Analog design techniques suit process-control needs

Although analog circuits are relatively inflexible, they can furnish process-control systems with operational features comparable to those attainable using digital methods. A stepper-motor pump-drive application illustrates the techniques involved.

Jim Williams, National Semiconductor Corp

For many process-control applications, analog control circuits prove a better choice than their digital counterparts, especially when you expect low product volumes and when fast design time and high noise immunity are design priorities. In fact, if you're working with well-defined operational specifications and don't anticipate having to make major modifications, analog methods serve as viable alternatives to intelligent but dedicated and expensive hardware/software approaches.

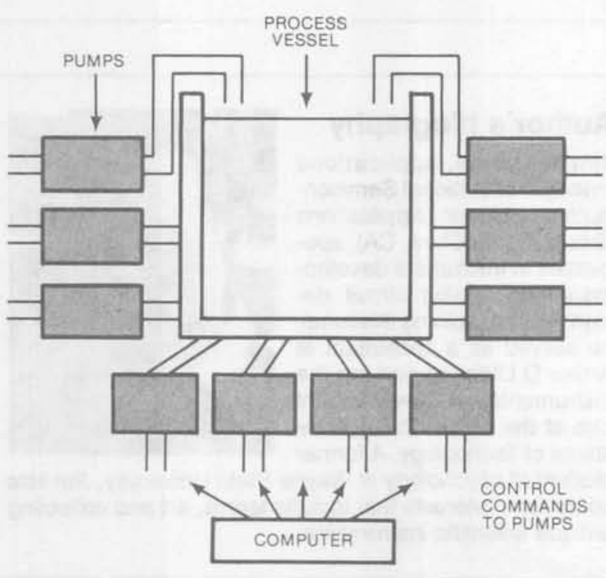


Fig 1—In this conceptual computer-controlled chemical-mixing system, *the computer governs several pumps delivering chemicals to the process mixing vessel by periodically sending updated pulse-width-modulated commands that control the pumps' speeds.*

Controlling a pump's speed

To demonstrate, this article describes the design of an analog pump controller that manipulates computer-generated command pulses to regulate stepper-motor-driven pumps in a critical chemical-mixing process. The controller/pump system furnishes precise fluid delivery at both fast and slow rates, a requirement often arising in chemical and biological process-control systems, which demand high pumping rates for flushing or process startup and slow but accurate flow rates for mixing precise amounts of liquid. Although dc motors can deliver adequate high-speed performance, they often need complex and expensive digital control to perform well at very slow speeds. In contrast, exponentially driven stepper motors can easily handle a pump's conflicting high- and low-speed drive requirements.

Fig 1 diagrams a computer-driven system that governs several pumps feeding an intricate chemical process. The computer controls each pump's speed by periodically sending a pulse-width-modulated control command. Because the computer runs in a time-shared manner, each pump controller must retain the last received pulse width's value.

In this application, each pump gets speed-updated every 30 sec by a 50- to 1000-msec pulse. The pump drive must provide optimum speed-setting resolution for the low-speed ranges to provide increasingly slower flow rates as the system approaches crucial mixing conditions. And the controller must possess a high degree of noise immunity to prevent spurious noise-induced responses from degrading process quality.

Fig 2 illustrates a μP-based-controller scheme. In this arrangement, the computer delivers an input pulse that gates a clock. The clock in turn serially loads a bank of parallel counters that determine the input pulse width. The counters address a processor section that converts input data to a frequency output, using an exponential transfer function—a nonlinear response that achieves the required high resolution (precise liquid delivery) at slow pump speeds. Finally, the frequency output activates a stepping-motor driver that runs the pump.

On the surface, this digital controller's operation appears relatively simple. However, the application masks some tricky design problems. For example, the lengthy period between speed updates, coupled with the need to avoid erroneous pump responses, mandates careful power-supply design, including provision of such functions as RFI filtering, memory battery backup and self-checking software.

