented by a first-order model, the process time constant may generally be used to determine the reset time for the PI (or PID) controller. More specifically, if the reset time is set equal to the process time constant, the reset contribution generally cancels out the proportional contribution such that, over time, the control routine 300 reflects the expected process response. In the example illustrated in Fig. 9, the reset contribution may be effected by a positive feedback network having a filter with the same time constant as the process time constant. While other models may be utilized, the positive feedback network, filter, or model provides a convenient mechanism for determining the expected response of a process having a known or approximated process time constant. For those processes that require PID control, the derivative contribution, also known as rate, to the PID output may also be recomputed and updated only when a new measurement is received. In those cases, the derivative calculation may use the elapsed time since the last new measurement. An example of a controller that may use other controller components to control more complex processes using non-periodic receipt of process measurements, but that may use the filtering technique of Fig. 9 to provide for robust control in response to setpoint changes will be described below in conjunction with Fig. 10.

**[0136]** In particular, Fig. 10 depicts an alternative controller (or control element) 400 that differs from the controller 300 described above in Fig. 9 in that a derivative, or rate, contribution component is incorporated into the controller 400. By incorporating the derivative contribution, the control routine implemented by the controller 400 includes an additional feedback mechanism such that, in some cases, a proportional-integral-derivative (PID) control scheme is implemented.

**[0137]** The control routine or technique of Fig. 10 includes a derivative contribution that is configured in a manner similar to that described above in connection with the systems of Fig. 7 and 8 to accommodate slow, non-periodic, or otherwise unavailable updates of the process variable measurement. The derivative contribution may be restructured to be based on the elapsed time since the last measurement update. In this manner, a spike in the derivative contribution (and the resultant output signal) is avoided. More particularly, the derivative contribution in the system of Fig. 10 is determined by a derivative block 432 that receives the error signal (multiplied by the proportional gain  $K_p$ ) from the gain block 310 in parallel with the elements dedicated to the proportional and integral contributions, and operates to produce a derivative control component  $O_D$  which is then provided to a change block 433. The change block 433 determines the change in the derivative control component  $O_D$  since the last controller iteration and provides this change value to a summer 434, which sums or adds this change in the derivative control component to the output of the summer 312 to produce the change in the control signal. In this case, the summer 322 in the integral contribution calculation path is connected to the output of the summer 434. However, the components of Fig. 10 also illustrated in Fig. 9 operate in the manner as described with respect to Fig. 9. Here it will be seen that the derivative block 432 only operates to calculate a new derivative component  $O_D$  only during controller iterations at which a new value flag is received therein (indicating a new value of the process variable measurement has been received at the controller). This operation, in effect, keeps the output of the change block 434 zero for all controller iterations during which or at which a new process variable measurement is not received at the controller 400.

**[0138]** To accommodate unreliable transmissions and, more generally, the unavailability of measurement updates, the derivative contribution  $O_D$  is maintained at the last determined value until a measurement update is received, as indicated by the new value flag from the communications stack 380. This technique allows the control routine to continue with periodic execution according to the normal or established execution rate of the control routine. Upon reception of the updated measurement, the derivative block 432, as illustrated in Fig. 10, may determine the derivative contribution in accordance with the following equation:

$$O_D = K_P K_D \cdot \frac{e_N - e_{N-1}}{\Delta T}$$

where  $e_N$  = Current error

$e_{N-1}$  = Last error

$\Delta T$  = Elapsed time since a new value was communicated

$O_D$  = Controller derivative term

$K_P$  = Proportional gain factor

$K_D$  = Derivative gain factor

**[0139]** Of course, if desired, the derivative component calculation block 432 may be connected directly to the output of the summer 308 to receive the error signal and the derivative gain term  $K_D$  may be set to incorporate a derivative gain with the proportional gain  $K_P$ . With this technique for determining the derivative contribution, the measurement updates for the process variable (i.e., control input) can be lost or not available for one or more execution consecutive periods without the production of output spikes, which enables bumpless recovery. When the communication is reestablished, or a new process variable measurement value is received at the controller, the term  $(e_N - e_{N-1})$  in the derivative contribution equation may generate the same value as that generated in the standard calculation of the derivative contribution. However, for a standard PID technique, the divisor in determining the derivative contribution is the execution period. In contrast, the control technique utilizes the elapsed time between two successfully received measurements. With an elapsed time greater than the execution period, the control technique produces a smaller derivative contribution, and reduced spiking, than the standard PID technique.

