Chapter 5

Real Time Active Control

5.1 Introduction

To simulate a desired load impedance, a specific control task must be performed. Achieving this control requires real time adaptive digital filtering and this can be performed using a digital signal-processing module. In our case the frequency range of interest is from around 10Hz to 2 kHz so that sample rates of a few kHz can be used. This requires extensive computation and therefore efficient high speed Digital Signal Processing (DSP). The digital signal processing has been implemented using a C30 System DSP Board. This board is a full-length IBM PC/AT-compatible plug-in board featuring the Texas Instruments TMS320C30 floating-point. Even though all preliminary studies have already been presented, there are still a few more steps necessary before starting active control tests.

The pre-existing adaptive feedforward controller software at VAL, Virginia Polytechnic Institute and State University, was reprogrammed for the specific requirements of this study. Further, the digital filters were designed to reproduce the desired impedance condition. Real time tests for different desired dynamic load conditions (i.e.: different Z_d) could then be performed.

5.2 Evolution of the Preexisting Software

The Vibration and Acoustics Laboratories (VAL) at Virginia Polytechnic Institute and State University were already equipped with software using the C30 DSP board to perform adaptive feedforward control. This software is developed into two different parts. First, there is the C^{++} programming part (created mainly by Michael. W. Wenzel) that directly controls the DSP board. The real time computation and control is all carried out on this board. Secondly, there is the interface that links the user and the DSP board (created by Francois Charette). This interface has been created using Labview software and allows time signals to be readily analyzed and the performance of the control system to be monitored.

The existing adaptive feedforward controller software has been developed for active noise and vibration control [11], [18]. As we have seen in Chapter 2, the task of an active controller is basically to cancel vibration and/or noise at a given location for certain frequencies. To do so, the controller is directly fed with an error signal that it tries to reduce. This error signal comes from an error sensor located where the noise or the vibration has to be cancelled. In our case, the control task is slightly different. If we consider the impedance, Z, as the ratio of a force, f, over a velocity, v, (see Chapter 3), then we want to drive the actual ratio, Z=f/v, due to the disturbance signal (input voltage of the sample actuator), to a desired ratio, Z_d. To achieve this, the control actuator needs to be driven with the correct control signal. Therefore, to tailor our control task to the adaptive feedforward controller, we have to create an error signal with the actual force and velocity and the desired impedance (see chapter 3). In this arrangement, the adaptive minimization process using the filtered-X LMS algorithm can be used.

Figure 5.1 shows a diagram of the control task performed by the DSP board. The sample actuator is driven with a reference signal generated internally on the DSP board. The

control actuator is driven with a control signal that alters the force and velocity output of the sample actuator. This is equivalent to altering the structural impedance that the sample actuator drives. The processor monitors the ratio between the force and the velocity (i.e. the impedance) and alters the controller H until the measured impedance matches the desired impedance. If the test signal is a single frequency, then the digital control filter H simply changes the magnitude and phase of the test signal and uses this to drive the control actuator.

The error signal has been defined as,

$$E=Z_d v-f, \tag{5.1}$$

where the error E is calculated by multiplying the measured velocity, v, by the desired impedance, Z_d, and subtracting the measured force output, f. The controller H is driven with the reference signal and adapted using the filtered-X LMS algorithm (shown in detail in Chapter 2).

For the controller to be able to perform this error calculation, the C^{++} program for the DSP board had to be modified. These modifications were to allow the coefficients of the digital filter, that modeled the desired impedance (Z_d), to be downloaded on to the DSP board and used with the force and velocity signals to compute the error E. The Labview interface also had to be modified to be able to recognize the new variables created on the DSP board and also such that it allows the user to select and download on the DSP board the desired impedance (vector of coefficients) that he wants to simulate.

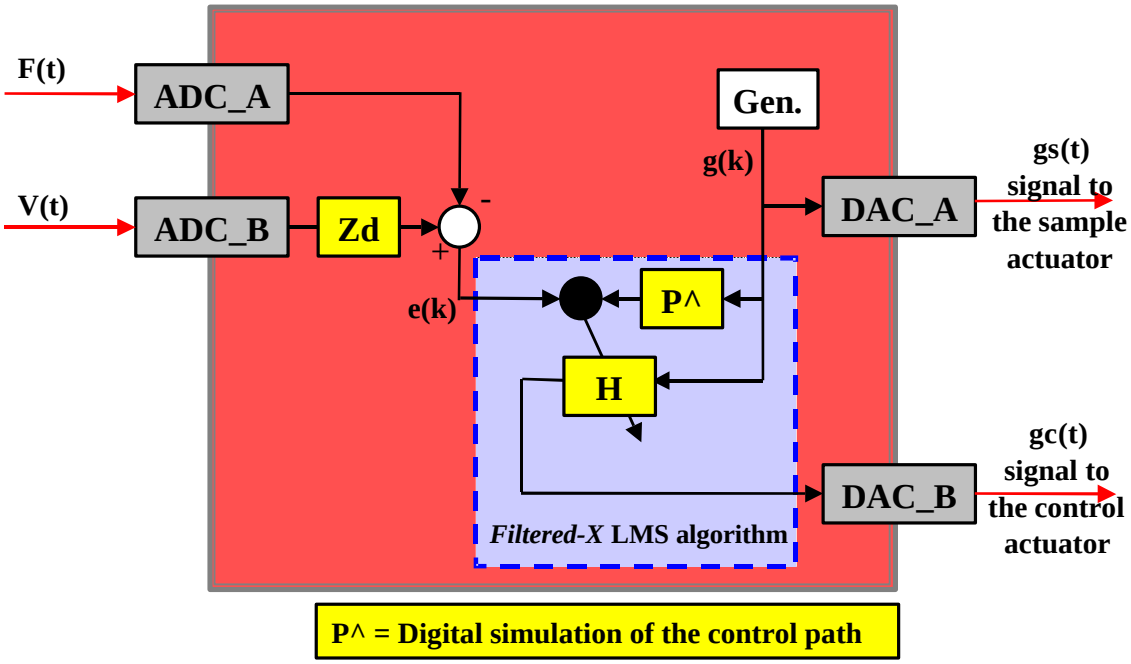


Figure 5.1: Control chart implemented on the DSP board

5.3 Design of Impedance Filters

Given enough authority and computational speed, the controller should be able to reproduce any desired impedance. This desired impedance could be purely real, purely imaginary or complex. However, the controller is limited to simulate high frequencies cases. Indeed, for high frequencies, the required sample rate becomes too high, which implies more computations and therefore, without enough computation power left, the controller cannot get the control algorithm converging properly. Extreme magnitudes of the desired impedance are also limited because of the sensor accuracy. The noise level of the sensors becomes too important when the force or velocity signals become too low (this is the case for impedance of very low or very high magnitudes, respectively). However the value of these extreme levels is directly linked to the magnitude of the controller gains, which then, become also a part of the limitation factors.

As part of the control process, it is necessary to model the complex desired impedance using a digital filter (see Figure 5.1). There are many software design packages available for designing FIR filters (Matlab, Labview…). All of these programs only allow the magnitude but not the phase of the filter to be specified. However, in our case phase matching is as important as magnitude matching since different phases imply different properties of the impedance (e.g.: the impedance of a damping system is purely real whereas the impedance of an inertial system is purely imaginary). The next section presents a method for specifying both the phase and the magnitude of a FIR filter using Fourier Transform analysis.

5.3.1 Fourier Transform Analysis

Figure 5.2 shows a simple block diagram of a FIR filter.

Figure 5.2: FIR filter

Let us call x the input signal to our FIR filter. As we will only be interested in harmonic control, the input, x, may be chosen, without loss of generality, to be a cosine function. Therefore at the n^{th} sample of the signal we have:

$$x(n) = \cos(\omega_0 nT), \tag{5.2}$$

where ω_0 is the frequency in rad/s.

Since the input signal, x, is harmonic, the magnitude and phase of the output signal, y, can be perfectly defined with only two coefficients:

$$y(n) = c_1.x(n-1) + c_2.x(n-2), \tag{5.3}$$

where c_1 and c_2 are the two coefficients of the FIR filter.

