ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01

FEEDBACK -- CLOSING THE LOOP DIGITALLY

J. Zagel, B. Chase
Fermi National Accelerator Laboratory*
P.O. Box 500
Batavia, IL 60510

Abstract

Many feedback and feedforward systems are now using microprocessors within the loop. We describe the wide range of possibilities and problems that arise. We also propose some ideas for analysis and testing, including examples of motion control in the Flying Wire systems in Main Ring and Tevatron and Low Level RF control now being built for the Fermilab Linac upgrade.

I. INTRODUCTION

The standard techniques used to design and analyze analog feedback systems can also be applied to digital systems. It is desirable to consider frequency response, maximum tolerable error, and stability questions for systems controlled by processors. In modern digital systems a considerable amount of software not only replaces analog circuit functions but also allows additional features to be built into the system.

II DEFINITIONS

A. Control System

A control system is generally described as a system that provides an control output variable C in response to an input reference R. This can be accomplished open loop or closed loop. Open loop control means that for a given input, the plant G provides a fixed response regardless of any external loading on the controlled device or process. Closed loop control uses feedback signal H to compare the output to the reference input and generate an error E which is then minimized by the loop.

A predictor of the desired output can be applied to the drive circuit thus producing a feed forward signal. Predictions are normally obtained by computations on a mathematical model combined with measurements of the actual process.

The plant G can be viewed as the combination of fixed drive characteristics plus an equalization, or compensation, filter applied to correct any undesirable behavior. The feedback can be a simple transfer function, such as position to voltage, or a complex filter to aid in measuring the controlled process.

An open loop system is described as the convolution of r(t) with g(t) in the time domain. It is more convenient to analyze these systems in the frequency domain using Laplace transforms. This transforms convolution integrals to multiplication for continuous, linear systems, or $C(s) = R(s) * G(s)$. With feedback, the transfer function is described for the closed loop configuration to evaluate the response and stability of the system. For a closed loop: $\frac{C(s)}{R(s)} = \frac{G(s)}{(1+G(s)H(s))}$

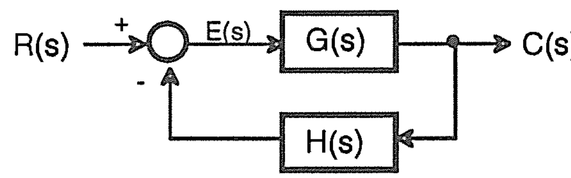


Figure 1. Basic Control System Terminology

B. PID Loops

The most common controller is the proportional - integral - derivative loop (PID). To understand the PID loop we will look at the pieces of a motion control system. When a position change is required the reference input to the system is modified. The system will generate a drive signal proportional to the position error developed between the now changed reference input and the previously held position.

For a step change in the reference input, the drive electronics may allow the motor to far overshoot the desired change. In this case it is useful to consider the first derivative term of the velocity, or acceleration, to maintain stability. This is also referred to as lead compensation.

To correct for long term or steady state errors in the desired output a third, integral, term is included in control loop. This term removes accumulated error over time. This is referred to as lag compensation.

The mathematical formulation[1] for the PID filter in time is: $u_{PID}(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de}{dt}$

which transforms to:

$$U_{PID}(s) = K_p + K_i \frac{1}{s} + K_d s$$

$$= \frac{K_d s^2 + K_p s + K_i}{s}.$$

This filter function has one pole at the origin and two zeros that are dependent on the three gain terms.

C. Hardware - Software Equivalents

To implement the PID equation above active elements are used. The hardware is shown in figure 2. Each term is shown as an independent active element however in practice this circuit can be simplified.

While Laplace transforms take us into a convenient domain to analyze analog circuit behavior, the z-transform better serves the transition into sampled time domain.

*Operated by Universities Research Association Inc., under contract with the U.S. Department of Energy.

