# Microcomputer Implementation of Digital Control Strategies for Structural Response Reduction

P. Quast,[a] M. K. Sain,[a] B. F. Spencer Jr. ASCE,[b] & S. J. Dyke[b]

[a] Department of Electrical Engineering,
University of Notre Dame, Notre Dame, Indiana 46556.

[b] Department of Civil Engineering and Geological Sciences,
University of Notre Dame, Notre Dame, Indiana 46556.

**Abstract:** *In the past, implementation of active structural control strategies employing PC-based hardware has often been problematic, and there seems to be a need to improve the understanding of the issues associated with implementation of digital control, as well as to reduce the excessive computational times required for I/O processes and calculation of control forces. Recently developed hardware based on dedicated digital signal processing (DSP) chips has offered new possibilities for control algorithm implementation. This paper discusses digital control system concepts and specific practical aspects of digital control implementation including the use of supervision. Details regarding active structural control using the Texas Instruments TMS320C30 DSP chip and experimental verification of the hardware performance are given.*

## 1 Introduction

Structural control has received wide attention in recent years.[1],[8],[13] However, the practical implementation of digital control for structural applications is fraught with many issues that must be thoroughly understood and resolved for successful controller implementation. In the past, the hardware to implement a digital controller typically included a personal computer (PC) to perform control algorithm calculations, and a data acquisition board. The primary tasks of the data acquisition board were to perform the analog to digital (A/D) conversion of the measured quantities for use by the PC and to perform the digital to analog (D/A) conversion of the command signal calculated by the control program. This configuration has many drawbacks. The time required to perform all of the A/D and D/A operations and to pass these quantities over the I/O space of the PC, as well as to perform the control algorithm computations on the PC, may require undesirably large sampling periods and also induce unacceptably long time delays. Also, this configuration makes it very difficult to create a supervisory control scheme which allows the operator to monitor and interact with the controller while the PC is engaged in performing control computations.

At the University of Notre Dame's Earthquake Engineering/Structural Dynamics and Control Laboratory (EE/SDCL), these problems have been greatly reduced or eliminated

with the aid of a dedicated Digital Signal Processing (DSP) System which is able to perform the A/D and D/A conversions and control calculations on a single board. This paper highlights general digital control system concepts as applied to active structural control. Practical issues regarding digital controller implementation such as the true sources of time delay, the considerations involved in evaluating the sampling rate required, and I/O device performance are discussed. The use of supervisory control techniques is also discussed. The particular DSP hardware used in the EE/SDCL and experimental verification of the hardware performance is presented for a seismically excited scale model of a three story building.

# 2 Fundamentals of Digital Control

There are many advantages to the use of digital control systems over conventional continuous-time/analog controllers. One of the primary advantages is that the control law that is used can be easily and precisely modified by simply changing the computer program used to carry out the digital control scheme. Further, digital control schemes allow for the relatively simple attainment of a high degree of precision in the implementation of a controller, whereas analog controllers may require rather elaborate and expensive circuitry in order to attain the same level of precision for a given control law. Also, digital control allows for a much simpler implementation of nonlinear control schemes and affords the opportunity to use decision making or logic capabilities within the controller so that supervisory control techniques can be easily incorporated. Such techniques would allow for such things as monitoring and operation of the control system by a computer implemented 'supervisor' and safe system shutdown in the event of operation of the system beyond acceptable operating bounds. This section discusses a number of important concepts and principles that need to be considered in the design and implementation of digital controllers.

## 2.1 Discrete-Time Systems

Digital control systems involve the use of discrete-time signals. Such signals have amplitudes defined only at specific, discrete instants in time. In most applications where discrete-time systems are used in digital control, discrete-time signals are defined at regularly spaced instants in time, the interval between these instants being referred to as the period $T$ of the discrete-time signal. The instants of occurrence of a discrete-time signal are usually described by some initial time, taken in the sequel to be 0, plus some multiple $k$ of the period $T$. Further, a digital signal is commonly defined as a discrete-time signal for which the amplitude can only take on a finite number of specific, discrete values.

In the same way that a linear continuous-time system can frequently be described by a set of first order differential equations, *i.e.* a state space realization, linear discrete-time systems can also often be described using a state space set of first order difference equations. Difference equations describe the values of 'states' of a discrete-time system at a given time instant based on the value of the states and inputs at previous time instants. An example of a state space realization of a linear, time-invariant, finite-dimensional, discrete-time system typically used in digital controllers is:

$$\mathbf{x}(kT+T) \;=\; \mathbf{A}\mathbf{x}(kT) + \mathbf{B}\mathbf{y}(kT) \tag{1}$$

$$\mathbf{u}(kT) \quad = \quad \mathbf{C}\mathbf{x}(kT) + \mathbf{D}\mathbf{y}(kT) \tag{2}$$

where $\mathbf{y}$ represents the vector of inputs to the system and $\mathbf{u}$ represents the vector of outputs of the system. For the particular state space difference Equations (1) and (2), the vector $\mathbf{x}$ contains the states of the system and the matrix fourtuple ($\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$) is commonly referred to as the realization of the system. Furthermore, when the discrete-time system represented by (1) and (2) is implemented on a digital computer, issues such as roundoff and quantization arise. We shall refer to such an implementation as a *digital filter*.

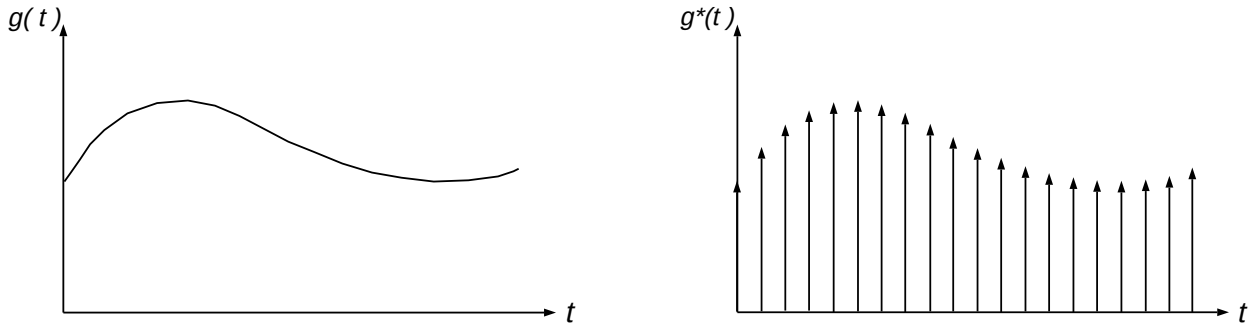
In the same way that the Laplace transform of a continuous-time signal is a frequency domain representation of that signal, the $z$-transform of a discrete-time signal is also such a frequency domain representation. The transfer function of a linear, time-invariant, finite-dimensional, discrete-time system describes the input-output relationship of that system as a ratio of polynomials in $z$, the ratio being the $z$-transform of the output signal to that of the input signal. Further, just as the transfer function of a linear, time-invariant, finite-dimensional, continuous-time system is considered stable (in the strictly Hurwitz sense) if all of its poles are located in the open left half complex $s$ plane, the transfer function of a linear, time-invariant, finite-dimensional, discrete-time system is stable if all of its poles are located within the open unit circle in the complex $z$ plane.

For digital control systems, a typical situation is to have a feedback configuration which involves a combination of continuous-time systems, such as structures, along with digital filters. Much of the controller processing takes place in the digital part of the feedback system. Inputs to this part of the system may arise from sampled measurements of the continuous-time elements via A/D devices. Control actions or forces used in the continuous-time system are constructed from the results of digital calculations subsequently processed by D/A devices. Such systems which include both continuous-time and discrete-time elements are often referred to as *sampled-data systems*.