**[0140]** To facilitate the determination of the elapsed time, the communications stack 80 may provide the new value flag described above to the derivative block 432 as shown in Fig. 10. Alternative examples may include or involve detection of a new measurement, or update, based on its value. Also, the process measurement may be used in place of the error in the calculation of the proportional or derivative component. More generally, the communication stack 380 may include or incorporate any software, hardware or firmware (or any combination thereof) to implement a communications interface with the process 301, including any field devices within the process 301, process control elements external to the controller, etc. However, in the controller 400 of Fig. 10, the continuously updated filter 316 and the switch module 318 operate the same as described above with respect to the controller 300 of Fig. 9 to provide for robust control in response to setpoint changes.

**[0141]** Practice of the control methods, system and techniques described herein is not limited to any one particular wireless architecture or communication protocol. Suitable exemplary architectures and communication support schemes are described U.S. Patent Application Serial No. 11/156,215 entitled "Wireless Architecture and Support for Process Control Systems," which was filed on June 17, 2005, the entire disclosure of which is hereby incorporated by reference. In fact, the modifications to the control routines are well-suited for any context in which the control routine is implemented in a periodic manner, but without process variable measurement updates for each control iteration. Other exemplary contexts include where a sampled value is provided irregularly or more seldom by, for instance, an analyzer or via lab samples.

**[0142]** Practice of the control technique is not limited to use with single-input, single-output PID control routines (including PI and PD routines), but rather may be applied in a number of different multiple-input and/or multiple-output control schemes and cascaded control schemes. More generally, the control technique may also be applied in the context of any closed-loop model-based control routine involving one or more process variables, one or process inputs or other control signals, such as model predictive control (MPC).

**[0143]** Figs. 11-14 provide graphical depictions of simulated operation of the control routines described herein (in particular that of Fig. 10) as compared to a prior art controller using a standard velocity

form of a PID control algorithm to illustrate the effectiveness of the present control routine in situations in which the process response time is similar to, or even less than the time between updates of the measurement of the controlled process variable. The graphs of Figs. 11-14 illustrate simulated control examples using primary control only although other types of control could be used, such as override control. Generally speaking, each of the graphs in Figs. 11A, 12A, 13A, and 14A illustrates the operation of the standard, prior art velocity based PID control algorithm using both a wired feedback configuration (which operation is illustrated on the left side of the graph) in which a new process variable measurement value is available during each controller iteration, and a wireless configuration (which operation is illustrated on the right side of the graph). Figs. 11A and 13A, however, illustrate control situations in which the feedback rate or time between process variable measurements is 8 seconds and in which the process response time is 8 seconds, while Figs. 12A and 14A illustrate the operation of the prior art controller in which the feedback rate or time between process variable measurements is 8 seconds and in which the process response time is 3 seconds. Likewise, Figs. 11 and 12 illustrate the controller operation in response to setpoint changes while Figs. 13 and 14 illustrate the operation of these same controllers in response to a disturbance changes. For comparison, the graphs of Figs. 11B, 12B, 13B, and 14B, illustrate the operation of the velocity based PID algorithm described herein in the same process control situations as those of Figs. 11A, 12A, 13A, and 14A, respectively.