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01

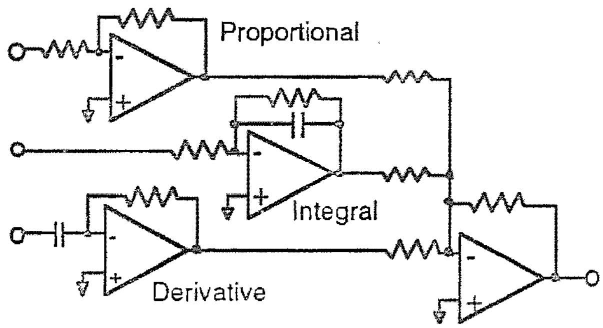


Figure 2. Analog Hardware Elements of PID Filter.

The transform: $s = \frac{2(z-1)}{T(z+1)}$ and $z = \frac{1+\frac{sT}{2}}{1-\frac{sT}{2}}$ known as the bilinear transformation, applied to the Laplace yields the z domain.

The function expressed in the z domain assumes a sampling in time where the sample interval is constant. The filter function is then transformed into a series of outputs correlated to the n^{th}, n - 1, n - 2, etc. samples in time. This yields a difference equation for the filter output :

$$u(t) = u(n-2) + K_1e(n) + K_2e(n-1) + K_3e(n-2)$$

where $K_1 = K_p + 2K_d/T + K_iT/2$,

$K_2 = K_iT - 4K_p/T$,

and $K_3 = 2K_d/T - K_p + K_iT/2$.

These constants can then be programmed into a digital signal processing system. [2]

D. Stability

The typical stability criterion applied to analog systems are Bode plots in frequency, and s-plane ($s = \sigma + j\omega$) plots for Laplace transfer functions. Bode plots graph the gain and phase of the system vs frequency to determine both gain and phase margin. A stable system has a gain less than 1 when the phase reaches 180^o.

The s-plane plots the imaginary and real parts of the Laplace transfer function. The requirement for stability is that all roots of the characteristic equation have negative real parts. Using the z-transform for discrete time sampling systems the z-plane stable area maps into the unit circle.

III. Motion Control

A. Introduction

Typically motion control systems fall into two categories. A velocity control system tries to maintain a continuous speed profile for a system while a position control system will try to first establish and then maintain a position corresponding to the desired input. The Fermilab flying wire system described previously[3] uses a combination of these to maintain position and control the velocity profile of the system when a position change is required.

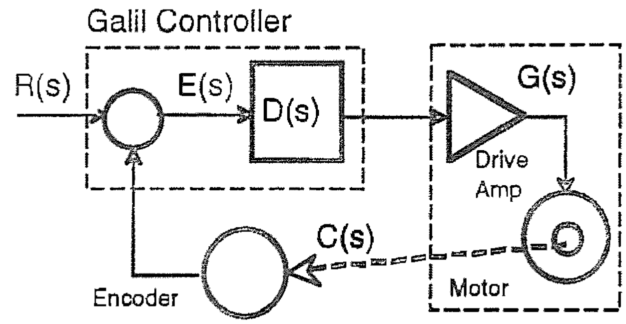


Figure 3. Flying Wire Control Loop

B. Velocity Profile

The position change of the wire is accomplished by accelerating at a constant rate, holding a constant velocity, and then decelerating at the same constant rate used for acceleration. The Galil Controller contains a digital filter[4] with $D(z) = GN\frac{z - \frac{zr}{256}}{z - \frac{pl}{256}}$ which transforms to $G(s) = K\frac{s+a}{s+b}$ which, for a<b, is a lead filter. K = GN and the zero and pole term relate to a and b as $GN\frac{256 - zr}{256 - pl} = K\frac{a}{b}$.

The wire speed in the Fermilab Tevatron is $5\frac{m}{s}$. The wire is mounted on a fork that is .0965 m radius yielding an operating frequency of 325 radians/sec. The optical encoder provides 16384 counts/rev. Hardware prescales the counts of the encoder and the processor periodically samples the output to determine the actual position error on a time scale that matches the required system performance. Typically a sampled system should run at 10 to 20 times the minimum sample rate for the measurement process. It is clear that the processor can not directly sample the encoder inputs in real time. However the motion of the wire can be corrected on the millisecond timescale required of the system.