### 2.1.1 Nyquist Sampling Theory

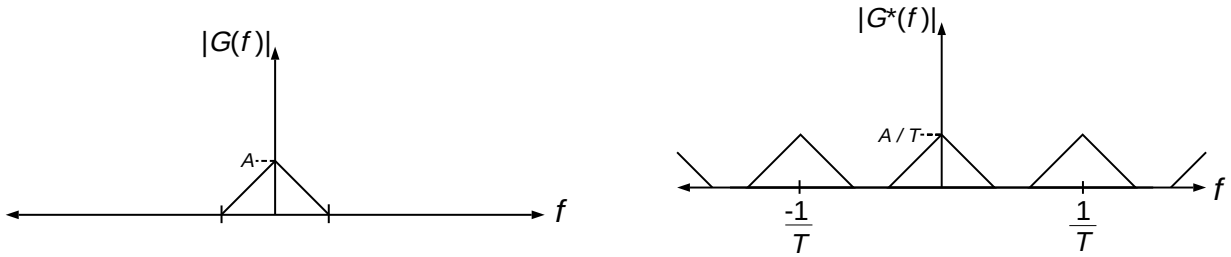
The intention of sampling a continuous-time signal is usually to capture the information contained within that signal in the form of a sequence of sample values that can then be more easily stored, transmitted or filtered. Suppose a signal $g(t)$, as shown in Figure 1a, is sampled uniformly once every $T$ seconds. The sampling processes results in a sequence of finite numbers that represent the amplitudes of the sampled signal at the sampling instants. However, this sequence is not convenient for frequency domain analysis. Therefore, frequency domain analysis of sampled signals is usually done by modelling sampling with a process known as *impulse modulation*.[6], [7] In impulse modulation, sampling is treated as the product of the signal to be sampled with the *ideal sampling function* which is a train of unit area dirac delta functions evenly spaced in time at intervals of $T$. If the signal $g(t)$ shown in Figure 1a is sampled using impulse modulation, the resulting signal $g^*(t)$, as illustrated in Figure 1b, is a sequence of dirac delta functions each centered at the sampling instants, with areas equal to the amplitude of the sampled signal at those instants.

Suppose the frequency spectrum of the original signal $G(f)$ is as shown in Figure 2a. The resulting frequency spectrum of the ideally sampled signal, $G^*(f)$, as illustrated in Figure 2b, is a never ending periodic replication of the original spectrum $G(f)$ at frequency intervals




**(a)** Continuous-Time Signal. **(b)** Sampled Signal.

Figure 1: Ideal Sampling of a Continuous-Time Signal.




**(a)** Spectrum of Unsampled Signal. **(b)** Spectrum of Signal After Ideal Sampling.

Figure 2: Effect of Sampling in the Frequency Domain.

of $\frac{1}{T}$, each replicant scaled by a factor of $\frac{1}{T}$. If a continuous-time signal has no significant frequency components greater than a value of half the sampling frequency (known as the Nyquist frequency) then, as illustrated in Figure 2b, the replicants in the sampled signal spectrum will not overlap. In this case, the frequency spectrum of the original continuous-time signal, from which the samples where derived, is contained undistorted, but scaled, within the sampled spectrum. The original continuous-time signal could be recovered exactly by passing the samples through an ideal low pass filter with a cutoff frequency of $\frac{1}{2T}$ and a constant passband gain of $T$.

If the sampling rate is insufficient, namely if it is less than twice the highest significant frequency component in the sampled signal (known as the Nyquist rate), then a phenomenon known as aliasing occurs. In this case, the spectrum of the sampled signal contains overlapping replicants of the continuous-time signal's spectrum, as shown in Figure 3. Consequently, there is no way that the original signal could be recovered exactly from its samples; for the samples no longer contain exactly the information of the original signal, but instead some distorted version of that information.

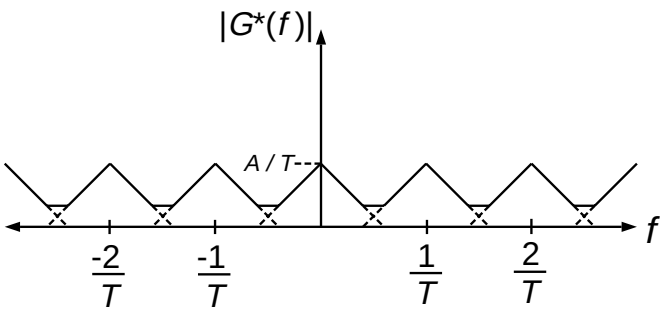



Figure 3: Spectrum of Ideally Sampled Signal Undergoing Aliasing.

### 2.1.2 Discrete-Time Equivalent Systems

The *discrete-time equivalent* representation of a continuous-time system is a discrete-time system whose output in some way approximates the sampled response of the continuous-time system when a given signal is the input to the continuous-time system and the sampled version of that same signal is the input to the discrete system.[6], [11] For example, a *step invariant* discrete-time equivalent system produces an output that is the same as the sampled output of a continuous-time system when the input to each system is the unit step function. An *impulse invariant* discrete-time equivalent system produces an output that is the same as the sampled output of the continuous-time system when the input to the discrete-time equivalent system is a unit pulse and the input to the continuous-time system is the unit impulse.

One technique for the development of discrete-time equivalent systems is numerical integration. In this method, the differential equations that describe the continuous-time system are numerically integrated one step forward in time with an integration step of $T$, from time $kT$ to time $kT + T$. In the integration, only the inputs to the continuous-time system at times $kT$ and $kT + T$ are used.

One of the most common ways to form a discrete-time equivalent system is by application of the *bilinear transformation* to the transfer function of a continuous-time system. The bilinear transformation yields a set of difference equations based on the numerical integration of the continuous-time system differential equations using the trapezoidal rule. In the frequency domain, if the transfer function of a linear, continuous-time, time-invariant, finite-dimensional system is represented by $H(s)$, the resulting discrete-time equivalent system has a z-transform representation, $\hat{H}(z)$, that is formed simply by the substitution of the quantity $\left(\frac{2}{T}\frac{z-1}{z+1}\right)$ for the complex frequency variable $s$ in $H(s)$.

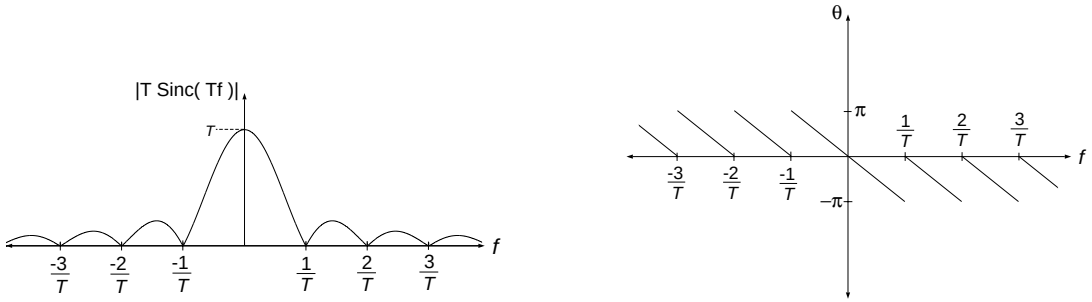
From a frequency domain standpoint, the bilinear transformation has the effect of 'squeezing' the entire frequency response (0 Hz to $\infty$ Hz) of the continuous-time system between 0 Hz and $\frac{1}{2T}$ Hz for the discrete-time equivalent system. This transformation results in the phenomenon of 'warping' in which there is good reproduction of the frequency response of the continuous-time system for frequencies well below $\frac{1}{2T}$ but increasingly poor reproduction at higher frequencies. The great advantage of the bilinear transformation is that stability

characteristics are preserved in the mapping from the complex $s$ plane to the complex $z$ plane. That is, stable and unstable poles for the continuous-time system are mapped respectively to stable and unstable poles in the discrete-time system, and vice versa.