**[0144]** Generally speaking, the following parameters were used in the simulated control operations depicted in Figs. 11-14, and these tests were run for wired and wireless inputs and for changes in setpoint and unmeasured disturbances as explained above. The control and process simulation used in the test were setup as follows:

#### Test for Process Response of 8 seconds

Primary Processes (Same gain and dynamics)

Process Gain = 1

Process Time Constant = 8 seconds

Process Deadtime = 0 seconds

PID Tuning for Primary (Lambda Factor 1.0)

Proportional Gain = 1

Integral Gain = 7.5 repeats/min

#### Test for Process Response of 3 seconds

Primary Process (Same gain and dynamics)

Process Gain = 1

Process Time Constant = 3 seconds

Process Deadtime = 0 seconds

PID Tuning for Primary (Lambda Factor 1.0)

Proportional Gain = 1

Integral Gain = 20 repeats/min

#### Module Execution Rate

0.5 sec for all tests

#### Wireless communication update rate

8 sec periodic for all tests

#### Disturbance Input

Only impacts primary measurement

Gain = 1

**[0145]** As illustrated in Figs. 11A and 13A, the prior art velocity based PID control algorithm operates somewhat satisfactorily in response to setpoint changes (Fig. 11A) and for disturbance changes (Fig. 13A) in both a wired configuration and in a wireless configuration in which the process response time is equal to the interval between process variable measurements (both are set to 8 seconds). However, as indicated by the circled parts of the graphs of Figs 11A and 13A, this control technique causes the valve position to undergo significant variations during the response during wireless control. As illustrated in Figs. 11B and 13B, the current control technique described herein operates a bit better in these situations (setpoint change in Fig. 11B and disturbance change in Fig. 13B) and very similarly to the operation of the wired configuration.

**[0146]** However, as illustrated in Figs. 12A and 14A, the prior art velocity based PID controller operates very poorly and, in fact, becomes unstable, during wireless control in response to both setpoint changes and disturbance changes when the process response time is 3 seconds and the process variable update rate is 8 seconds. However, as illustrated in Figs. 12B and 14B, the current velocity based control routine still operates very satisfactorily in these situations, illustrating the effectiveness of the currently described control routine when the process variable measurement update interval time is greater than (longer than) and even significantly greater than (e.g., 2-4 times) the process response time.

**[0147]** The term "field device" is used herein in a broad sense to include a number of devices or combinations of devices (i.e., devices providing multiple functions, such as a transmitter/actuator hybrid), as well as any other device(s) that perform(s) a function in a control system. In any event, field devices may include, for example, input devices (e.g., devices such as sensors and instruments that provide status, measurement or other signals that are indicative of process control parameters such as, for example, temperature, pressure, flow rate, etc.), as well as control operators or actuators that perform actions in response to commands received from controllers and/or other field devices such as valves, switches, flow control devices, etc.

**[0148]** It should be noted that any control routines or modules described herein may have parts thereof implemented or executed in a distributed fashion across multiple devices. As a result, a control routine or module may have portions implemented by different controllers, field devices (e.g., smart field

devices) or other devices or control elements, if so desired. Likewise, the control routines or modules described herein to be implemented within the process control system may take any form, including software, firmware, hardware, etc. Any device or element involved in providing such functionality may be generally referred to herein as a “control element,” regardless of whether the software, firmware, or hardware associated therewith is disposed in a controller, field device, or any other device (or collection of devices) within the process control system. A control module may be any part or portion of a process control system including, for example, a routine, a block or any element thereof, stored on any computer readable medium. Such control modules, control routines or any portions thereof (e.g., a block) may be implemented or executed by any element or device of the process control system, referred to herein generally as a control element. Control routines, which may be modules or any part of a control procedure such as a subroutine, parts of a subroutine (such as lines of code), etc., may be implemented in any desired software format, such as using object oriented programming, using ladder logic, sequential function charts, function block diagrams, or using any other software programming language or design paradigm. Likewise, the control routines may be hard-coded into, for example, one or more EPROMs, EEPROMs, application specific integrated circuits (ASICs), or any other hardware or firmware elements. Still further, the control routines may be designed using any design tools, including graphical design tools or any other type of software/hardware/firmware programming or design tools. Thus, the controller 11 may be configured to implement a control strategy or control routine in any desired manner.