C. Tuning the Loop

We want to select the parameters to produce a fast but stable response. For the flying wire system the most stable set of operating parameters have historically been determined empirically. We have since taken measurements with an HP3563A Control System Analyzer to determine the true best parameters over a wide range of operating conditions.

From these measurements the natural oscillation frequency can be obtained. The zero is then selected to be half to two thirds of the natural frequency such that: $ZR = 256e^{-0.4\omega_c T}$ where T is the sample interval. For an ω_c = 200 rad/sec and T = 500 μsec yields ZR = 246. Our zeros are in the range of

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01

245 to 248. The digital number input is 0 to 255. The gain varies from system to system but in the range of a few counts (min 4 to max 7) out of 256.

The mechanical components of the system are subject to change due to temperature, humidity, and wear. If the closed loop parameters are set appropriately then the system will operate reliably over time. For our system the pole is set at its default value.

D. Control System Analyzers

There is a new class of instrumentation available to aid in the process of designing and implementing closed loop systems referred to as Dynamic System Analyzers. These analyzers work in both time and frequency domain. A special case is the Control System Analyzer (CSA) which also allows inputs or outputs to be either digital or analog. Once the measurement has been made, a wide range of math functions is available for analysis.

The CSA has been used to make step response measurements in the time domain. It will be used to make measurements of the closed loop performance of the system. This work has not yet been accomplished.

IV. LINAC LLRF

A. Introduction

The last three 201 MHz accelerator sections of the Fermilab linear accelerator are being replaced by seven new sections operating at 805 MHz, each driven by a 12 MW Klystron. Due to the higher frequency and higher gradient this upgrade will increase the beam energy of the linac from 200 to 400 MeV. In order to minimize momentum spread of the beam, tight regulation of the RF gradient in the cavities is needed. Feedback alone does not have the needed bandwidth to meet the regulation specification, therefore a processor based learning feedforward system was added.

B. The System

The accelerating field gradient in the cavity is regulated by a feedback system controlling the magnitude and phase of the RF field inside the cavity. The design specification is to regulate the gradient field to 1% magnitude and 1 degree phase during the time that beam is present. This requirement for regulation is made difficult by three aspects of the RF system. First, there are long group delays, (400ns), through the Klystron and wave-guides connected to the cavity. The Laplace transform of this delay(Td) is e^{-sTd}. This is a frequency dependent phase shift that limits the closed loop bandwidth of the feedback loop to less than 400KHz. Second, the beam loading is a rectangular pulse that has a very fast rise time. This demands a fast response from the control loops. The third problem is that while the cavity responds like a simple LC resonator at its central resonant frequency, it can also be excited in other modes at frequencies that are within ±500KHz. Any positive loop gain at these resonant frequencies will make the system unstable.

A fast feedforward loop that is able to learn from the past history of accelerating cycles is needed. We chose a digital system to generate the feedforward waveform because of its versatility and accuracy. Digital signal processing techniques enable the use of a learning algorithm. Because true system testing will not be done until all installation is done and there is beam in the machine, the hardware needs to be made as general as possible. Any last minute modifications can then be done in software. A block diagram of the RF system is shown in figure 4. Note that all the electronics for the low level system resides in the VXI crate.