### 2.1.3 Holds

A hold is a device which constructs a continuous-time signal, whose sampled version is the input to the hold. One class of hold devices uses polynomic interpolations among samples to construct a continuous-time signal that the samples represent. The simplest and most commonly used type of polynomic hold is the zero-order hold (ZOH) which uses a zero-order polynomial, a constant, to estimate the continuous-time signal between samples. In the zero-order hold, for each sampling interval, the discrete pulse level is 'held' at that same constant level between sampling instants. Such a hold is the type of device that is ordinarily found in D/A converters.

If the zero-order hold is analyzed in the frequency domain, its transfer function has a magnitude characterized by the function $|T\text{Sinc}(Tf)|$ as illustrated in Figure 4a ( $\text{Sinc}(a)$ is defined as $\frac{\text{Sin}(\pi a)}{\pi a}$). If the frequency content of the baseband replicant is well below the Nyquist frequency as in Figure 2b, then the zero-order hold will pass the baseband replicant with a gain of approximately $T$ (cancelling the factor of $\frac{1}{T}$ introduced by the sampling process) and essentially annihilate the higher frequency replicants, each being centered at the nulls located at frequency multiples of $\frac{1}{T}$. More importantly, the phase characteristic of



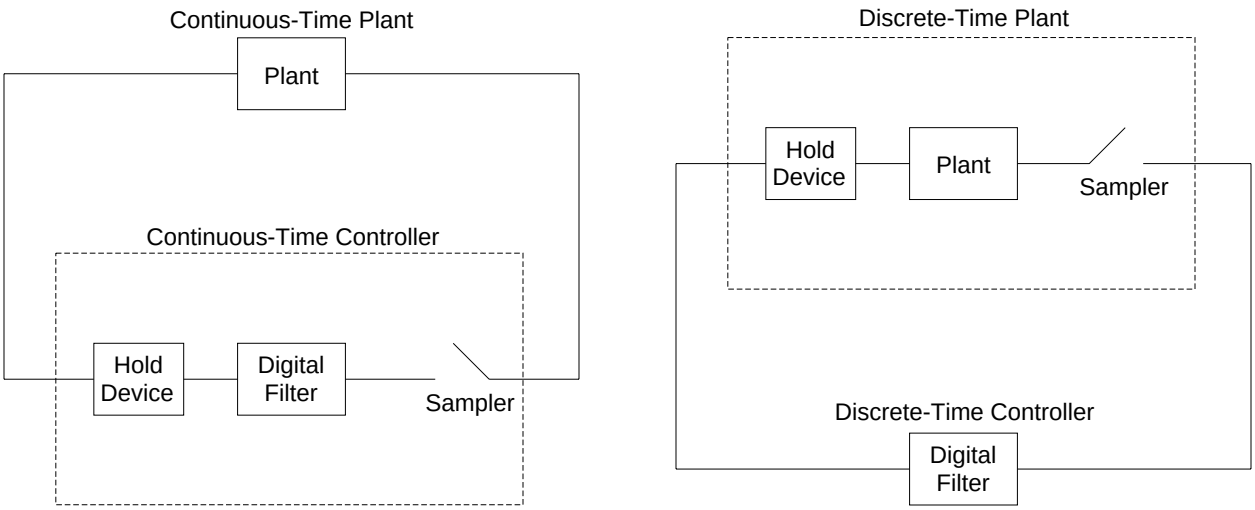

**(a)** Magnitude. **(b)** Phase.

Figure 4: Frequency Domain Characteristics of the ZOH.

the ZOH is linear with frequency, as illustrated in Figure 4b. Such a phase characteristic corresponds to a time delay with the slope corresponding to a delay of the signal equal to $\frac{T}{2}$. As will be discussed in Section 3, this delay must be considered in the design and implementation of digital control systems.

## 2.2 Digital Control System Design

There are two fundamentally different approaches to the design and implementation of digital control schemes for the control of continuous-time plants:[6]




**(a)** Emulation. **(b)** Direct Digital Design.

Figure 5: System Models for Digital Control System Design.

1. **Emulation.** The method of emulation involves two steps. First, a suitable controller is designed using continuous-time control techniques. Second, the continuous-time controller design is then approximated or 'emulated' with a discrete-time equivalent digital filter, typically using a bilinear transformation. Such a configuration is illustrated in Figure 5a. The controller samples the measured outputs of the plant and passes the samples through a digital filter implemented on a computer. The output of the digital filter is then passed through a hold device to create a continuous-time signal which becomes the control input to the plant. Thus, the series combination of sampler, digital filter and hold emulate the operation of a continuous-time controller. Typically, if the sampling rate of the digital controller is greater than about 25 times the closed-loop system bandwidth, the discrete equivalent system will adequately represent the behavior of the emulated continuous-time system over the frequency range of interest.

   **Remark**: For multi-input, multi-output plants, the notion of bandwidth may have different meanings in different circumstances.[9] It suffices here to point out that the upper singular values of the loop gain matrices may be used to obtain practical bandwidth estimates.

2. **Direct Digital Design.** In the technique of direct digital design, like that of emulation described above, the measured outputs of the plant are sampled, the samples are passed through a digital filter, the output of the filter is passed through a hold device, and this signal is used as the control input to the plant. However, this method differs from that of emulation in that the configuration is viewed as shown in Figure 5b. Although the system components are the same as in the case of emulation, and are also interconnected in the same way, the mathematical analysis of the system and control system design are conducted differently. Neglecting modelling errors of the continuous-time plant itself, the series combination of the hold device, continuous-time plant and sampler is itself a discrete-time system and can be viewed as the 'discrete-time plant'.

The discrete-time plant can then be controlled by applying discrete-time feedback design techniques that parallel those for classical control of continuous systems.[6] For example, once the discrete-time plant model is available (*e.g.* as a transfer function or in state space form), a discrete-time control scheme can be designed to achieve control objectives with techniques such as closed loop pole placement in the complex $z$ plane, discrete-time optimal control methods and discrete-time robust control theory. For direct digital design, the minimum sampling rate required is twice the bandwidth of the continuous-time plant that is being controlled. If such a requirement is not met, higher frequency plant dynamics would be aliased to lower frequencies after sampling; the digital control system would respond to them as if they were dynamics at the aliased frequency, possibly causing deteriorated or even unstable system behavior.

In the direct digital design method, the delay of $\frac{T}{2}$ introduced by the zero-order hold is explicitly incorporated into the model of the plant and therefore into the design of the control system. In addition, this method does not require an approximation using a discrete-time equivalent representation of a continuous-time system as in the case of design by emulation. In many control design applications, these differences allow the sampling rate required for satisfactory performance of a direct digital design controller to be significantly less than that required for one designed using emulation. However, as will be discussed further in Section 3.2, there are several issues that may dictate that a very high sampling rate be required for successful control implementation. In particular, for systems that are perturbed by random disturbances, digital controllers used to control these systems (whether designed using the direct method or by emulation) will require a sampling rate at least 10 to 20 times the bandwidth of the system in order to perform well. For many digital control applications, the sampling rate required may be high enough to minimize the advantages of direct digital design over emulation. Thus, depending on the specific application, it may be simpler and equally adequate to use emulation as the method of controller design.

# 3 Digital Control Implementation Issues

Once a filter is designed for use in a digital control system, it must be implemented on a computer. For multi-input, multi-output systems, perhaps the most convenient way to implement the digital filter is by realizing it in a state space form. Performing the arithmetic to implement a state space realization of a digital filter is not elaborate. However, there are many practical considerations that need to be addressed, including such things as time delay, sampling rate and I/O devices.