**[0149]** Alternatively or additionally, the function blocks may be stored in and implemented by the field devices themselves, or other control elements of the process control system, which may be the case with systems utilizing Fieldbus devices. While the description of the control system is provided herein using a function block control strategy, the control techniques and system may also be implemented or designed using other conventions, such as ladder logic, sequential function charts, etc. or using any other desired programming language or paradigm.

**[0150]** When implemented, any of the software described herein may be stored in any computer readable memory such as on a magnetic disk, a laser disk, or other storage medium, in a RAM or ROM of a computer or processor, etc. Likewise, this software may be delivered to a user, a process plant or an operator workstation using any known or desired delivery method including, for example, on a computer readable disk or other transportable computer storage mechanism or over a communication channel such as a telephone line, the Internet, the World Wide Web, any other local area network or wide area network, etc. (which delivery is viewed as being the same as or interchangeable with providing such software via a transportable storage medium). Furthermore, this software may be provided directly without modulation or encryption or may be modulated and/or encrypted using any suitable modulation carrier wave and/or encryption technique before being transmitted over a communication channel.

**[0151]** While the present invention has been described with reference to specific examples, which are intended to be illustrative only and not to be limiting of the invention, it may be apparent to those of ordinary skill in the art that changes, additions or deletions may be made to the control techniques without departing from the spirit and scope of the invention.

**[0152]** When used in this specification and claims, the terms "comprises" and "comprising" and variations thereof mean that the specified features, steps or integers are included. The terms are not to be interpreted to exclude the presence of other features, steps or components.

**[0153]** The features disclosed in the foregoing description, or the following claims, or the accompanying drawings, expressed in their specific forms or in terms of a means for performing the disclosed function, or a method or process for attaining the disclosed result, as appropriate, may, separately, or in any combination of such features, be utilised for realising the invention in diverse forms thereof.

21 02 22

## CLAIMS

1. A method of controlling a process, comprising:

implementing, on a computer processing device, multiple iterations of a control routine to generate a control signal including, during each iteration of the control routine:

generating, using a computer processing device, an integral feedback contribution for use in producing the control signal, including using, during each of the multiple iterations, including a current iteration, a continuously updated iterative filter to determine a current integral feedback contribution value for the current iteration of the control routine from an integral feedback contribution value of a preceding iteration of the control routine and a summed control signal produced by summing the control signal generated during each iteration of the control routine since the last iteration of the control routine at which a new process response indication was received; and

using, as determined by a switch coupled to the continuously updated filter, the current integral feedback contribution to generate the control signal for the current iteration of the control routine during each controller iteration at which a new process response indication is received, and not using, as determined by the switch, the current integral feedback contribution to generate the control signal during controller iterations at which a new process response indication to the control signal is not received; and

using the control signal to control the process.

2. The method of claim 1, further including generating a proportional contribution during each iteration and using the proportional contribution during each iteration to generate the control signal.

3. The method of claim 1 or 2, wherein the new process response indication is a measurement of a process parameter being affected by the control signal.

4. The method of claim 3, wherein the process parameter is a process variable controlled by a field device that is responsive to the control signal.

5. The method of any preceding claim, wherein determining the integral feedback contribution includes generating the integral feedback contribution value based on the difference between the control signal for the current iteration of the control routine and the integral feedback contribution value of the preceding iteration of the control routine multiplied by a factor dependent on a reset time and a controller execution period.

6. The method of any preceding claim, wherein implementing multiple iterations of a control routine to generate a control signal includes, during each iteration of the control routine, generating the control signal based on a setpoint value, a most recent measurement of a process variable and the integral feedback contribution.

7. The method of any preceding claim, wherein implementing multiple iterations of a control routine to generate a control signal includes, during each iteration of the control routine, generating a proportional component from a setpoint value, a most recently received process response indication, and a proportional gain value, and generating the control signal using a difference between the proportional component generated during the current iteration and the proportional component generated during a previous iteration.