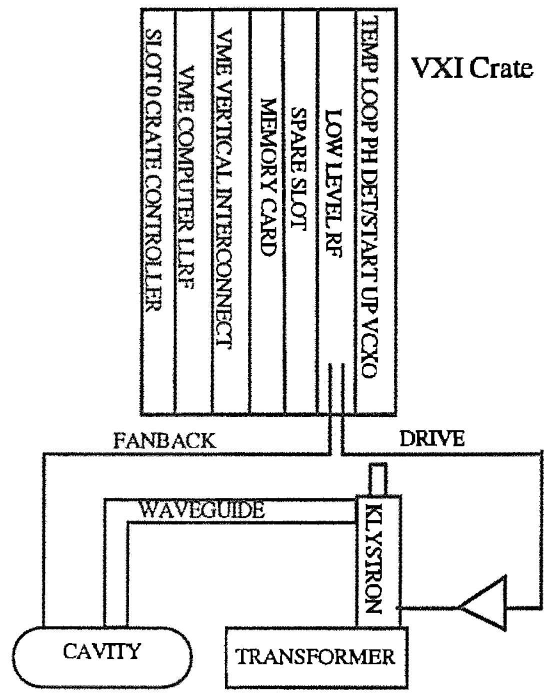


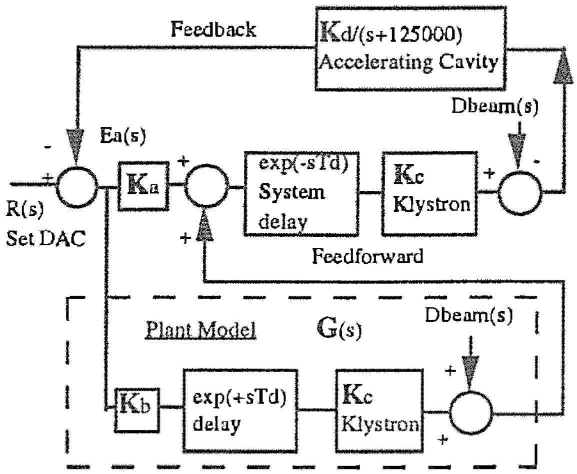
Figure 4. LINAC RF System

C. The Model

The system was modeled to a great degree of accuracy using a time domain analysis program, EXTEND[5]. Using EXTEND, the nonlinear components such as the Klystron, mixers and phase shifter are modeled so that the system can be understood over its full dynamic range of operation However a much simpler linear model works well over a narrow dynamic range. This linear model is shown in Figure 5.

All signals shown in the diagram are modulation signals of the 805 MHz signals magnitude or phase. The 805 MHz carrier is mixed in and out with the control signals, but has no direct relation to the feedback loops.

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01



Ka is feedback loop gain.
Kb is a "learned" gain by an IIR filter.
G(s) is a heuristically determined function.
Dbeam(s) is the disturbance due to the beam
Feedforward is active only during beam.

Figure 5 Linear Model of RF System.

The feedforward system was dictated by the conflict between the need for fast system response (200 ns) and large group delay (400 ns) due to the Klystron and waveguide. Feedforward is made possible because the system, running at 15Hz, is very repetitive from one acceleration cycle to the next. This enables the algorithm to learn the gain K_b by the use of an IIR Filter that is incremented once each cycle. Therefore, the filtering function is in the domain of cycle to cycle time and not the linear time of the 125us of one accelerating cycle.

G(s) is the plant model for the transfer function of the beam loading effect and Klystron combined. G(s) can be any function or array that is found to be the best fit and is limited to only a 2.5 MHz bandwidth by an anti-aliasing filter.

D. The Hardware

In order to implement the feedforward system, a fast wave-form capture and playback generator is needed. The amount of signal processing and I.O. needed demanded a high-speed dedicated processor and wide-band computer bus. Good RF shielding at 805Mhz and a quiet environment for signal conditioning is necessary. The system also needs to be remotely operated and must have good remote diagnostics. Due to the large number of components a large card form factor was needed. VXIbus was chosen as the platform, as it was conceived for this general type of application. Figure 6 is a block diagram of the Low Level Module.

The control system is designed to be as flexible as possible, since full testing of the system will not be done until there is beam in the machine. This philosophy is embodied by giving the processor full access to the wideband error signals of the feedback loops and 16 circuit diagnostic points that may be sampled at any time during the 125 us of the RF pulse.

The error signals for both the magnitude and phase loops are sampled at 10 Mega-samples/s by an 8 bit ADC and stored in FIFO memory. After the error waveforms have been digitized, the module generates an interrupt that starts the DSP routines in the processor. The computed feedforward waveform is then loaded into another FIFO memory and played back during the next accelerating cycle through a 12 bit DAC. This signal is summed with the feedback signal before driving the phase shifter or the magnitude modulation mixer. The 8-bit data from the ADC is averaged by the CPU which gives a higher resolution to the playback DAC.