In addition, the need for a high-resolution, smoothly varying frequency-output function demands careful design attention to how the processor synthesizes its

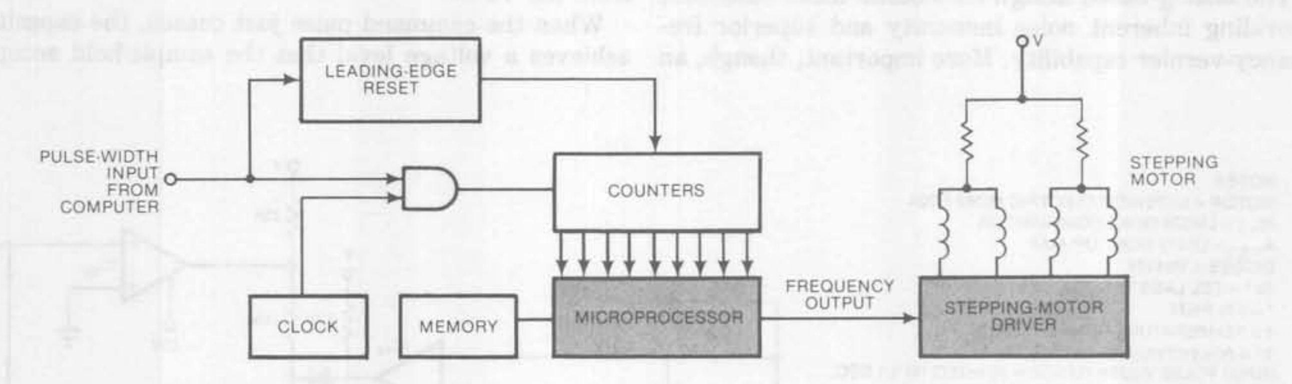


Fig 2—Upon receiving gated pulses, *a μP converts timed computer data into a frequency output, using an exponential transfer function. This nonlinear response results in the necessary high-resolution-at-low-speed characteristics for accurately controlling pump operation with a stepper-motor driver. The problems that can arise with this digital approach to controlling* ***Fig 1****'s mixing system include noise sensitivity, memory-retention difficulties and an undesirable quantized frequency-shift characteristic.*