Substituting equation (5.2) into equation (5.3) we get:

$$y(n) = c_1.\cos(\omega_0.(n-1).T) + c_2.\cos(\omega_0.(n-2).T) \tag{5.4}$$

Using the trigonometric identity: cos(a-b) = cos(a)cos(b) + sin(a)sin(b), equation (5.4) becomes:

$$\begin{aligned} y(n) = c_1\,[&\cos(\omega_0 nT)\cos(\omega_0 T) + \sin(\omega_0 nT)\sin(\omega_0 T)] + \\ c_2\,[&\cos(\omega_0 nT)\cos(2\omega_0 T) + \sin(\omega_0 nT)\sin(2\omega_0 T)]. \end{aligned} \tag{5.5}$$

We can rearrange equation (5.5) as:

$$\begin{aligned} y(n) = \ &\cos(\omega_0 nT)\,[c_1.\cos(\omega_0 T) + c_2.\cos(2\omega_0 T)] + \\ &\sin(\omega_0 nT)\,[c_1.\sin(\omega_0 T) + c_2.\sin(2\omega_0 T)]. \end{aligned} \tag{5.6}$$

By analogy with complex number, where the part of the sum with the cosine term can be referred as the real part and the part with the sine term can be referred as the imaginary part, if the expressions into brackets are simply considered as coefficients, any change of these coefficients would directly affect the magnitude and the phase of the output y(n). The output can then be monitored only with the two coefficients, c_1 and c_2.

Then in the frequency domain equation (5.3) becomes:

$$Y(\omega) = Z(\omega)X(\omega) \tag{5.7}$$

Where X, Y, and Z are the Fourier Transform of x, y, and the impedance FIR filter Z, respectively.

Then, from equation (5.6), identifying real and imaginary parts of Z at the design frequency ω_0 we have the following expression:

$$Z(w_0)=\begin{bmatrix} Z_{real}(w_0) \\ Z_{imag}(w_0) \end{bmatrix}=\begin{bmatrix} \cos(w_0T) & \cos(2w_0T) \\ \sin(w_0T) & \sin(2w_0T) \end{bmatrix}*\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \tag{5.8}$$

and then as long as the matrix $A=\begin{bmatrix} \cos(w_0T) & \cos(2w_0T) \\ \sin(w_0T) & \sin(2w_0T) \end{bmatrix}$ is invertible, we can easily calculate the coefficients of the FIR filter to produce any desired phase and amplitude (i.e.: real and imaginary components) at ω_0:

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix}=\begin{bmatrix} \cos(w_0T) & \cos(2w_0T) \\ \sin(w_0T) & \sin(2w_0T) \end{bmatrix}^{-1}*\begin{bmatrix} Z_{real} \\ Z_{imag} \end{bmatrix} \tag{5.9}$$

Using this filter design method, the desired impedance filter Z_d can be created and used to model any complex impedance condition at ω_0. In the case where the matrix A is poorly conditioned (very small determinant) the coefficients c_1 and c_2 can become very large and potentially cause frequencies away from the design frequency to be amplified. To avoid this problem, a solution is to add an extra coefficient is determined as follows:

$$\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}=F^{+}*\begin{bmatrix} Z_{real} \\ Z_{imag} \end{bmatrix} \tag{5.10}$$

where

$$F = \begin{bmatrix} \cos(w_0 T) & \cos(2w_0 T) & \cos(3w_0 T) \\ \sin(w_0 T) & \sin(2w_0 T) & \sin(3w_0 T) \end{bmatrix} \tag{5.11}$$

and F^+ is the pseudoinverse defined by the Moore-Penrose conditions:

$$FF^+F=F,$$
$$F^+FF^+=F^+,$$
$$(FF^+)^T=FF^+,$$
$$(F^+F)^T=F^+F.$$

Computation of the pseudoinverse matrix, F^+, requires the use of the singular value decomposition of $F=U\Sigma V^T$, where U and V are orthogonal matrices and Σ is the diagonal matrix of the singulars values, σ_i, of F. Therefore, $F^+=V\Sigma^+U^T$, where the components σ_i^+ of Σ^+ are given by:

$$s_i^+ = \begin{cases} 1/s_i, & if \ \ s_i \neq 0 \\ 0, & if \ \ s_i = 0 \end{cases} \tag{5.12}$$

A more detailed approach is given by Zwillinger [28].

This methodology can easily be extended to accommodate as many extra coefficients as needed. The consequence of these extra coefficients is that they allow the desired impedance filter to behave like a band-pass filter for the frequencies other than ω_0 (this will be shown later in this Chapter).

The advantage of this method is that the magnitude and phase can be set exactly for any FIR filter. In addition, this method can be used for both single frequency and for multiple frequencies. Another important point is that we may use as many coefficients as we want for the filter. The adaptive feedforward controller software has been written such that we have to set the size of all of the different filters, created and used during the control task,

to be the same length. Indeed, it is often necessary, for control purposes, to vary the size of the control filter, and therefore we also have to be able to design impedance filters of the same size. This method gives the ability to adapt the size of our desired impedance filters without altering the magnitude and phase for the desired impedance at the designed frequency.

5.3.2 Implementation in the Labview Interface

Once the theory had been developed, it was possible to implement this filter design into the controller software. The idea was to give the user the ability to design their desired impedance filter taking into account the other settings for the control system that may have a direct effect on the desired impedance filter itself (frequency, sample rate, and number of coefficients). Therefore, the filter design program was written using Labview software so that it could be easily incorporated into the pre-existing interface software for the controller. Rick Wright of VAL had written the general code for filter design and some modifications have been added for the purpose of this research. These modifications also allowed implementing the filter design code into the controller program.

Figure 5.3 shows the main window of the filter design program. To design a particular filter, the user simply enters the desired frequency, the sample rate, the number of coefficients and the magnitude and phase at the given frequency. Then, the program displays a plot of the magnitude and phase of the frequency response of the filter, its norm, and an array of its coefficients. It is also possible to save these coefficients into a file. This file may be later downloaded to the DSP board before starting the control task. The example shown on this graph (Figure 5.3) is a perfect illustration of the importance to have the ability to take as many coefficients as necessary. Indeed, the filter shown here is designed to have a magnitude of 50N/m/s at 500Hz. By taking 15 coefficients, a good reduction of the magnitude away from the design frequency is ensured. Since this

desired impedance filter behaves as a band-pass filter around the design frequency, the controller will be able to focus on this frequency without being disturbed by other high level frequencies. This is important to avoid that the controller exerts all its effort on those frequency levels that are different than the frequency of interest where the control is required.