E. The Operating System

The MTOS-UX kernel from Industrial Programming, Inc. is a true multi-processor multi-tasking real-time operating system. This kernel is in wide use at Fermilab and has been ported to many different platforms.

The system is designed so that on startup, the software detects the number of low level RF modules and processors that are installed in the VXI mainframe. It then configures itself to distribute the processing capability as needed by each LLRF system. This allows the system to be reconfigured by users without the support of a system software expert.

F. The Software

The main job of MTOS (see Figure 6), is to complete two main tasks when it receives an interrupt from the LLRF module. The interrupt starts the producer task "I_15HZ", which then reads the 16 MADC channels and the two ADC FIFO memories. Next, it reads the control buffer memory and then writes to the LLRF Module registers. Finally, it places pointers to the data arrays in the MTOS message buffer, and then unblocks the consumer task M_15HZ.

"M_15HZ" processes the error waveform, creates the feedforward waveform, and writes it to the DAC memory in the LLRF module . It then resumes waiting on the message buffer for the next task. These tasks must be completed before the next interrupt is received.

The task "TEST" keeps track of the percent of idle time for the processor and displays it on an SSM module. This allows the developers to monitor processor bandwidth as changes are made. This has proved to be a very useful function.

"DIAGNOSTICS" is off-line code that will test many of the signal paths in the module. This includes memory, ADCs, DACs, MADCs and some of the RF paths. The human interface for this will be LabView running on a Macintosh 2ci. Communication from the Macintosh to the LINAC control system is done over TokenRing.

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01

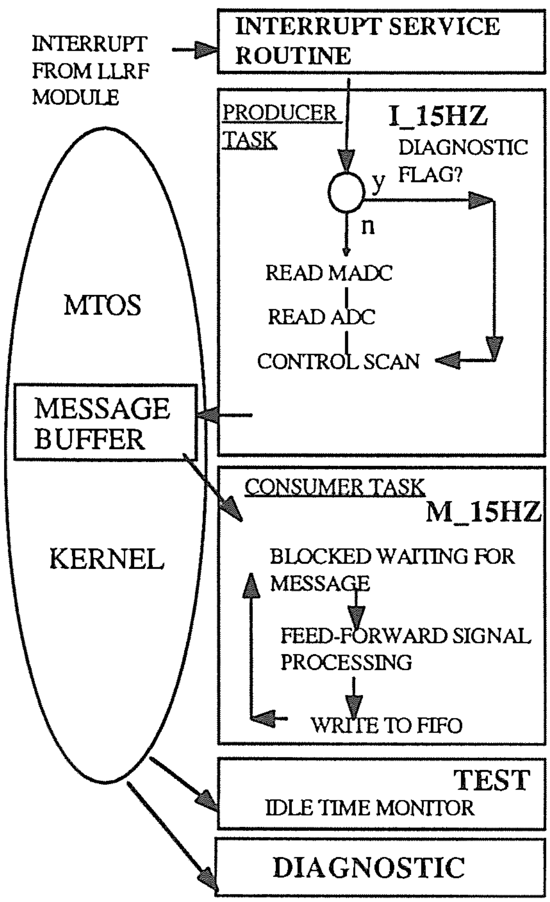


Figure 6 LLRF software overview

G. Feedforward

Originally the concept was to have the feedforward loop learn the entire error waveform. Each sample would have an IIR filter that would learn the best value for that point in time to minimize the error signal. In the model this worked well as all the system delay could effectively be removed from the loop. This would allow the effective system closed loop bandwidth to be increased to about 2.5MHz, the frequency of the anti-aliasing filters. This algorithm continuously adapts to any changes of the system. The main problem with this approach is the other resonant modes of the cavity that are within ±2.5 MHz of the main resonant mode. These modes are ideally nulled out of the fan-back signal by summing all 8 cells of the cavity together. However, mismatches in the signal combiner cause gain slope changes in the response that cause the closed loop system to oscillate. Another approach is needed in order to achieve the needed closed loop response.