8. A process controller that generates a control signal for controlling a process variable during each of a multiplicity of controller iterations of the process controller, comprising:

a communications unit that receives a new value of the process variable at less than each of the multiplicity of iterations of the process controller;

a proportional control component that generates a proportional control signal value during each of the iterations of the process controller including:

a first summer that determines a difference between a set point value for the process variable and a received value of a process variable, and

a proportional gain unit coupled to the summer;

an integral control component that generates an integral control signal value during each of the iterations of the process controller including:

a continuously updated iterative filter that determines a preliminary integral control component during each iteration of the process controller based on a previous value of the preliminary integral control component generated during a previous iteration of the process controller and based on a summed control signal produced by summing the control signal generated during each iteration of the process controller since the last iteration of the process controller at which a new value of the process variable was received; and

a switch coupled to the continuously updated iterative filter that receives the preliminary integral control component and a further value, wherein the switch operates to provide the preliminary integral control component generated by the continuously updated iterative filter during process controller iterations associated with the receipt of a new value of the process variable at the communications unit as the integral control signal value and that provides the further value as the integral control signal value during controller iterations that are not associated with the receipt of a new value of the process variable at the communications unit; and

a second summer that sums the proportional control signal value and the integral control signal value during each process controller iteration to produce the control signal.

9. The process controller of claim 8, wherein the summed control signal is reset upon receipt of a new value of the process variable at the communications unit.

10. The process controller of claim 9, wherein the summed control signal is reset to zero upon receipt of a new value of the process variable at the communications unit.

11. The process controller of any of claims 8 to 10, wherein the further value is zero.
12. The process controller of any of claims 8 to 11, wherein the proportional control component includes a difference unit coupled to the proportional gain unit.
13. The process controller of claim 12, wherein the difference unit is coupled between the proportional gain unit and the second summer and determines a difference value between the output of the proportional gain unit from a previous controller iteration and the output of the proportional gain unit at the current controller iteration and provides the difference value as the proportional control signal value to the second summer.
14. The process controller of any of claims 8 to 13, further including a control signal conversion unit coupled to the second summer that converts the control signal to an output control signal to be sent to control a device within a process.
15. The process controller of claim 14, wherein the control signal represents a change in the output control signal to be sent to the process.
16. The process controller of any of claims 8 to 15, further including a derivative control component that determines a derivative control signal value, the derivative control component including a derivative gain unit coupled to the first summer, and a second difference unit coupled to the derivative gain unit that determines a further difference value between the output of the derivative gain unit from a previous controller iteration and the output of the derivative gain unit at the current controller iteration and provides the further difference to the second summer as the derivative control signal value, and wherein the second summer sums the derivative control signal value with the integral control signal value and the proportional control signal value to produce the control signal.
17. The process controller of any of claims 8 to 16, wherein the further value is an integral control signal value output by the switch in a previous controller iteration.

18. A method of generating a velocity based process control signal during each of a multiplicity of iterations of a control routine, comprising:

receiving, via a computer processing device, a new value of the process variable being controlled at less than each the multiplicity of iterations of the control routine;

generating, using a computer processing device, during each of the multiplicity of iterations of the control routine, a difference based integral feedback contribution for use in producing the control signal, including summing the control signal generated during each iteration of the control routine since the last iteration of the control routine at which a new process variable value was received to produce a summed control signal, providing the summed control signal to an iterative filter, and using, during each of the multiplicity of iterations, the iterative filter to determine a current integral feedback contribution value for the

21 02 22

current iteration of the control routine from an integral feedback contribution value of a preceding iteration of the control routine and the summed control signal; and

using the current integral feedback contribution value to generate the control signal for the current iteration of the control routine during each iteration at which a new process variable value is received, and not using the current integral feedback contribution value to generate the control signal during iterations at which a new process variable value is not received from the process; and

using the control signal to control the process variable.

19. The method of generating a velocity based process control signal of claim 18, further including generating, using a computer processing device, a difference based proportional contribution during each of the iterations of the control routine and using the proportional contribution during each of the iterations of the control routine to generate the control signal.