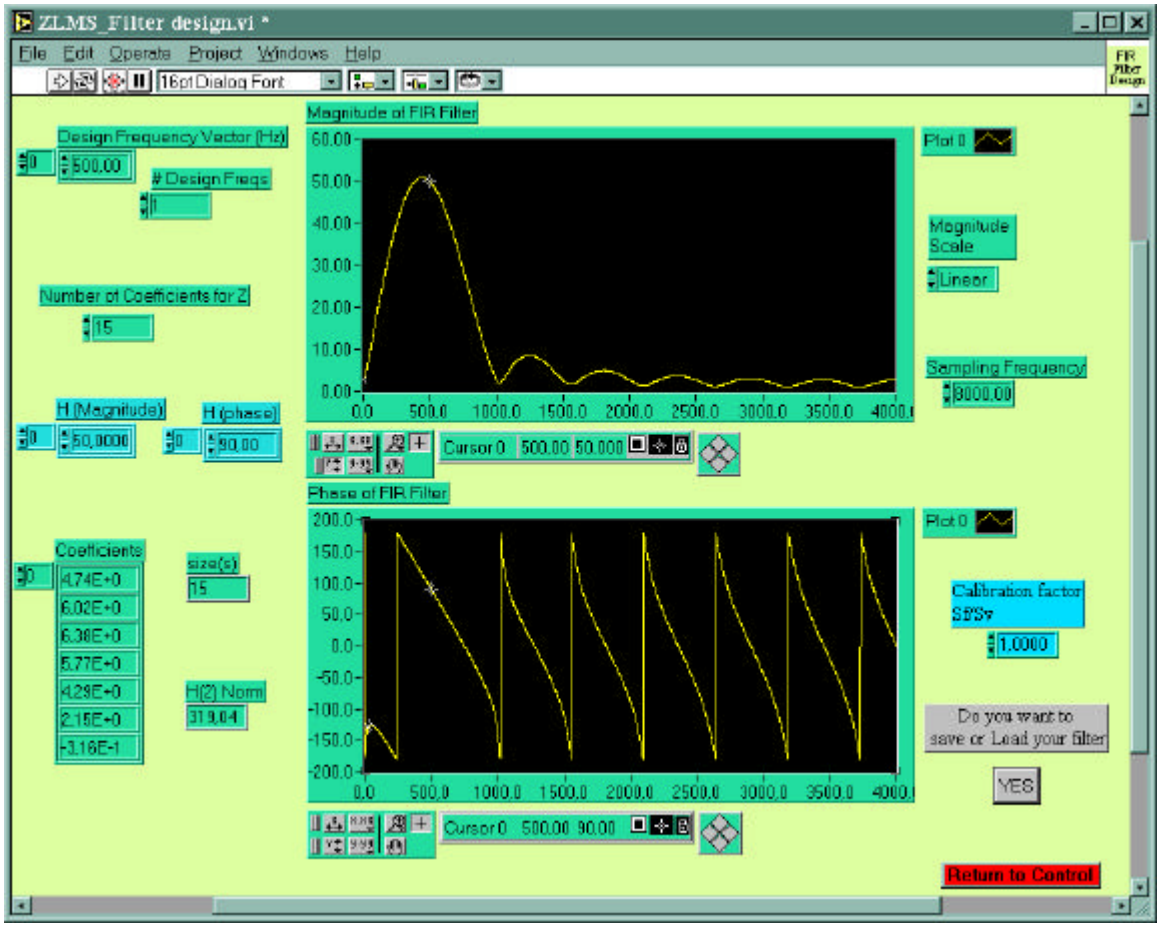


Figure 5.3: Labview command panel for designing desired impedance filter

5.4 Test set-up

From the results of the previous set-up used for the simulation (Cf: Figure 4.4), some modifications have been made. First, to solve the problem previously raised concerning the phase synchronization between the accelerometer and the force gauge, a new sensor has been purchased. This new sensor is an impedance head type 8001 from Bruel and Kjaer (Cf: Figure 5.4). It allows the force and velocity to be both measured with a single device with a maximum phase delay of 1°.

Figure 5.4: Impedance head type 8001 from Bruel & Kjaer

The actuators, used for the results in this section (and in Chapter 6) are the 1_3 piezoelectric tube arrays provided by MSI (Figure 5.5). Some modifications had to be made on the actuators. Aluminum plates had to be glued to both sides of the actuators. Threads were drilled into these aluminum plates such that the different devices could be

screwed together in series. However, these 1_3 tube array actuators require a minimum static compressive load. The actuators are composed of piezoceramic tubes fixed on a soft urethane matrix (Cf: Figure 5.5). The compressive load not only ensures that all of the stiff piezoceramic tubes are in contact with the actuator faceplates but also prevents the tubes from cracking in case of to much stretching du to large displacements of the actuator. This allows the maximum output of the actuator to be achieved. Figure 5.6 shows how the pre-load has been determined. The required value for the pre-load that has to be applied on the actuator is located in the region where the plot of the force versus pre-load starts to flatten. In that case, the required pre- load has been chosen around 180psi. A higher value could have dangerous for the sensor devices in series with the actuators on the rig. Compared to the PCB's actuators used for the simulation, these tube array actuator have a much larger force output and a much greater stiffness. As a result, the rig presented in Chapter 4 was no longer robust enough and the rig base could not be considered rigid. The approximation of infinite impedance of the base was no longer valid. Therefore, we had to build a much more robust rig. We had welded three pieces of plain steel that are approximately 10 inches long and whose cross section area is 9 inches^2. The total weight of this test structure is around 100lbs. This rig is shown in Figure 5.7.

The static load cell seen above the control actuator and whose display screen is on the bottom right of the picture, is used to monitor the compressive load. This load could be altered by screwing or unscrewing the bolts on top of the middle rod. The compressive load was monitored using the load cell to make sure the maximum compressive loads of the devices were not exceeded

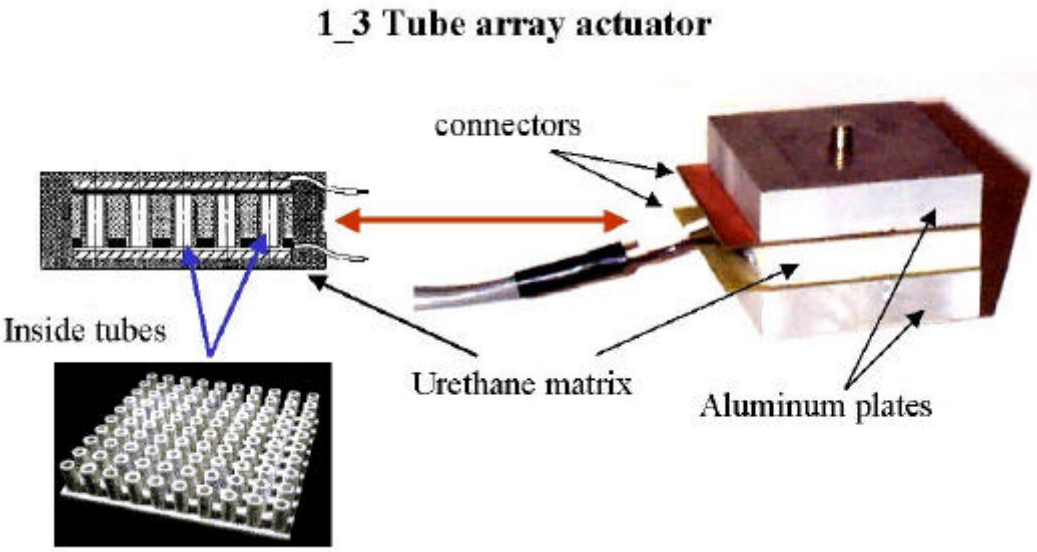
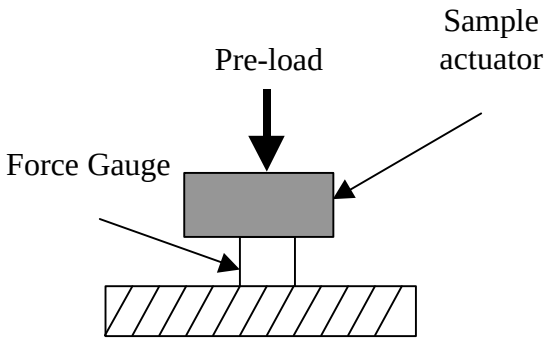


Figure 5.5: 1_3 tube array piezoelectric actuator with aluminum plates glued on both sides and on the left an inside view.



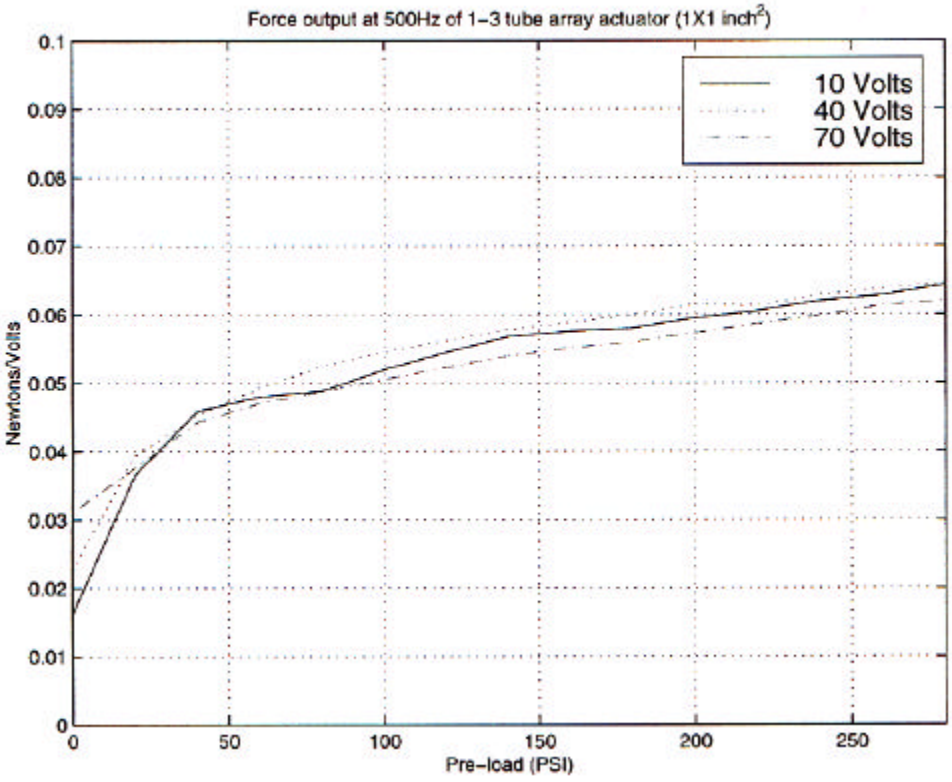


Figure 5.6: Force versus pre-load applied to an actuator under 3 different voltages drive conditions at 500Hz