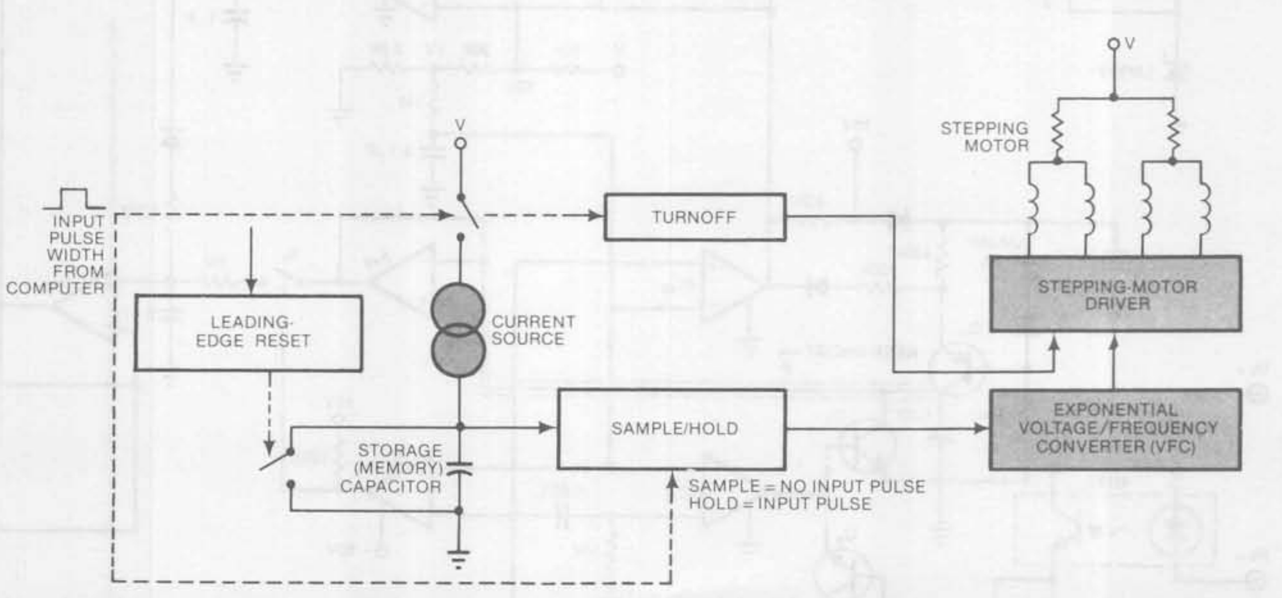


Fig 3—In this analog-pump-controller approach, *a computer's command pulses direct a current source, which in turn charges a storage capacitor that provides noise-immune analog-data retention. When the command pulse ceases, the sample/hold amp receives the capacitor's stored voltage and delivers it to the exponential voltage-to-frequency converter (VFC). The VFC activates the stepper-motor driver in a continuous, smooth manner; the turn-off stage deactivates the motor driver.*

Analog functions prove adequate in simple process-control tasks

output. Although these problems are amenable to solution, they complicate the controller's design and entail lengthy development time and high cost.