## 3.1 Time Delay in Digital Control Systems

The term *time delay* has been used in reference to many different phenomena in active structural control. For example, the term is often used when accounting for the phase lag introduced by unmodelled dynamics in a control system loop.[3], [4], [10] The introduction of phase lag that is not taken into account in control design can have a significant effect on the performance and stability of the system. Over the frequency range of interest, if the phase lag introduced by the unmodelled dynamics is approximately linear with frequency,

and the magnitude response constant, then the dynamics may be adequately modelled as a pure time delay.

However, an important issue in the digital implementation of structural control strategies is the true, pure time delay introduced by the computer controller. There are two sources of time delay in the practical implementation of digital control systems.

1. **Latency.** When implementing digital controllers, there are often instances in which it is required that a control signal be output at a given time instant, and that its calculation be partially based on measurements that are to be taken at that same instant. This situation occurs when a controller is implemented which has a transfer function numerator of degree equal to that of the denominator, or equivalently, a state space representation in which $\mathbf{D}$ matrix calculations are required to be performed. A reality of digital control implementation is that it is not possible to sample the plant measurement $\mathbf{y}$, perform required calculations, and output a control signal $\mathbf{u}$ at the same instant. Thus, in such instances, there will be an unavoidable pure time delay introduced between the sampling of the measurement at a given time instant, say $kT$, and the output of the control signal $\mathbf{u}$ at what should be the same instant $kT$. This delay is called *latency* and is due to the time required for A/D conversion, computation, and D/A conversion.[6]

   However, this delay due to latency for the output of a command signal at a time $kT$ can be minimized by performing during the time between instants $kT - T$ and $kT$, all calculations that do not require the measurement $\mathbf{y}$ at time $kT$ but only the state of the system and measurements at time $kT - T$. Then, when measurements are performed at time $kT$, the remaining calculations can be performed and the control signal $\mathbf{u}$ sent out. In the case of implementation of state equations, the state vector $\mathbf{x}$ for time $kT$ is calculated using Equation (1) and multiplied by $\mathbf{C}$ during the interim between $kT - T$ and $kT$. This quantity is then saved. At the next sampling instant, the measurement $\mathbf{y}$ will be sampled for time $kT$, multiplied by $\mathbf{D}$ and added to the previously computed quantity (*i.e.* $\mathbf{Cx}(kT)$). This result is then sent to the plant. If A/D and D/A rates and processor computational speeds are high enough, the delay may be small and perhaps negligible. This delay was actually measured to be about 200 $\mu$sec for the control system implemented in the experiment discussed in Section 5.

   Interestingly, if the $\mathbf{D}$ matrix multiplications are not required, the delay associated with latency may not be present. In fact, as soon as the $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ matrix multiplications and additions are completed, it is possible to output the control signal in advance of the next sampling instant, resulting in a phase lead.

2. **ZOH.** As discussed above, if the method of emulation is used for control system design, the ZOH introduces a delay of $\frac{T}{2}$ in the control loop. If the sampling rate is increased, the delay decreases proportionally. Therefore, if a sufficiently high sampling rate is achieved, the phase lag caused by the ZOH may be negligible and hence ignored.

If the time delay introduced by the digital control system is not short enough to be neglected, the delay may need to be accounted for explicitly in the model of the system. Approximations of the time delay can be difficult to implement in design of the control

system, primarily because the better the model of the delay, the higher the order of the approximation. This results in a higher order controller which may be undesirable due to the accompanying increase in computational time.

## 3.2 Sampling Rate

The sampling rate that is achievable by a digital control system is limited by such things as the rate at which A/D and D/A conversions can be performed, the speed of the processor and the number of calculations required to be performed by the processor during a sampling cycle. There are many factors that must be considered when evaluating the sampling rate that is required for satisfactory performance of a digital control system.

1. **Aliasing.** As discussed in Section 2.1.1, if a continuous-time signal is sampled and contains significant frequency components greater than the Nyquist frequency, aliasing will occur and cause a distortion of the frequency composition of the original signal as represented by its samples. Aliasing of measured quantities in digital control applications may lead to degraded or even unstable system performance. Thus, a digital control system needs to be implemented so that significant aliasing of the measured signals is prevented.

   One method of eliminating aliasing is to sample at a fast enough rate that the highest significant frequency component of the measured signal, including measurement noise, is less than the Nyquist frequency. This was the approach taken in the experiment discussed in Section 5.

   Another method for reducing or eliminating aliasing is to prefilter the continuous-time signal with a low-pass filter, often called an *anti-aliasing* filter, prior to sampling. The implementation of such filters in digital control applications must be done with great care however, taking into account several factors:

   (a) The anti-aliasing filters should have a cutoff frequency above the bandwidth of the system.

   (b) The cutoff frequency of the filter should be sufficiently below the Nyquist frequency so that the frequency composition of the filtered measurement will be negligible at and above the Nyquist frequency. Depending on the filter type used and the performance requirements of the system, this constraint may require that the cutoff frequency of the anti-aliasing filter be many times and perhaps as much as an order of magnitude below the Nyquist frequency.

   (c) A critical consideration in the implementation of anti-aliasing filters is accounting for the phase lag they introduce. The lag could be explicitly accounted for by including models of each filter in the control system design. This might cause an increase in the order of the controller equal to the sum of the orders of each of the anti-aliasing filters, which may be undesirable from a control implementation standpoint. If models of each anti-aliasing filter are not explicitly included in the system model, their cutoff frequencies should be sufficiently higher than the closed loop bandwidth of the system such that the amount of phase lag introduced

at dynamically significant frequencies is negligible. Depending on the order and type of filters used, this may require that the cutoff frequency be anywhere from several times to an order of magnitude above the bandwidth of the system.

Thus, no matter which approach is taken in order to eliminate performance concerns regarding aliasing, it will be necessary in most applications to sample at a rate much greater than twice the bandwidth of the system. However, the use of an anti-aliasing filter changes a measured physical variable to one that is not measured directly and may have consequences from the standpoint of robustness.

2. **Smoothness.** If a ZOH is used as the D/A device for a digital controller, there will be step discontinuities in the control signal at each instant the value of the ZOH changes. The relative smoothness of the output of the ZOH will be directly related to the sampling rate used. If a slower sampling rate is used in the controller, then there will be relatively larger discontinuities at each step. Likewise, faster sampling rates will result in smaller discontinuities at each step. The amount of smoothness required by a digital control system will depend on the specific application. For example, use of rough control signals when controlling hydraulic actuators may result in damage to the hydraulic systems over time. Thus, smooth control signals should be used when commanding a hydraulic actuator, requiring a sampling rate 6 to 40 times the closed loop bandwidth of the system, depending on the specific application.[6] If a slow sampling rate must be used in such a control application, resulting in a relatively rough control signal, one method for reducing the degree of roughness in the signal is to use a low-pass filter on the output of the ZOH. However, as in the case of anti-aliasing filters, this will result in an introduction of phase lag that must be accounted for in the model of the system.

3. **Control With Random Disturbances.** The use of a well designed continuous-time controller to reject random disturbances represents an upper bound of possible performance for digital controllers. The reason for this is that the sampled values of system measurements only capture the behavior of the system at the sampling instants and therefore do not completely represent the effects of the continuous-time disturbance. Therefore, the performance of a well designed continuous-time controller will represent the best control performance possible. When compared to this upper limit of performance, a general rule of thumb is that the digital controller should have a sampling rate of 20 times, but no less than 10 times the system bandwidth in order to perform comparably.[6] Compared to the best possible continuous-time controller, the performance degradation of a digital control system in random disturbance rejection as a function of sampling rate will vary depending on the specific application.

4. **Emulation.** If emulation is used as the method of digital control system design, there are additional reasons why it is desirable to have a high sampling rate. First, a greater sampling rate will increase the frequency up to which equivalent performance between the continuous-time controller and its digital 'emulation' is maintained. Second, if the delay introduced by the ZOH is not explicitly accounted for in the system model, increasing the sampling rate decreases the time delay produced by the ZOH.