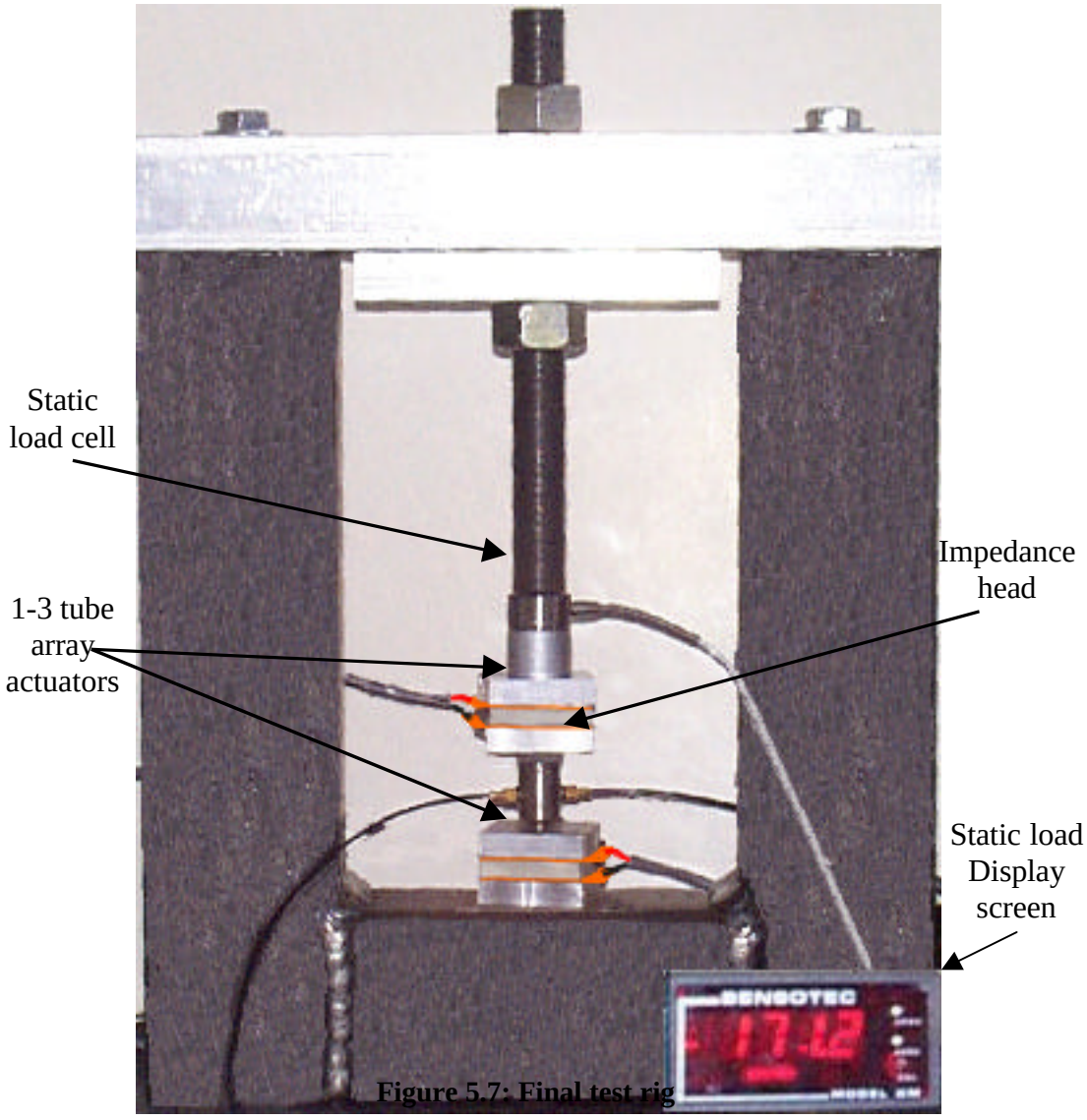


Figure 5.7: Final test rig

The signals measured by the impedance head were amplified using a charge amplifier Type 2635 from Bruel & Kjaer. The charge amplifiers were both set according to each transducer's sensitivity (Force gauge and accelerometer from the impedance head). The signals from the DSP board used to drive the actuators were also amplified using PCB's 790 series power amplifiers. These power amplifiers have an adjustable gain that was used to control the voltage to the control actuator and to the sample actuator.

The four signals coming in and out the DSP-board were all filtered. To do so, 4302 dual Ithaco filters were used as band pass filters. High pass filtering is essentially for signals coming in (force and velocity signals) to avoid low frequency noise problems coming mainly from the accelerometer (vibrations from the table where the set-up was). Low pass filtering is also for incoming signals. Since the sample rate cannot be too high as it has been seen previously (for computing reasons), by cutting high frequencies, low pass filtering is necessary to avoid aliasing problems. For the signals coming out (control and sample signals), the same filters have been used but as reconstruction filters, to smooth the shape of the signals.

A diagram of the set-up is shown Figure 5.8 and Figure 5.9.