20. The method of generating a velocity based process control signal of claim 18 or 19, wherein the new process variable value is a measurement of a process parameter being affected by the control signal.

21. The method of generating a velocity based process control signal of any of claims 18 to 20, wherein determining the integral feedback contribution includes generating the current integral feedback contribution value based on the difference between the summed control signal for the current iteration of the control routine and the integral feedback contribution value of the preceding iteration of the control routine multiplied by a factor dependent on a reset time and a controller execution period.

22. The method of generating a velocity based process control signal of any of claims 18 to 21, wherein not using the current integral feedback contribution value to generate the control signal during iterations at which a new process variable value is not received from the process includes using a fixed value as the current integral feedback contribution to generate the control signal during iterations at which a new process variable value is not received from the process.

23. The method of generating a velocity based process control signal of any of claims 18 to 22, wherein the fixed value is zero.

24. A velocity based process controller that generates a control signal for controlling a process variable during each of a multiplicity of controller iterations of the process controller, comprising:

a communications unit that receives a new value of the process variable at less than each of the multiplicity of iterations of the process controller;

an integral control component that generates an integral control signal value during each of the iterations of the process controller including:

a continuously updated iterative filter that determines a preliminary integral control component during each iteration of the process controller based on a previous value of the preliminary integral control component generated during a previous iteration of the process controller

and based on a summed control signal produced by summing the control signal generated during each iteration of the process controller since the last iteration of the process controller at which a new value of the process variable was received; and

a switch coupled to the continuously updated iterative filter that receives the preliminary integral control component and a further value, wherein the switch operates to provide the preliminary integral control component generated by the continuously updated iterative filter during process controller iterations associated with the receipt of a new value of the process variable at the communications unit as the integral control signal value and that provides the further value as the integral control signal value during controller iterations that are not associated with the receipt of a new value of the process variable at the communications unit; and

a control signal generator coupled to receive the integral control signal value to generate a control signal for use in controlling the process during each of the controller iterations.

25. The velocity based process controller of claim 24, further comprising a proportional control component that generates a proportional control signal value during each of the iterations of the process controller including,

a first summer that determines a difference between a set point value for the process variable and a received value of the process variable, and

a proportional gain unit coupled to the summer; and

wherein the control signal generator includes a second summer that sums the proportional control signal value and the integral control signal value during each process controller iteration to produce the control signal.

26. The velocity based process controller of claim 25, wherein the proportional control component includes a difference unit coupled to the proportional gain unit.

27. The velocity based process controller of claim 26, wherein the difference unit is coupled between the proportional gain unit and the second summer and determines a difference value between the output of the proportional gain unit from a previous controller iteration and the output of the proportional gain unit at the current controller iteration and provides the difference value as the proportional control signal value to the second summer.

28. The process controller of any of claims 25 to 27, further including a derivative control component that determines a derivative control signal value, the derivative control component including a derivative gain unit coupled to the first summer, and a second difference unit coupled to the derivative gain unit that determines a further difference value between the output of the derivative gain unit from a previous controller iteration and the output of the derivative gain unit at the current controller iteration and provides the further difference to the second summer as the derivative control signal value, and wherein the second summer sums the derivative control signal value with the integral control signal value and the proportional control signal value to produce the control signal.

29. The velocity based process controller of any of claims 24 to 28, wherein the summed control signal is reset upon receipt of a new value of the process variable at the communications unit.

30. The velocity based process controller of claim 29, wherein the summed control signal is reset to zero upon receipt of a new value of the process variable at the communications unit.

31. The velocity based process controller of any of claims 24 to 30, wherein the further value is zero.

32. The velocity based process controller of any of claims 24 to 31, further including a control signal conversion unit coupled to the control signal generator that converts the control signal produced by the control signal generator to an output control signal to be sent to control a device within a process.

33. The velocity based process controller of claim 32, wherein the control signal represents a change in the output control signal to be sent to the process.

21 02 22