Clearly, there are many issues involved with the selection of the sampling rate for a control system. While there are some guidelines regarding the selection of the sampling rate, the actual sampling rate that will yield satisfactory performance of the controller is dependent on many factors particular to the system being controlled.

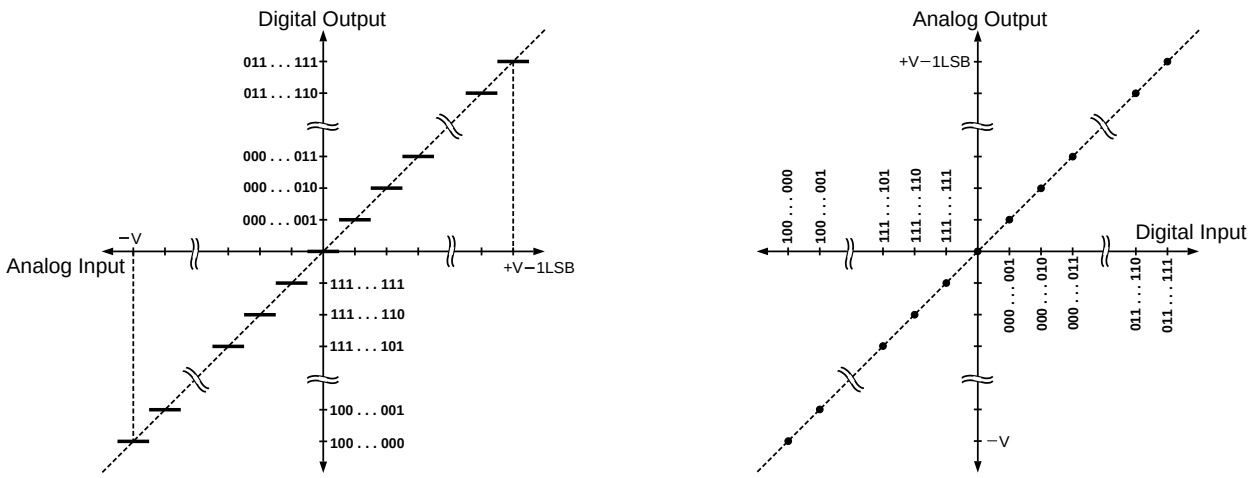
## 3.3 I/O Devices

The purpose of A/D and D/A devices in a sampled-data system is to realize a mapping from the analog signals that occur in the continuous-time portion of the system to the digital signals which represent them in the discrete-time portion, and conversely. In a precise, technical sense, these mappings cannot be linear according to the tenets of modern algebra. However, in certain graphical displays of the mappings, functions which are nearly straight lines appear. For example, an A/D device converts an analog signal into a binary digital number according to some relationship that approximates a straight line, as illustrated in Figure 6a. D/A devices convert a binary digital number to an analog signal by a relationship such as that shown in Figure 6b. The actual mappings will not be straight lines for two primary reasons:

1. **Quantization Noise.** There are only a finite number of binary numbers that can be used to represent the entire range of analog inputs to an A/D device. Thus each binary output level of the A/D converter will represent a range of possible input levels resulting in an ambiguity as to the analog voltage that the binary output represents. This ambiguity is commonly referred to as quantization error. For an A/D device with an output binary representation of n bits, there are $2^n$ possible unique binary outputs. If the A/D device has a range of analog inputs of $\pm V$, the magnitude of the quantization error will be at least $\frac{|V|}{2^n}$. If the I/O relationship of the A/D were as depicted in Figure 6a, and if n were large enough so that $2^n$ was very large, then the quantization error could be modelled as a random variable[7] uniformly distributed between $\pm\frac{V}{2^n}$. Clearly, the quantization error will decrease as the number of bits of binary representation of the output increases. The quantization errors can be treated as noise sources that directly corrupt the plant measured quantities $\mathbf{y}$ and control signals $\mathbf{u}$ respectively.

   One way of minimizing the magnitude of quantization noise is to amplify the quantity to be measured so that the input signal to the A/D converter spans the full scale of the input range of the device. Once the digital signal from the A/D converter is received by the digital filter, the signal can then be scaled down mathematically by the amount that the input signal was originally amplified. In this way, the quantization error will be smaller in proportion to the input signal.

2. **Calibration.** For both types of devices, the output-input relationships illustrated in Figures 6a and 6b approximate a straight line. However, the output-input relationships for these devices will not be precisely as depicted in these figures since it is not possible to fabricate these components exactly. For example, in the case of the A/D device depicted in Figure 6a, each step may not be exactly centered on the line between 0 and full scale.




**(a)** A/D Converter. **(b)**D/A Converter.

Figure 6: Input-Output Representations for I/O Devices. (Binary representation is in 2's complement form. LSB refers to the magnitude of the least significant bit).

Nevertheless, it may be possible to perform a software calibration to improve the relationship between the output and input of a given I/O device so that it will more closely correspond to the intended relationships such as those depicted in Figures 6a and 6b. In particular, in many instances it may be critical that the output of a device be equal to 0 when the input is 0. The actual characteristics of the devices can be established by supplying known inputs to each device and measuring the outputs. Functions can then be added to the digital filter at the point where the output of the A/D devices enters the filter or the point at which the results are sent to the D/A converter to compensate for imperfections and bring the effective input-output behavior of these devices more into keeping with the desired relationships like those illustrated in Figures 6a and 6b.

# 4 Digital Controller Hardware Implementation

One typical way in which digital control schemes were implemented in the past was through the use of data acquisition boards in the expansion bus of a PC. In this configuration, the data acquisition board would be programmed or commanded to take samples of system measured quantities at regular intervals. When available, the samples of the measured quantities would be passed to the PC's main CPU through the I/O space of the PC. The PC would then perform the arithmetic calculations required for implementation of the digital filter and forward the results through the I/O space to the D/A devices for output conversion. These continuous-time signals would then be used as the control inputs to the plant.

This configuration has many drawbacks. The time required to perform all of the A/D and D/A operations and pass these quantities over the I/O space of the PC, as well as perform the control algorithm computations on the PC, may require undesirably large sampling periods and induce unacceptable time delays. Also, with such an equipment configuration, it is

very difficult to create a scheme which allows the operator to monitor and interact with the controller while the PC is engaged in performing the control computations.

A more powerful arrangement for implementation of a digital control system is realizable through the use of one of many different DSP boards available that are placed in the expansion bus of a personal computer. State of the art DSP chips allow for very fast computational speeds as well as dedicated processing. The DSP boards have A/D and D/A converters locally on the board which reduces the delay involved in the transmission of signals between the processor and the I/O devices. Further, such configurations enable the control and monitoring of the DSP board by the PC in a supervisory control scheme. This allows the DSP board to conduct the actual control processes while the PC oversees the operation of the DSP board and provides system performance information and advisories to the operator.

## 4.1 TMS320C30 Digital Signal Processor

The control system employed in the EE/SDCL utilizes the Real-Time Digital Signal Processor System made by Spectrum Signal Processing, Inc. It is configured on a board that plugs into a 16-bit slot in a PC's expansion bus and features a Texas Instruments TMS320C30 Digital Signal Processor chip, RAM memory and on board A/D and D/A systems. The TMS320C30 DSP chip has single-cycle instructions, a 33.3 MHz clock, a 60 ns instruction cycle and can achieve a nominal performance of 16.7 MFLOPS. A special feature of the chip that allows a floating point multiplier and adder to be used in parallel yields a theoretical peak performance of 33.3 MLFOPS. Moreover, this board has a number of built-in functions that make it ideal for control applications. For example, there are notch filters to cancel mechanical resonances, adaptive Kalman filter algorithms to reduce sensor noise, vector control algorithms for real-time axis transformation and fuzzy set control algorithms.