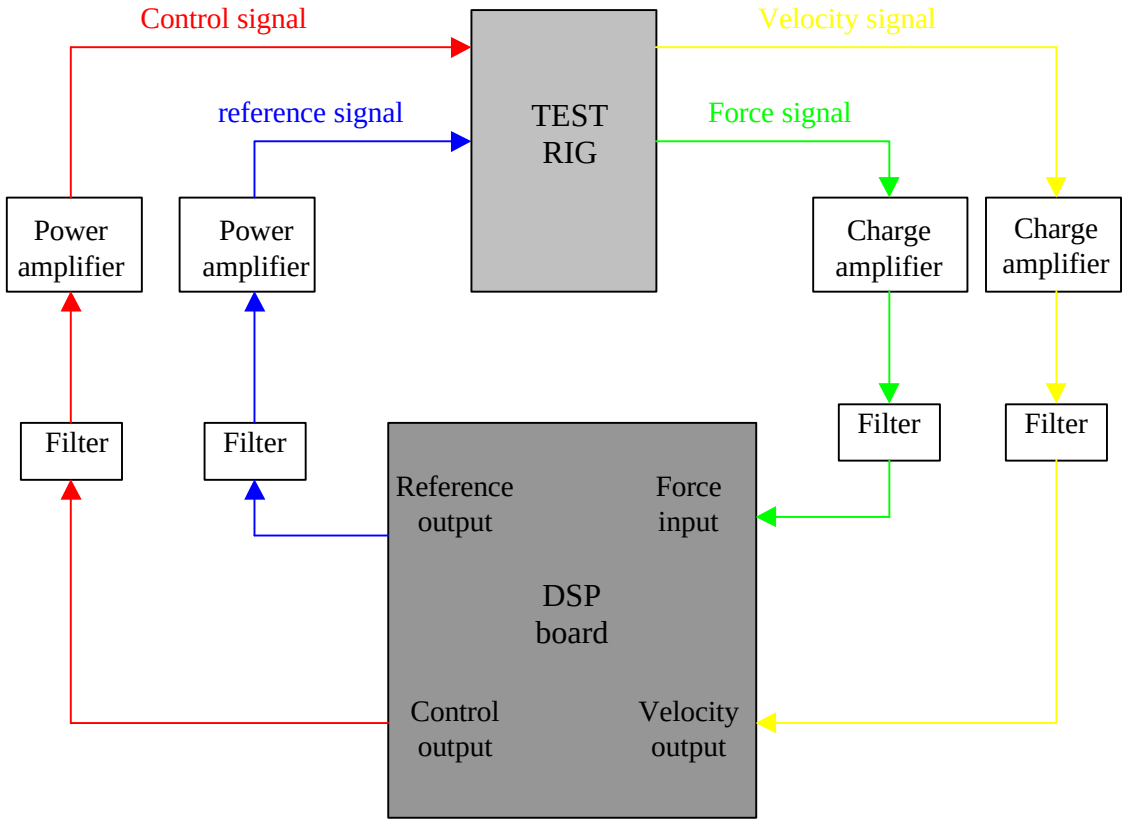


Figure 5.8 : Schematic of the control loop for the entire set up

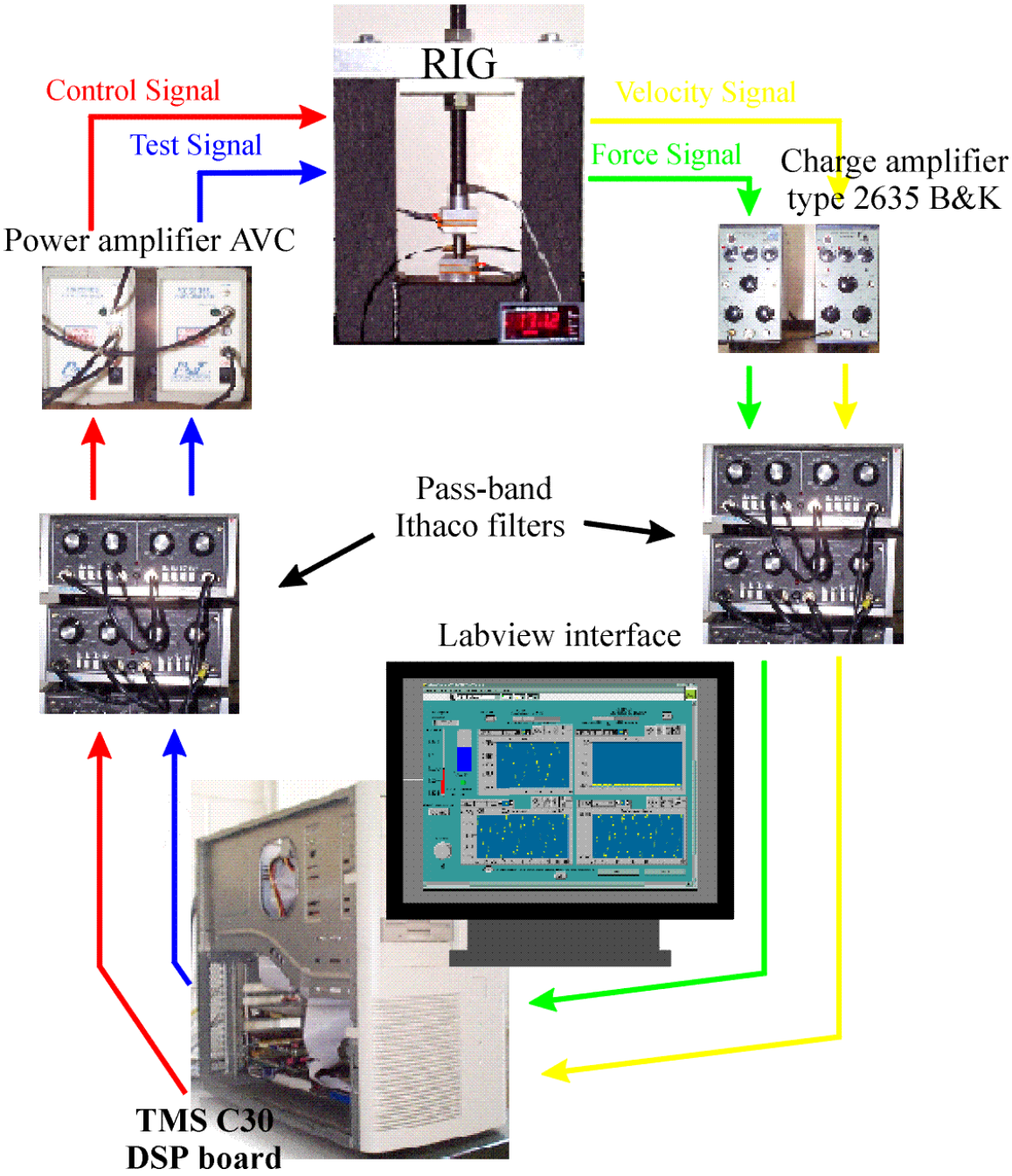


Figure 5.9 : Pictures of the control loop for the entire set up

5.5 Test measurements

Once the all test set-up was ready, the feedforward controller had to be tested. First, it was necessary to ensure that the design of the filters for the desired impedance was correctly implemented in the program and that the convolution of the velocity signal with the filter coefficients was accurate. It was important to verify that the coefficients were not flipped or delayed during this operation. Therefore, it was important to check if the controller was able to reproduce the exact phase of any desired load impedance. Thus, four FIR filters of real, real negative, pure imaginary positive and pure imaginary negative were designed. Each had unit magnitude. Running the controller after having downloaded each of these four filters gave us the following time signals of the force and velocity (Cf: Figure 5.10). For the real positive impedance, Z=F/V=1 (which represents the case of a load acting like a mechanical damper), both signals were perfectly in phase. For the real negative impedance, Z=F/V=-1(which represents the case of a load acting like a mechanical negative damper, adding energy into the actuator), both signals were out of phase. For the pure imaginary positive impedance, Z=F/V=j (which represents the case of a load being a simple mass), the signals were in quadrature (the force signal was in advance by a phase of $\pi/2$). For the pure imaginary negative impedance (which represents the case of a load acting like a spring), the signals were also in quadrature (but in this case the velocity signal was in advance by a phase of $\pi/2$).

The ratio between the force and the velocity was also checked. To do so, we designed a set of impedance filters representing the impedance of different masses, m, for different frequencies. A number of test frequencies were chosen and the appropriate load impedance was calculated (the impedance of the mass m, $Z_m=jm\omega$, at the frequency, ω) and implemented using the control system. For each test, the time signals during control were measured and used to calculate the transfer function between the force and the velocity (F/V) to determine the impedance actually achieved. This impedance was

calculated only at the drive frequency and compares extremely well with the desired impedance (Figure 5.11). This result shows that the load impedance created by a mass can be accurately re-created using the control system.

The worse results are for extreme impedance (Z = 0 or Z $\rightarrow \infty$) where either the force or the velocity signals are driven to very low levels, close to zero and where the sensitivity limits of the sensor devices are reached.