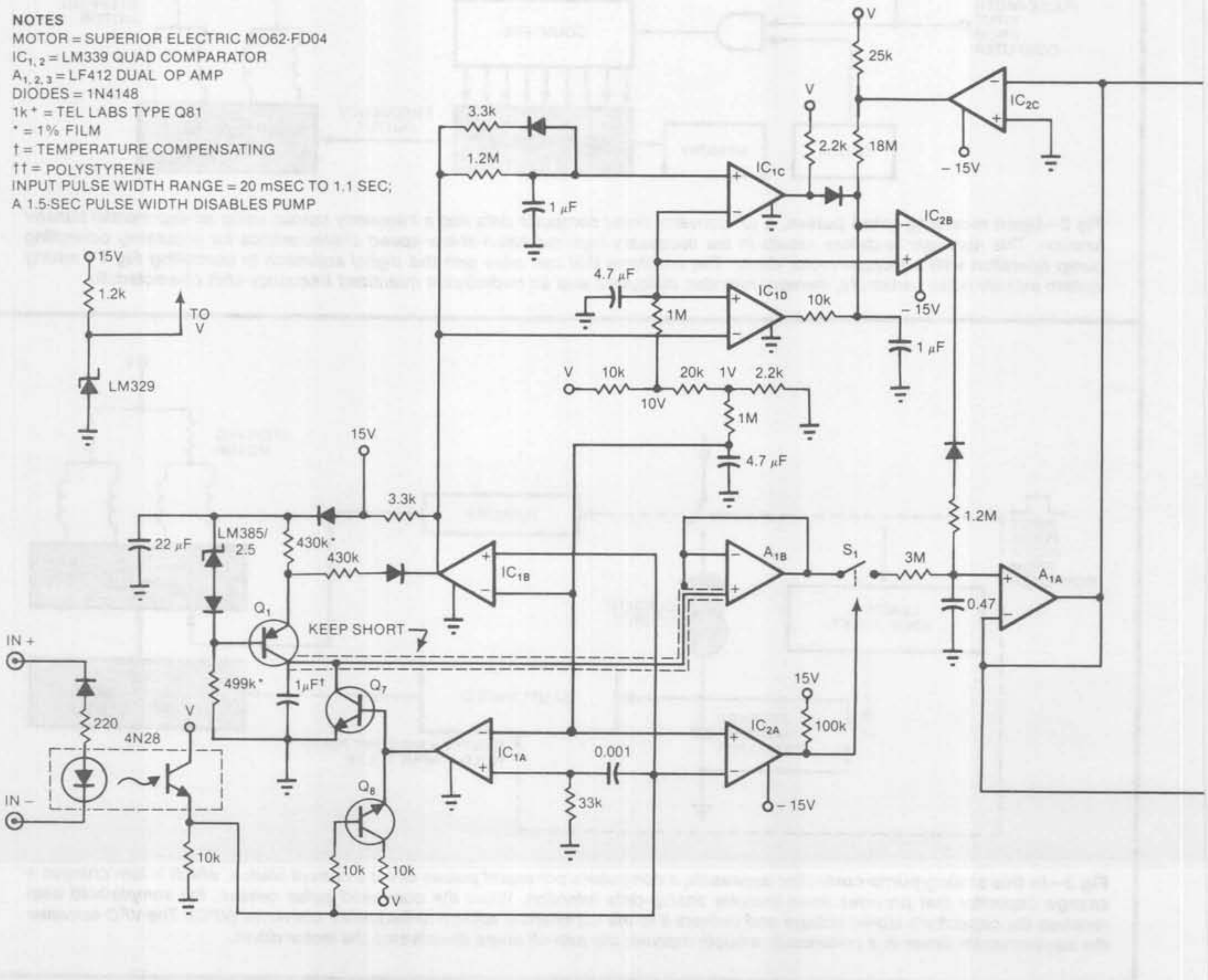
Take the analog route

Considering the task's conceptual simplicity, however, reveals a clear edge for an analog-control approach to satisfying this application's critical requirements. A turnkey system, it needs little intelligence or flexibility and can employ a straightforward data-retention structure. And although the digital μP-based approach can also meet these requirements, it involves substantial hardware and software overhead to overcome noise-immunity and frequency-shift-resolution problems.

The analog-based design surmounts these obstacles, providing inherent noise immunity and superior frequency-vernier capability. More important, though, an analog approach eliminates the intensive software effort required by μP-based methods. As a matter of record, the analog pump-controller design was conceived, breadboarded and released for production in just 4 wks—and at a cost competitive with an alternative μP-based method.

Fig 3 depicts the analog system. In this scheme, a capacitor furnishes memory storage. An exponentially responding voltage-to-frequency converter (VFC) fulfills the function of **Fig 2**'s processor. In operation, the computer's command pulse gates a current source that linearly charges the storage capacitor. While the capacitor is charging, the sample/hold stage enters Hold mode, blocking the capacitor's ramping action from the VFC.

When the command pulse just ceases, the capacitor achieves a voltage level that the sample/hold accepts



and feeds to the VFC. By issuing an extremely wide pulse, the computer actuates the turn-off stage, which deactivates the stepper-motor drive.

Optoisolation eliminates noise

Fig 4 shows the analog pump controller's schematic diagram. To initiate circuit action, the computer sends an input pulse to the 4N28 optoisolator, which eliminates noise-pickup-induced ground-loop and data-line problems. Appearing at its emitter, the optoisolator's output (**Fig 5,** waveform A) goes to IC_{1A} and IC_{1B}. IC_{1A}'s differentiator setup—a 0.001-μF/33-kΩ combination—generates a short pulse (**Fig 5,** waveform B) that biases Q_7. This transistor in turn resets its associated 1-μF capacitor (**Fig 5,** waveform C).

Note that Q_8's emitter supplies the current to base-bias Q_7 ON because IC_{1A} is an open-collector device. In turn, Q_8 receives its base bias from the optoisolator, which provides a drive output only when a command pulse appears at the controller circuit's input. Consequently, in the highly unlikely event that a severe noise disturbance causes IC_{1A}'s output to rise, Q_7 still doesn't receive a drive pulse, and its 1-μF capacitor does not get reset.