In addition, the on-board A/D system has two channels, each with 16 bit precision and a maximum sampling rate of 200 kHz. The two D/A channels, also with 16 bit precision, allow for even greater output rates so as not to be limiting. An expansion I/O daughter board, which connects directly to the DSP board, provides an additional four channels of input and two channels of output capability, each with 12 bit precision. All four input channels share the same conversion device, which is the limiting factor for the board's sampling rate. The maximum sampling rate for the daughter board is 200 kHz for one channel and proportionally less for multiple channels, with a rate of 50 kHz per channel if all four channels are used. The maximum rate for each of the two daughter board output channels is 300k samples/second. Additional daughter cards may be added to the system to further expand the system's I/O capabilities. Clearly, with the high computation rates of the DSP chip and the extremely fast sampling and output capability of the associated I/O system, high overall sampling rates for the digital control system are achievable.

As mentioned previously, the board plugs into a 16 bit expansion slot in a PC, which allows for communication between the DSP board and the PC through the I/O space of the PC. The PC is used to download the control code to the DSP board through this I/O interface. Further, while the DSP board runs the control algorithm, a supervisory program running on the PC can monitor the performance of the control system, monitor and display measured quantities, and allow the operator to send commands to the DSP board, starting and stopping the controller or changing control parameters. This configuration allows for a

very powerful and flexible implementation of a digital control system for structural control.

## 4.2 Software Considerations

Once the discrete-time control system is designed, it is implemented using the Real-Time Digital Signal Processor System. The code for these programs can be written directly in the C programming language. The code is compiled, linked with library functions and made into executable files on the PC. The PC is then used to download the control code to the DSP board through the PC's I/O interface. The SPOX operating system provides standard I/O library support for the C language so that pre-existing C programs will execute with limited code modification. SPOX also provides a library of standard DSP functions which free the operator from writing lower level code such as device drivers for managing incoming data. Applications written in C under SPOX are portable to other hardware environments supporting SPOX.

For the EE/SDCL implementation, the timing of the sampling is done using a timer on the DSP board which is set to send an interrupt to the processor at intervals equal to the sampling period. The interrupt calls an interrupt service routine which in turn sets a software flag demarcating the beginning of a sampling period. The control program, waiting for this flag, then executes the steps required for A/D conversion. When the A/D process is complete, the measurement $\mathbf{y}$ is multiplied by the $\mathbf{D}$ matrix, the result added to the previously computed quantity $\mathbf{Cx}$, as discussed in Section 3.1, and this result sent to the plant through the D/A converter.

In addition to the implementation of the control algorithm difference Equations (1) and (2), several other tasks are performed on the DSP board during controller operation. In particular, standard deviations are calculated for all measured quantities, averaged over many samples. These values are read iteratively by the supervisory program running on the PC and are continuously displayed on the PC display so the operator can monitor performance of the system. Further, as is done extensively in control system implementation in general as well as in structural control, at each sample instant, the magnitudes of certain measured quantities such as actuator force and building displacement are compared to maximum allowable values specified by the user.[14], [12] If at any instant the measured values exceed the maximum allowable values, the controller is immediately shut off and the command signal is set equal to zero. In addition, if the command output calculated by the control algorithm exceeds the range of the D/A devices, a controller shutdown also occurs. In this case, if the controller were allowed to continue to operate with a saturated command output, the control signal would effectively be corrupted by a noise signal equal in magnitude to the difference between the commanded output and the saturation level of the D/A device. Such a situation could have devastating effects on the performance and stability of the system.

If a shutdown occurs, the supervisory program on the PC, iteratively checking the condition of the controller, detects the shutdown and displays an advisory on the monitor for the operator. This feature is designed to protect the system and structure from damage due to excessive or unstable response caused by modelling errors, high ground excitation or mistakes in the hardware or software implementation of the controller. The supervisory program running on the PC also allows the operator to turn the controller on and off as well

as to change control parameters of the controller during its operation. Although changing parameters in the control algorithm while the controller is running may be desirable in some cases, it is generally not advisable unless a careful assessment is made of the effects of such changes during control operation.

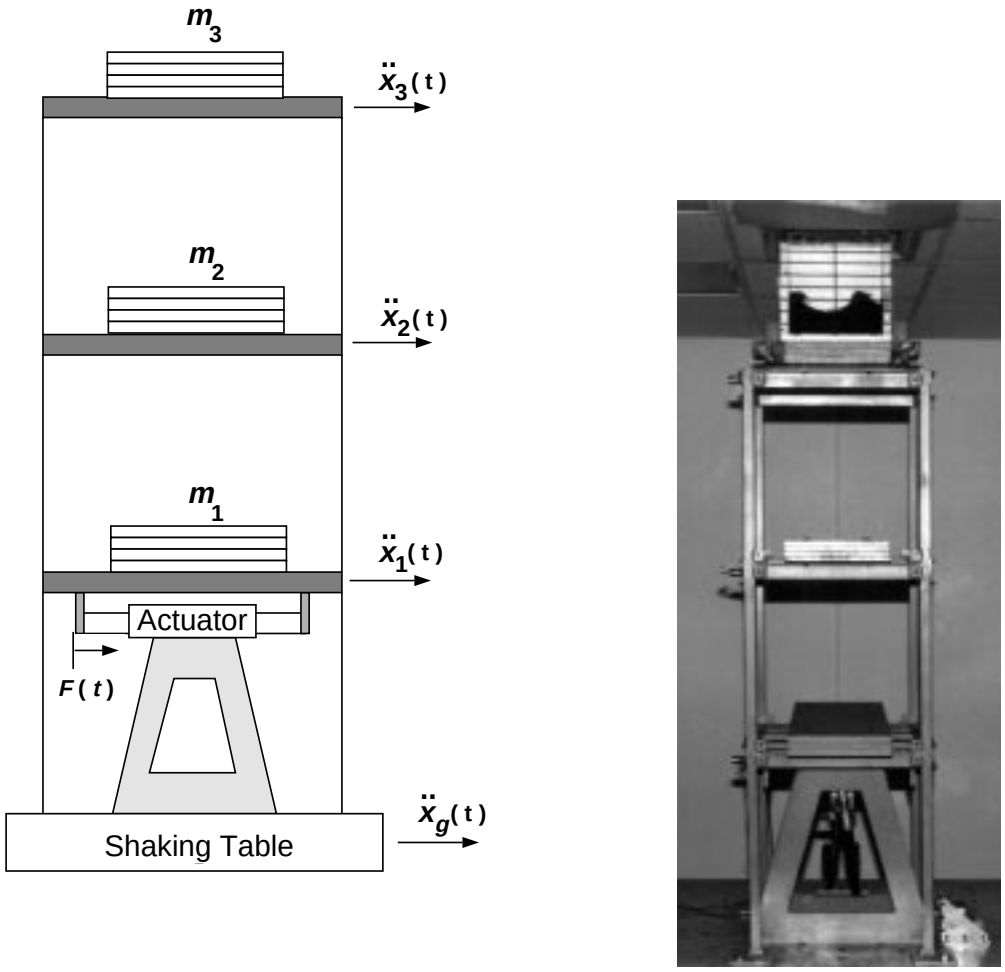
# 5 Experimental Verification

To demonstrate the performance of the DSP hardware for active structural control, tests were performed on a three-story, single-bay, model building (see Figure 7) in the Earthquake Engineering/Structural Dynamics and Control Laboratory (EE/SDCL) at the University of Notre Dame. The test structure is constructed of steel and is designed to be a scale model of the prototype building discussed in Reference [2]. The height of the model is 62 in. The columns are 1 in $\times$ 1 in $\times$ 1/8 in angle, and the floor masses are comprised of steel plates bolted to rigid floor beams. The total floor mass is 500 lb, distributed evenly between the three floors. The time scale factor is 0.2, making the natural frequencies of the model five times those of the prototype. The resulting frequencies for the uncontrolled three degree-of-freedom model are 5.7 Hz, 17.3 Hz, and 28.3 Hz.