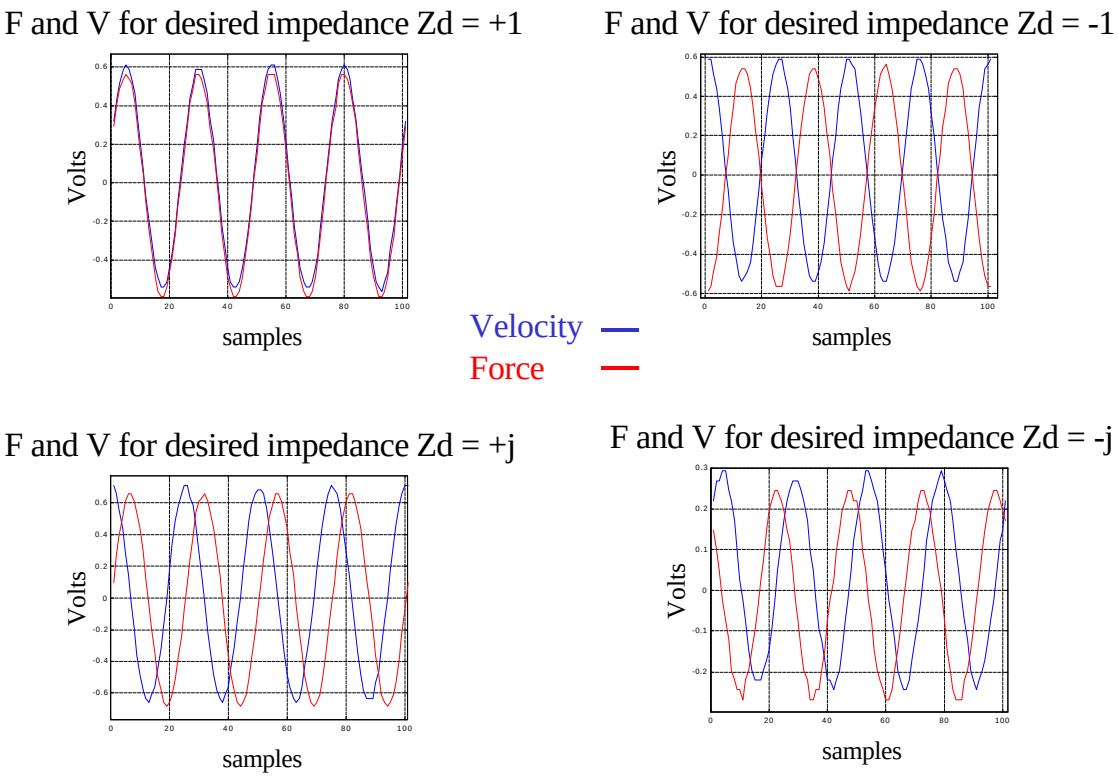


Figure 5.10: Force and velocity signals for four different impedance filters

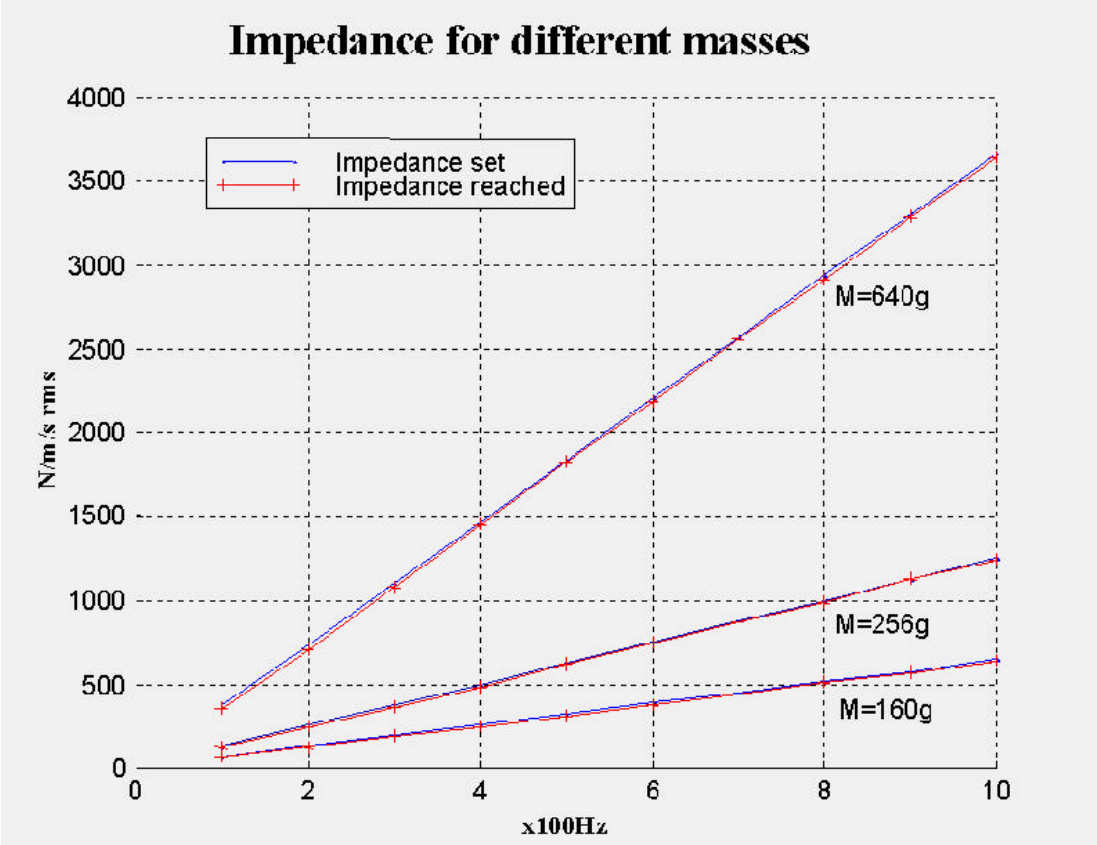


Figure 5.11: Magnitude of impedance set (Z_d) in the controller and of impedance actually reached during the control

5.6 Application of the Technique reproducing the Impedance of an Aluminum Plate

As a demonstration of the technique to a realistic structure, this section shows how the controller can effectively reproduce, over a certain range of frequencies, the impedance load of an aluminum plate excited by a flextensional piezoelectric actuator (model 710MO2 from PCB) used as the tested actuator. The experimental test conditions were as shown in Figure 5.12. The aluminum plate is .035 inch thick, 16 inches long and 11/2 inches large. It was simply supported on two blocks of foam at the extremities. The

blocks of foam had a very low stiffness and were assumed to behave similar to bungee cords, providing close to free-free boundary conditions of the plate. An impedance head (Type 8001 from Bruel and Kjaer) was attached to the bottom of the plate at a given point. The flextensional actuator was fixed to the impedance head on one side and clamped on the other side to a 100lbs block of metal. On top of the plate, another block of foam was glued to slightly damp the system. By adding some damping to the system, this ensured that the plate input impedance measured by the impedance head would not be purely imaginary (the damping effect of the foam increases the real part of the complex impedance of the plate). The reason for this modification (adding damping on the plate) was to ensure that the experiment would provide a wide range of various impedances over a range of frequencies, and ultimately demonstrate that the controller was able to reproduce them.

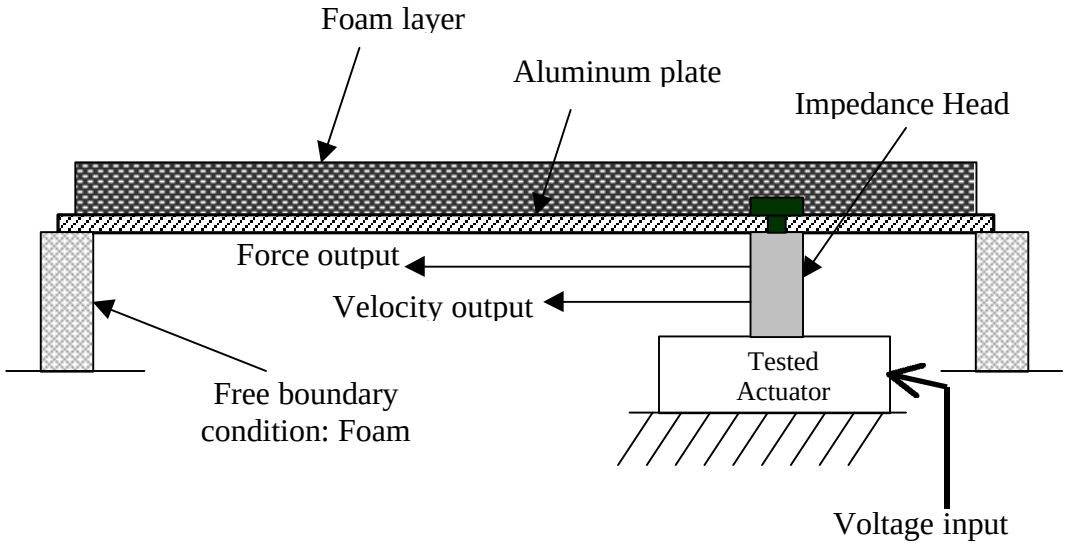


Figure 5.12: Experimental test set-up to measure the impedance of an aluminum plate with free-free boundary conditions