The 1-MΩ/4.7-μF filter, which feeds IC_{1A}'s minus input, provides additional noise immunity by ensuring a stable trip point during noise disturbances. The optoisolator's output also goes to IC_{1B}, which gates the Q_1 current source. When Q_7 turns off, its 1-μF capacitor immediately starts to ramp up (**Fig 5, waveform C**). (Circuit-operation speed in **Fig 5** has been increased to provide optimum waveform photographs.) Then, the A_{1B} follower unloads the capacitor.

Diode/capacitor decoupling of Q_1 assures high noise

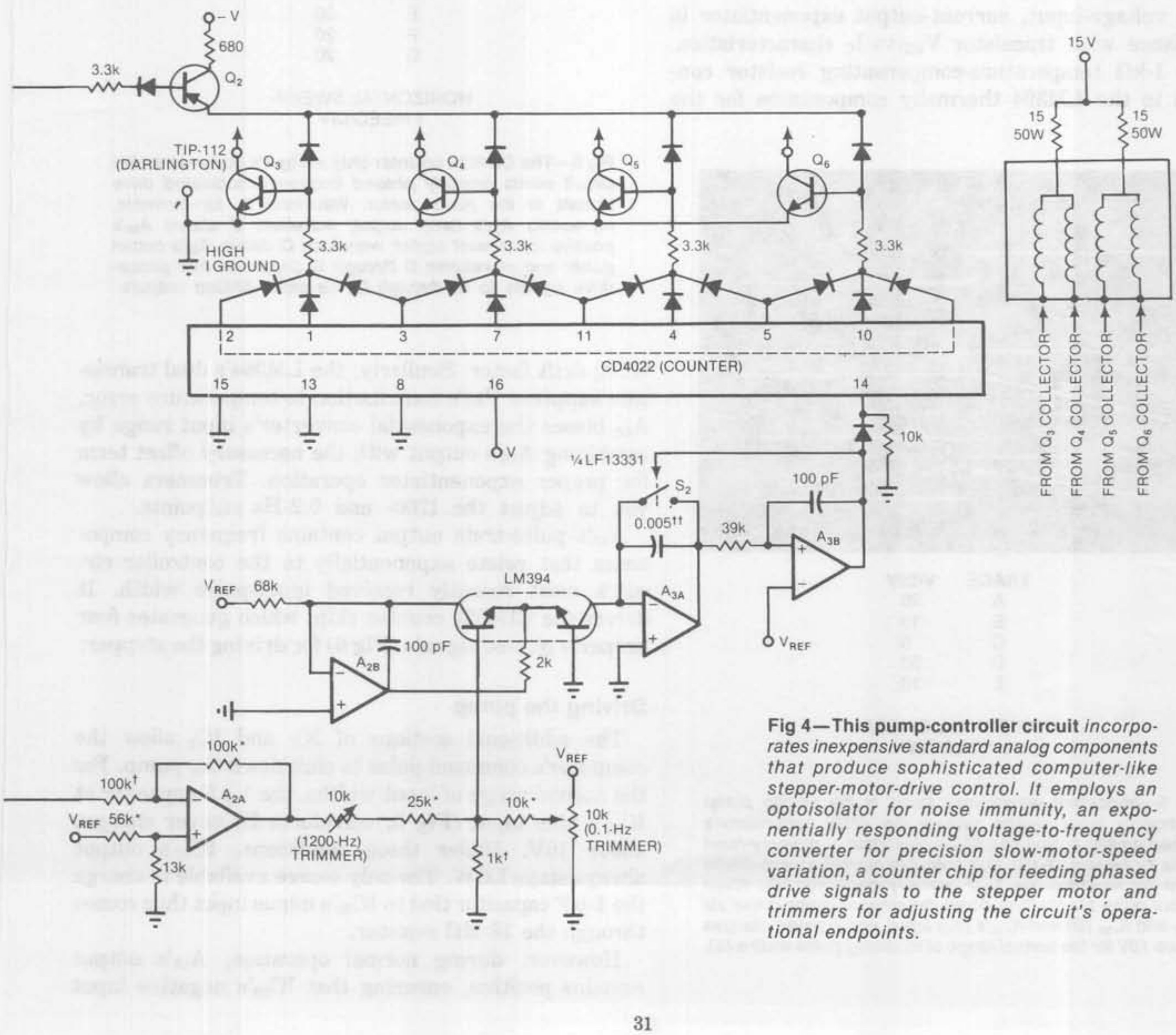


Fig 4—This pump-controller circuit *incorporates inexpensive standard analog components that produce sophisticated computer-like stepper-motor-drive control. It employs an optoisolator for noise immunity, an exponentially responding voltage-to-frequency converter for precision slow-motor-speed variation, a counter chip for feeding phased drive signals to the stepper motor and trimmers for adjusting the circuit's operational endpoints.*

A voltage-to-frequency converter controls stepper-motor drive

rejection, even for supply dropouts, during the capacitor's ramp time. During ramping, IC_{2A}'s output stays LOW and shuts off S_1. This switch maintains A_{1A}'s output at a dc level. When the controller's input pulse ceases, IC_{1B}'s output goes LOW and disables Q_1. The integrating 1-μF capacitor therefore stops charging. Concurrently, IC_{2A}'s