The structure is excited by a uniaxial earthquake simulator to which it is mounted. The simulator consists of a hydraulic actuator/servovalve assembly that drives a 48 in $\times$ 48 in aluminum slip table mounted on high-precision, low-friction, linear bearings. The actuator is a 2.5 in bore Nopak hydraulic cylinder with a $\pm$ 3 in stroke, and the servovalve/amplifier is distributed by Continental Hydraulics. The actuator and table/bearing system are attached to a reaction mass, which is a 30 in $\times$ 60 in $\times$ 96 in sand-filled steel box sitting on 4 inflatable air-spring isolators. The hydraulic power unit for the simulator is a 26 gpm gear pump with a nominal supply pressure of 3,000 psi. The pump is driven by a 60 amp, 50 HP electric motor. A 2.5 gallon hydraulic accumulator is attached to the supply line to the actuator. The nominal capabilities of the simulator are: maximum displacement $\pm$2 in, maximum velocity $\pm$35 in/sec and maximum acceleration $\pm$ 4g's with a 1000 lb test load. The operational frequency range of the simulator is 0-50 Hz.

As shown in Figure 7a, the structure is controlled by a hydraulic actuator mounted between the first floor of the building and a steel frame which is attached to the seismic simulator table. The actuator consists of a 1.5 in bore Nopak cylinder with a $\pm$2 in stroke mounted with a servovalve having a 45 Hz bandwidth. A position sensor (LVDT) provides feedback for the control actuator and to measure the displacement of the first floor. The actuator is operated in displacement control mode. The force transmitted to the building by the control actuator is measured with a piezoelectric force ring manufactured by PCB Piezotronics, Inc. Accelerometers measure the absolute acceleration of each floor as well as the ground acceleration. Figure 7b is a picture of the structure as tested in the EE/SDCL during uncontrolled response.

The first step in the design and implementation of an active control strategy was to identify an accurate mathematical model of the structure to be controlled. A Tektronix 2630 Fourier analyzer was used in conjunction with an 80486-based PC to provide necessary time and frequency domain functions (*e.g.*, auto-correlation, cross-correlation, impulse response, power spectral density, energy spectral density, cross power spectrum, coherence, *etc.*). This




**(a)** Diagrammatic Representation. **(b)**Laboratory Realization.

Figure 7: Three Degree-of-Freedom Test Structure with Active Bracing.

analyzer has 12 base-band frequency ranges from DC-5 Hz to DC-20 kHz, 9 zoom ranges from $\pm 5$ Hz to $\pm 2$ kHz and built-in anti-aliasing signal conditioners. Voltage ranges span $\pm 55$ mV to $\pm 10$ V in 16 steps of 3 dB. The floating point averager can be set to record all possible cross spectra or any desired subset. Dynamic range is 75 dB, and the real-time bandwidth is 10 kHz. The system also has a built-in signal generator and a digital record/playback feature that provides high-speed, bidirectional transfer of digitized input signal between the 2630 and the PC's hard disk. This system is capable of obtaining vibrational mode shapes and frequencies, as well as transfer functions and their associated pole/zero patterns. The experimentally obtained transfer functions utilized herein were based on 10 averages of data sampled at 512 Hz. For maximum spectral resolution, a 4096 point FFT was used in conjunction with a Hanning window.

Based on the pole/zero patterns determined from the experimental transfer functions, a two-input/five-output state space model with 14 states was constructed. The inputs to the system were the ground excitation and the control force. The measured outputs were the acceleration of each floor, the displacement of the control actuator and the control force

applied to the structure. This state space model accounted fully for the inherent interaction between the structure and the controller. A detailed discussion of the importance of control-structure interaction in structural control design can be found in References [3] and [4].

Once an accurate state space model was obtained, a continuous-time, output-feedback controller was designed using $H_2/LQG$ methods.[15],[16],[17] The measured outputs on which the control action was based were the acceleration of each floor, the displacement of the control actuator and the control force applied to the structure. The primary goal of the controller design employed herein was to minimize the floor accelerations. An equivalent discrete-time controller was then calculated by direct application of the bilinear transformation. Because sample frequencies were on the order of 1 kHz, the resulting frequency responses of this discrete-time controller were essentially the same as that of the continuous-time controller, with no significant distortion due to warping over the frequencies of interest (*i.e.*, less than 100 Hz).

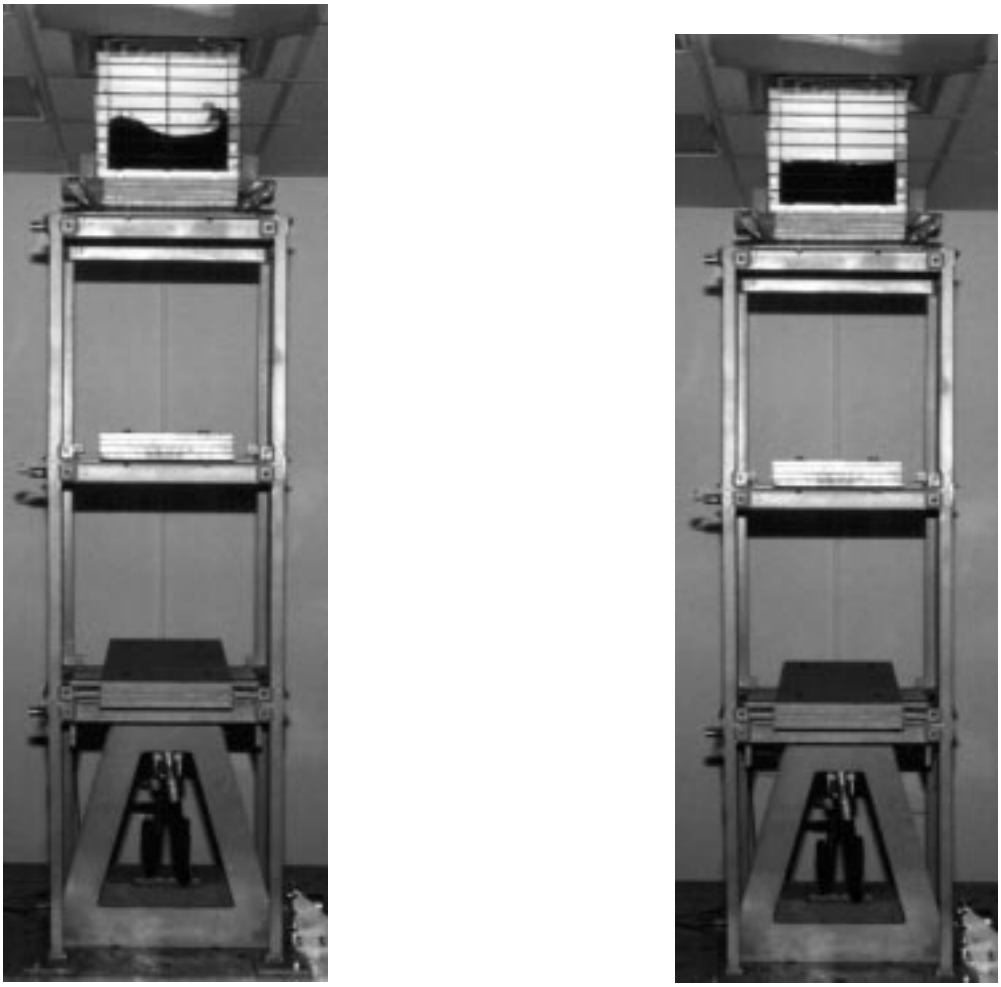
The implementation of the discrete-time controller was then made on the Spectrum Real-Time Digital Signal Processor System, which utilizes the TMS320C30 DSP chip as discussed in Section 4.1. The total time required within each sampling period for the five A/D reads, the associated conversion and calibration calculations, as well as for all state space calculations (for a 14 state system) and the D/A write, was less than 400 $\mu$sec. Thus, sampling rates at least as high as 1 kHz were completely feasible for the system. At this sampling frequency, aliasing and the time delay of $\frac{T}{2}$ introduced by the ZOH were negligible.