The first set of experimental measurements was taken while the tested actuator was driven with a random input voltage. The time signal of the force output, F, and velocity output, u, measured from the impedance head were used to compute the impedance (Z=F/u) in the frequency domain. The solid lines in Figure 5.13 show the magnitude and

phase versus frequency of the measured input impedance of the aluminum plate excited with the actuator. As it can be seen on these plots, at low frequencies (below 600Hz) measurements are affected with background noise. This was mainly due to the fact that for these corresponding low frequencies, the force and velocity signals outputs normalized to voltage (Cf: Figures 5.14 and 5.15) were very low and because of the limited sensitivity of the impedance head, the background noise was relatively important. In Figure 5.13, resonances of the plate system are not easy to see. They correspond to local minimums of the magnitude of the impedance when the phase approaches close to zero. Indeed, at a resonant frequency, the velocity usually increases, which decreases the magnitude of the impedance (|Z|=|F|/|u|). Furthermore, the mass of the system tends to cancel the effect of its stiffness, and therefore, the damping becomes dominant. Thus, the ratio F/u, which is then proportional to damping, becomes real and its phase almost equals zero. This phenomenon can be observed in Figure 5.13 around 950 Hz.

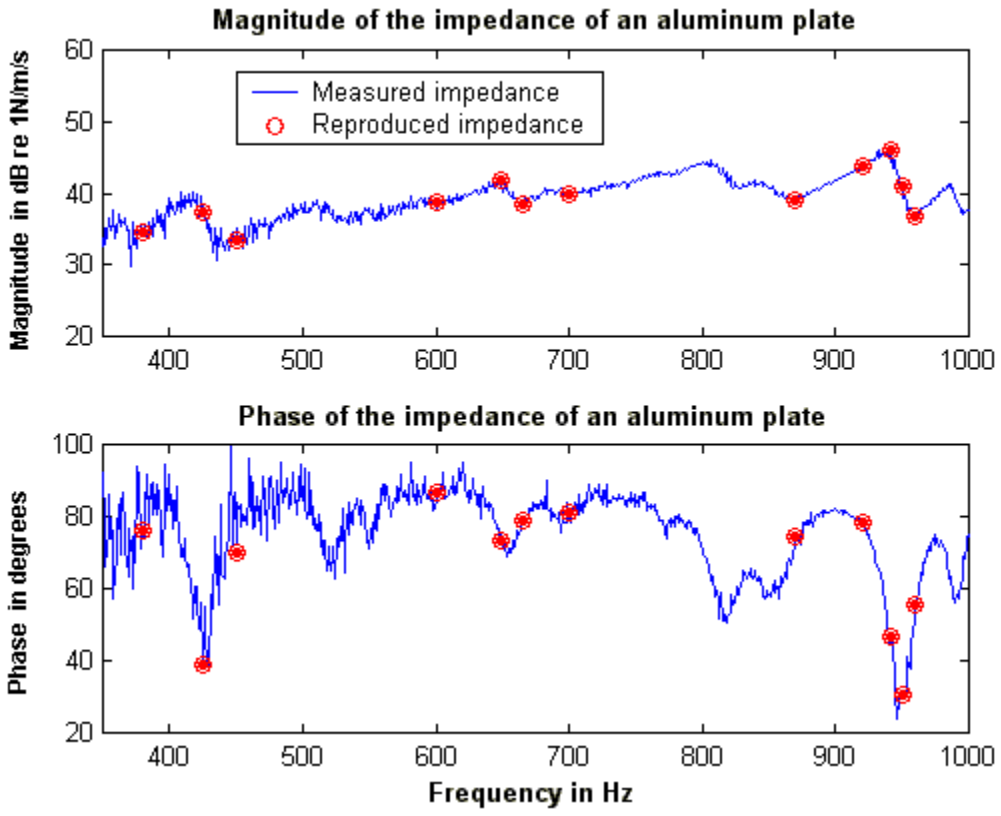


Figure 5.13: Magnitude and phase of the impedance of a plate measured experimentally and reproduced with the controller

To compare these results with the results that the controller can provide, twelve single harmonic frequencies were chosen in the frequency band where the measurements of the impedance of the aluminum plate were taken. The frequencies were selected to correspond to various representative levels of impedance (low levels, high levels, complex impedance, around and at resonance of the plate … etc.).

Once the frequencies were selected, the corresponding magnitudes and phases of the measured impedance were calculated (see Table 5.1) and used to design the desired impedance filters that had to be downloaded on the DSP board to perform the control (Cf: Section 5.3).

	Impedance measured during the experiment		**Impedance measured during the control**	
Frequency	**Magnitude in N/m/s**	**Phase in degrees**	**Magnitude in N/m/s**	**Phase in degrees**
380 Hz	56	74.6	54	76
425 Hz	73.6	40.2	73.8	38.9
450 Hz	47.5	72.2	46.5	70.1
600 Hz	85	85.7	87	86.7
648 Hz	121.6	73.3	122	73.3
665 Hz	83.7	79.5	83.6	79.1
700 Hz	97.6	80.6	98	81
870 Hz	89.9	74	89.8	74.2
920 Hz	155.4	77.4	154.3	78.3
942 Hz	200.4	47.8	201.8	46.4
950 Hz	110.3	30.3	110.6	30.6
960 Hz	69.6	55.3	69.2	55.7

Table 5.1: Magnitude and phase of the impedance of a plate measured experimentally and reproduced with the controller for different frequencies

The test set-up used to perform the active control of impedance was similar to the one shown in Figure 5.8, except that the sample actuator was of course the same flextensional actuator used for the experiment with the aluminum plate. Because the tested actuator cannot handle any pre-load, the control actuator was also a flextensional actuator (same model as the tested actuator), since it was impossible to use a control actuator requiring a pre-load. As all the components of the rig are in series, a pre-load applied for the control actuator would also have affected the tested actuator, which would have then been potentially damaged. The control was then performed for every selected frequency, using the corresponding calculated desired impedance filter, by matching as closely as possible the impedance measured from the impedance head with the desired one (Cf: Figure 3.1).

Results are shown in Figure 5.13 (the levels reached with the controller are represented with the circles) and in Table 5.1. As it can be seen, the results provided by the controller match reasonably well the experimental results. However, the most important thing revealed with this experiment is that not only the impedance ratio reached with the controller matches the experimental case, but both force and velocity signals normalized with voltage, taken individually, are comparable from the experimental case to the controlled case (Cf: Figure 5.14 and 5.15). This means that the behavior of the test actuator is not modified by using the controller arrangement to simulate the experimental system. Therefore, the characterization of an actuator, while under an experimental load impedance condition actively reproduced with the controller, appears to be perfectly legitimate.

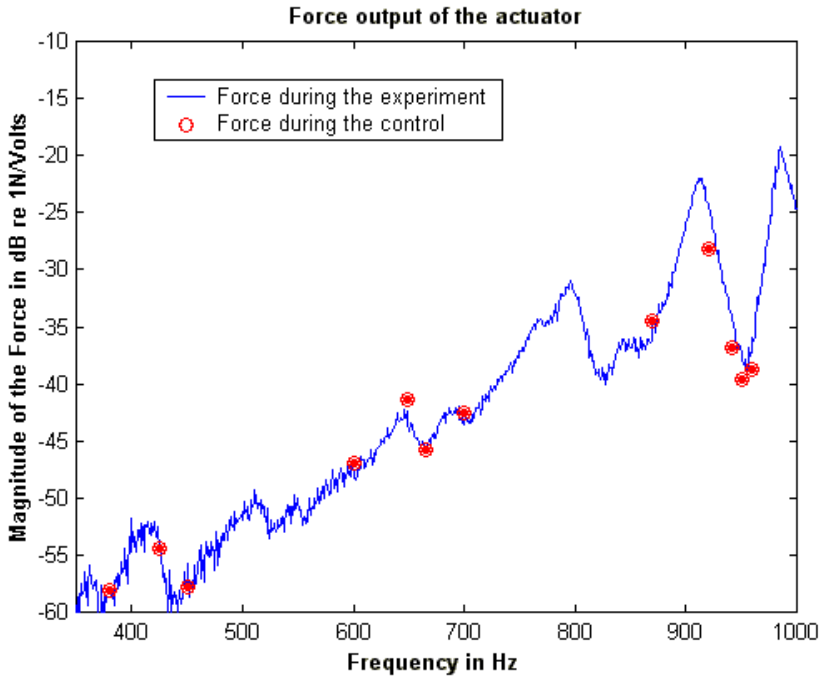


Figure 5.14: Force output of a plate measured experimentally and reproduced with the controller