The beam profile is very close to a rectangular pulse as it is run through a beam chopper after the ion source. See figure 7. To model the beam loading effect, the only information needed is the start time, stop time, and the amplitude. If the error waveform does not change except in amplitude, a simple software routine can create the proper shape of the feedforward signal. The amplitude can be "learned" by looking at past error waveforms. Thus fast changes can be made to the drive signal, (100ns), without affecting the closed loop bandwidth.

The proper amplitude of the feedforward pulse is computed in the following manner. A section of the error waveform, before the beam, is averaged in order to establish a baseline value. In the same way the average error is found while beam is present. The beam present value is subtracted from the pre-beam value and multiplied by a gain constant. This value is then filtered over many accelerating cycles by an IIR filter. The filter algorithm is simply 5% new value plus 95% old value. This filtered value is then added to a baseline value of the feedback waveform between the specified start and stop times of the beam. When the loop comes into its steady state value the error waveform will be flat across the top. This is because the feedforward loop is doing all of the beam loading correction while the feedback loop is taking care of slower errors.

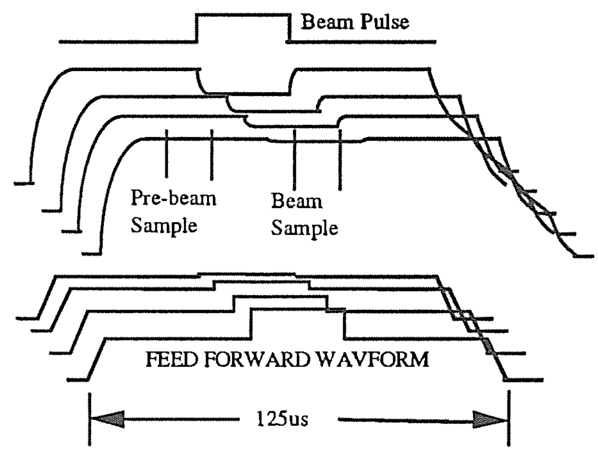


Figure 7 Feedforward learning over several pulses.

In this system, the processor is not in the feedback loop, therefore processing time does not create phase shifts and possible loop instabilities. Feedforward, in the time frame of a single pulse is an open loop process. Therefore loop stability is not an issue as long as the processor gets the job done in the 66ms before the next cycle.

H. Results

The Low Level RF system has been operated on a complete system test bed in the Fermilab A0 lab. It responds very closely to the model with few surprises. One area that is not accurate in the model is the complex interaction of the Klystron, waveguide and cavity. True beam loading effects have not been seen as of yet, however, system settling time to 1% was measured to be less than 300 ns for a 20% step change introduced by a test system. The versatility of the system was demonstrated when system parameters for tuning were changed on the fly from the parameter page. Problems such as the saturation of ADC's or DAC's were easy to find by using the

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S12FC01

quick plot routines. The LLRF system was operational at A0 in the first week after installation.

V. CONCLUSION

The use of feedforward for the LLRF system allows flexibility in the design and implementation. Once the system becomes operational some minor changes in the feedforward parameters or the algorithm can easily be made.

For the Flying Wire system we hope to generate an optimal set of parameters with a single command to the system. The processor should be able to manipulate the motion control system to determine the actual friction, inertia, and loading of the entire system as installed . When a failure occurs that requires component replacement, the system will be programmed to self tune the loop for the best response.

VI. REFERENCES

[1] DiStefano, J. et. al. *Feedback and Control Systems*, McGraw-Hill, 1990, or other Control Systems Texts.
[2] Ahmed, I., "Designing Control Systems", *Digital Signal Control Applications with the TMS320*, Texas Instruments, 1991, pp 48-52.
[3] Gannon, J.,et al., "Flying Wires at Fermilab," Proceedings of the 1989 IEEE Particle Accelerator Conference, pp. 68-70, March 20-23, 1989.
[4] Tal, J., *Motion Control Applications*, Galil Motion Control Systems, 1989.
[5] EXTEND copyright by Imagine That, Inc. San Jose, CA, USA. Details of analysis in Pasquinnelli, R., "Modeling of the Linac Upgrade RF System", Fermilab internal memo, February 6, 1991.