output goes HIGH and closes S_1. As a result, A_{1A}'s output changes to the capacitor's newly acquired level. Located in A_{1A}'s input section, the 3-MΩ/0.47-μF filter provides a time constant that limits the stepper motor's acceleration rate, thereby preventing stalling.

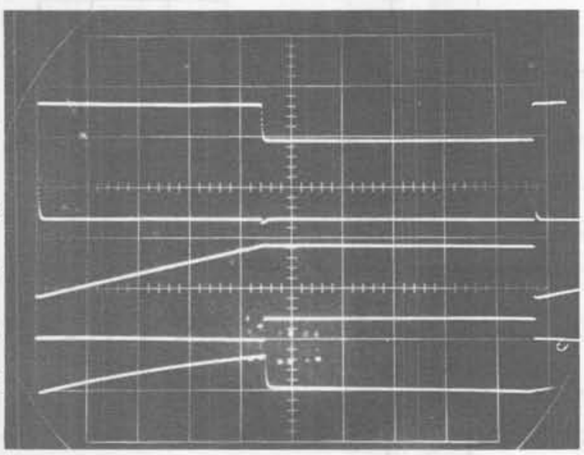
Try an exponentiator

Op amp A_{1A}'s output feeds the A_2-A_3 configuration, which forms an exponentially responding VFC that controls the input current to the A_{3A}-A_{3B} integrator-comparator-type oscillator stage. To accomplish this function, A_{2B} and the LM394's dual transistors constitute a voltage-input, current-output exponentiator in accordance with transistor V_{BE}-vs-I_C characteristics.

The 1-kΩ temperature-compensating resistor connected to the LM394 thermally compensates for the KT/Q drift factor. Similarly, the LM394's dual transistors suppress V_{BE}'s contribution to temperature error. A_{2A} biases the exponential converter's input range by combining A_{1A}'s output with the necessary offset term for proper exponentiator operation. Trimmers allow you to adjust the 1200- and 0.2-Hz endpoints.

A_{3B}'s pulse-train output contains frequency components that relate exponentially to the controller circuit's most recently received input-pulse width. It drives the CD4022 counter chip, which generates four properly phased signals (**Fig 6**) for driving the stepper.