To obtain a visual assessment of the effectiveness of the control strategy, a sloshing tank was placed on the top floor of the structure. Figure 8a is a snap-shot of the uncontrolled structure responding to a broadband excitation. Here, large waves are present, with some of the liquid sloshing completely out of the tank. Note that the lines of the reference grid on the tank are at 1 in intervals. Figure 8b shows the controlled structure responding to the same broadband excitation. As is readily seen, the waves have been damped out by the action of the controller.

A quantitative measure of the performance of the system is obtained by examining the RMS responses of the structure. Table 1 provides a comparison between the RMS responses for the uncontrolled and controlled configurations of the structural system. The input in this test was a broadband ground excitation (0-100 Hz) with an RMS value of 36.3 in/sec$^2$. The *uncontrolled* configuration refers to the case in which the active bracing system is completely disconnected from the structure. The *zeroed control* configuration corresponds to the case in which the actuator is attached, but the command signal is set equal to zero. From the response of the zeroed configuration it is shown that the stiffness of the actuator has a significant effect on the displacement (97.3%) and some effect on the accelerations. Notice that with control, the absolute accelerations of the three floors are reduced by 37.8%, 56.4% and 61.0%, respectively, and the first floor displacement is reduced by 95.6%. Note that the controlled configuration also requires less force than the zeroed control case.

# 6 Conclusions

Implementation of active structural control strategies has become more practicable with the introduction of new hardware based on digital signal processing chips. Many issues partic-



**(a)** Uncontrolled Response. **(b)** Controlled Response.

Figure 8: Response of Water Tank Mounted on Third Floor of Structure.

ular to digital control must be thoroughly understood and resolved for successful controller implementation. Fast processors and I/O systems have allowed for very fast sampling rates in digital control implementation and have nearly eliminated the problem of computer time delays in the laboratory setting. With the TMS320C30 DSP chip, a sampling rate of 1 kHz has been achieved for control applications in the laboratory. Use of a dedicated DSP processor also allows for a supervisory program to be run on a supporting PC overseeing the performance and ensuring the safe operation of the controller. The performance of the digital control system based on acceleration feedback was experimentally verified using a three-story test structure experiencing an earthquake excitation. An active bracing system employing a hydraulic actuator was used to create the control force. Experimental results indicate that this control strategy is very effective in reducing the response of the structure. The absolute accelerations of the top two floors of the test structure were reduced by more than a factor of two, thus verifying the hardware performance. More extensive verification of the control implementation can be found in Dyke *et. al.*[5]

| Configuration | $\sigma_{x_1}$, in | $\sigma_{\ddot{x}_{a1}}$, in/s$^2$ | $\sigma_{\ddot{x}_{a2}}$, in/s$^2$ | $\sigma_{\ddot{x}_{a3}}$, in/s$^2$ | $\sigma_f$, lbf |
|---|---|---|---|---|---|
| Uncontrolled | $6.867e^{-1}$ | 62.74 | 70.65 | 91.33 | — |
| Zeroed Control | $1.871e^{-1}$ | 48.00 | 54.26 | 66.33 | 66.60 |
| Controlled | $3.020e^{-1}$ (95.6%)[a] | 39.29 (37.8%)[a] | 30.80 (56.4%)[a] | 35.59 (61.0%)[a] | 37.39 |

a. Percent reduction of response over uncontrolled case.

Table 1: Experimental RMS Values.

# Acknowledgment

This research is supported by National Science Foundation Grants No. BCS 90-06781 and No. BCS 93-01584.

# References

[1] ATC 17-1, Proceedings on Seismic Isolation, Passive Energy Dissipation, and Active Control, Applied Technology Council, Redwood City, California, 1993.

[2] Chung, L.L., Lin, R.C., Soong, T.T., and Reinhorn, A.M., "Experiments on Active Control for MDOF Seismic Structures," Journal of Engineering Mechanics, ASCE, Vol. 115, No. 8, 1989, pp. 1609-1627.

[3] Dyke, S.J., Spencer Jr.,B.F., Quast, P. and Sain, M.K. 1994, "Protective System Design: The Role of Control-Structure Interaction," Proceedings of International Workshop on Structural Control, Honolulu, Hawaii, August 5-7, 1993.

[4] Dyke, S.J., Spencer Jr.,B.F., Quast, P. and Sain, M.K. "The Role of Control-Structure Interaction in Structural Control" Journal of Engineering Mechanics, 1994, in press.

[5] Dyke, S.J., Spencer Jr., B.F., Quast, P., Sain, M. K., and Kaspari, D.C., "Experimental Verification of Acceleration Feedback Control Strategies for MDOF Structures" Proceedings of the Second International Conference on Computational Mechanics, Athens, Greece, 1994.

[6] Franklin G.F., Powell J.D., Workman M.L., Digital Control of Dynamic Systems 2nd Ed., Addison-Wesley Publishing Company Inc., 1990.

[7] Haykin S.,Communication Systems, Second Edition, John Wiley & Sons Inc., 1983.

[8] Housner, G.W. and Masri, S.F., Eds., Proceedings of the U.S. National Workshop on Structural Control Research, USC Publications No. M9013, University of Southern California, October 25-26, 1990.

[9] Maciejowski, J.M., Multivariable Feedback Design, Addison-Wesley Publishing Company, 1989.

[10] McGreevy, S., Soong, T.T. and Reinhorn, A.M., "An Experimental Study of Time Delay Compensation in Active Structural Control," Proceedings of the 6th International Modal Analysis Conference and Exhibits, Vol. 1, 1988, pp. 733-739.

[11] Oppenheim A.V., Schaffer R.W., Digital Signal Processing, Prentice Hall Inc., 1975.

[12] Reinhorn, A.M., Soong, T.T., Riley, M.A., Lin, R.C., Aizawa, S. and Higashino, M., " Full Scale Implementation of Active Control. II: Installation and Performance.", Journal of Structural Engineering, ASCE, Vol. 119, No. 6, June 1993, pp 1935-1960.

[13] Soong T.T., Active Structural Control: Theory and Practice, Longman Scientific and Technical, Essex, England, 1990.

[14] Soong, T.T., Reinhorn, A.M., Wang, Y.P, Lin, R.C., " Full Scale Implementation of Active Control. I: Design and Simulation.", Journal of Structural Engineering, ASCE, Vol. 117, No. 11, November 1991, pp 3516-3536.

[15] Spencer B.F. Jr., Suhardjo J. and Sain M.K., "Frequency Domain Control Algorithms for Civil Engineering Applications," Proceedings, International Workshop on Technology for Hong Kong's Infrastructure Development, Hong Kong, 1991, pp. 169-178.

[16] Spencer Jr., B.F., Suhardjo, J. and Sain, M.K., 1993, "Frequency Domain Optimal Control Strategies for Aseismic Protection." Journal of Engineering Mechanics, ASCE, Vol. 120, No. 1, 1994, pp. 135-159.

[17] Suhardjo, J., Spencer Jr., B.F. and Kareem, A., "Frequency Domain Optimal Control of Wind Excited Buildings." Journal of Engineering Mechanics, ASCE, Vol. 118, No. 12, 1992, pp. 2463-2481.