A
B
C
D
E

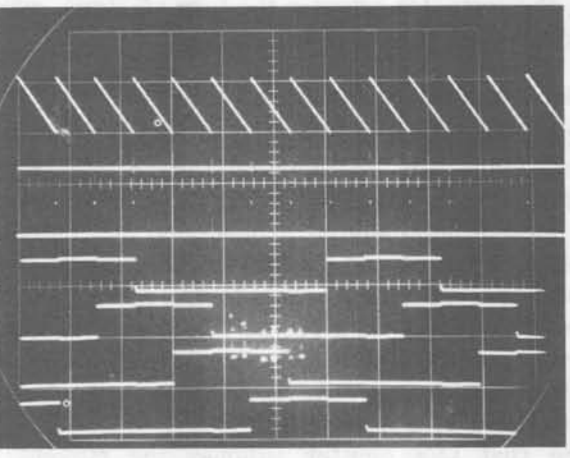


TRACE	V/DIV
A	20
B	10
C	5
D	50
E	10

HORIZONTAL SWEEP-
1 mSEC/DIV

Fig 5—Important waveforms *found in the analog pump controller's input section include the 4N28 optoisolator's pulsed emitter output* ***(A)****, IC_{1A}'s plus input or memory-reset spike for biasing Q_7* ***(B)****, Q_7's output or current-source-driven ramp for resetting the 1-μF memory capacitor* ***(C)****, IC_{1D}'s output pulse for shutting down the stepper-motor driver via IC_{2B} and IC_{2C}* ***(D)*** *and IC_{1C}'s plus input, which never charges above 10V for the normal range of incoming pulse widths* ***(E)****.*

A
B
C
D
E
F
G



TRACE	V/DIV
A	5
B	50
C	20
D	20
E	20
F	20
G	20

HORIZONTAL SWEEP-
1 mSEC/DIV

Fig 6—The CD4022 counter chip *in* **Fig 4***'s pump-controller circuit sends properly phased frequency-modulated drive signals to the pump motor. Waveform* ***A****, for example, represents A_{3A}'s ramp output; waveform* ***B*** *shows A_{3B}'s positive input reset signal; waveform* ***C*** *details A_{3B}'s output pulse; and waveforms* ***D*** *through* ***G*** *depict the four phase-drive signals to Q_3 through Q_6 via diode-ANDed outputs.*

Driving the pump

The additional sections of IC_1 and IC_2 allow the computer's command pulse to shut down the pump. For the normal range of input widths, the 1-μF capacitor at IC_{1C}'s plus input (**Fig 5,** waveform E) never charges above 10V. Under these conditions, IC_{1C}'s output always stays LOW. The only source available to charge the 1-μF capacitor tied to IC_{2B}'s minus input thus comes through the 18-MΩ resistor.

However, during normal operation, A_{1A}'s output remains positive, ensuring that IC_{2B}'s negative input stays that way. This condition forces IC_{2B}'s open-collector output to float. If the controller circuit receives an input pulse substantially wider than the normal maximum, therefore, IC_{1C}'s input charges above 10V. This action quickly dumps a large charge into IC_{2B}'s 1-μF capacitor, forcing its voltage level to rise to the negative rail. This value pulls A_{1A}'s input negative, turns on Q_2 and cuts off all drive signals to the output transistors (Q_3 to Q_6).

A_{1A}'s negative output also feeds back to IC_{2C}, driving that device's output positive. This output supplies a continuous topping-off current to IC_{2B}'s input capacitor. The connection completes a positive feedback latch, which prevents the pump from operating until the counter receives a pulse width within the controller circuit's normal range. IC_{1D} functions to clear out the IC_{2B} capacitor's charging action (**Fig 5,** waveform D) as each new command pulse arrives.

The time constant associated with A_{1A}'s input section lets the controller circuit examine each received pulse and never disables this clamping performance unless the pulse width resides within established limits. Although the latch's positive feedback doesn't require the computer to send successive shutdown instructions to the pump, the controller circuit ensures that the pump's motor can't be energized, even briefly, if successive turn-off-length pulses appear. **EDN**

Author's biography

Jim Williams, now a consultant, was applications manager with National Semiconductor's Linear Applications Group (Santa Clara, CA) when he wrote this article. Before working at National, he was employed by Arthur D Little Inc and the Instrumentation Development Lab at the Massachusetts Institute of Technology. A former student of psychology at Wayne State University, Jim enjoys tennis, art and collecting antique scientific instruments in his spare time.