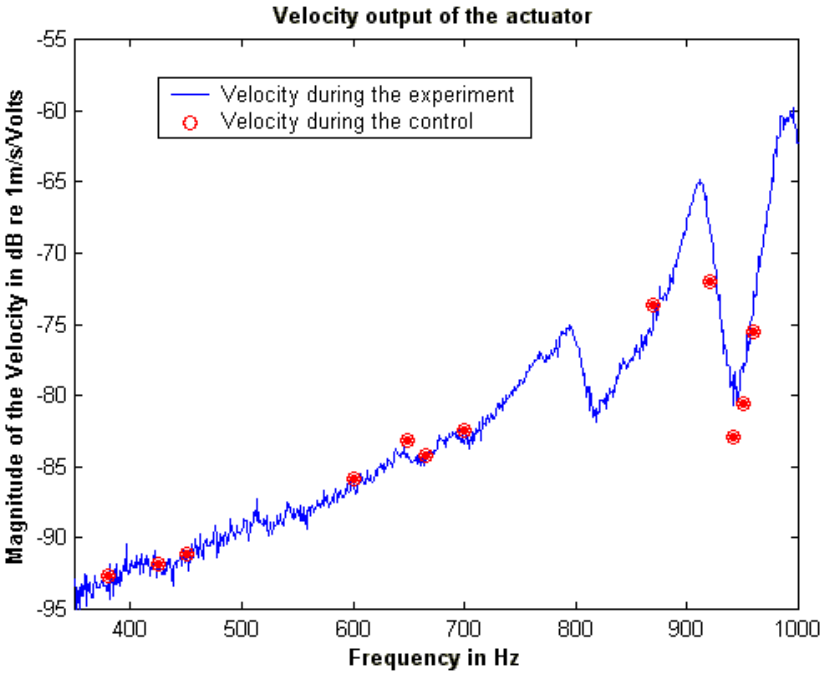


Figure 5.15: Velocity output of a plate measured experimentally and reproduced with the controller

5.7 Limitations

Limitations to the precision of the control are due essentially to the sensitivity of the sensor devices that are used. When one of the input signals (force or velocity) has the same level as the background noise (for extreme impedance values (very high or very low)), magnitude and phases of the signal can no longer be measured accurately (i.e. poor signal to noise ratio). In this case, the error calculated becomes noisy and the control can not be improved.

Another issue that limits performance is the authority required for the control actuator to perform some high impedance levels especially around resonant conditions. In difficult cases it was necessary to increase the gain of the power amplifier to the control signal and decrease the gain to the sample actuator, but sometime even these adjustments do not result in acceptable control. The solution to this problem would be to swap the control actuator for one having greater authority (i.e. more force output).

A further limitation that is due to the test set-up itself, is the condition of infinite impedance at the bottom of the sample actuator. As long as we remain in a range of frequencies away from any resonant frequency of the test rig and as long as we do not want to simulate a very high impedance, the rig performs sufficiently well. In fact, this limitation has some similarity with a real application where the sample actuator would be between two structures of different impedance but none of them necessarily infinite. A solution to this double-sided problem is to set a new control actuator under the sample actuator (both sides of the sample actuator can then be controlled) and to use a two-channel, fully coupled, feedforward controller. In this way, both sides of the sample actuator can be driven to different impedances.

Figure 5.16 shows how this double-sided controller would work. First, a different desired impedance (Z_d1 and Z_d2) is chosen for each side of the sample actuator. Then, the corresponding error signals (Error Signal 1 and Error Signal 2) are calculated (as for the

single-sided controller) and since the system is fully coupled, they are both taken into account to update each of the two digital control filters (H1 and H2) that compute the control signals for the two control actuators. With this double-sided controller, for example, we could simulate an actuator mounted on a metallic plate driving water acoustic field (i.e. representative of a sonar transducer) for instance or any other possible combination. So far, the program for this double-sided controller has been developed but it is still under testing.

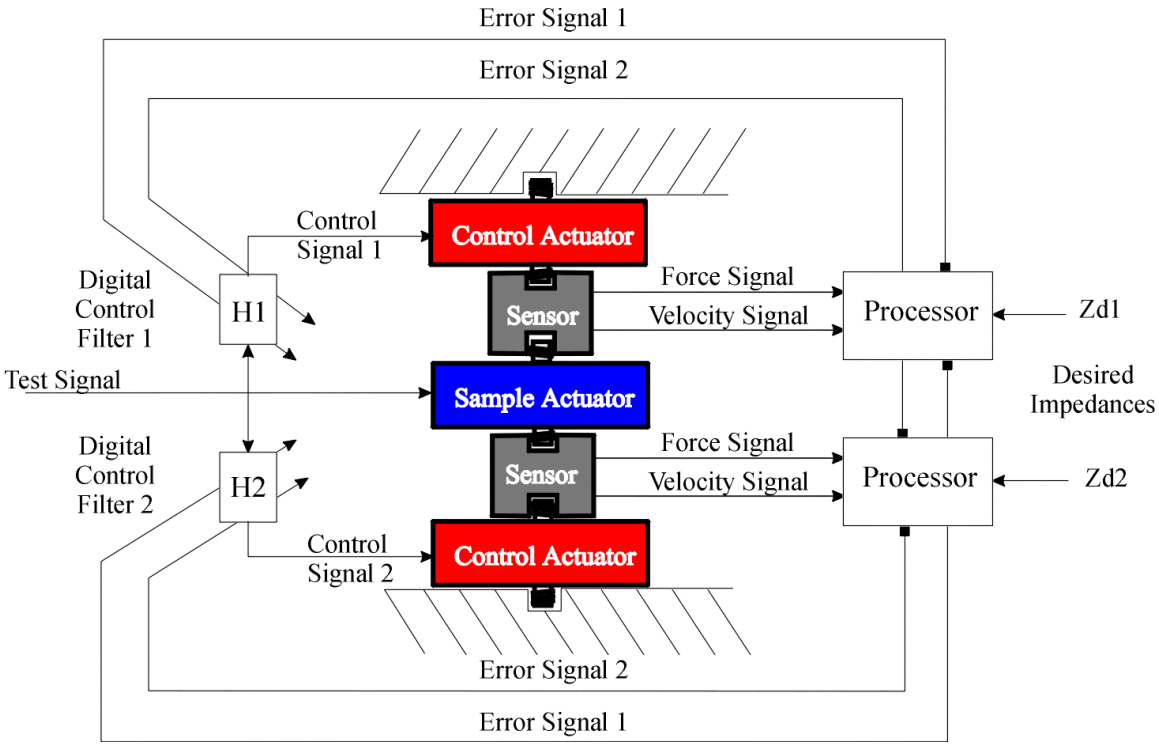


Figure 5.16: Test set-up for the double sided controller

5.8 Discussion

A method for designing the desired impedance filters to specify the correct phase and amplitude at a set of design frequencies has been developed and implemented in the interface program of the controller. After some modifications to the pre-existing controller software necessary to achieve the active control of impedance, the software

was then successfully tested for various desired impedances over a range of frequencies. However, some limitations do exist even for the most recent version of the test set-up that has been built. As long as we understand these limitations, they can always be dealt with.

Another way to approach the problem will be to circumvent these difficulties. As explained in chapter 6, thanks to the controller that has been developed we can create a model of the actuator using data provided by the test set up. After having tested this model with this same test set-up, we can use it to get data for frequencies where the control is not accurate. However, now that the controller works, the process of actuator characterization can be executed. Once the controller has set the desired impedance seen by the sample actuator, data of interest for the characterization (voltage, current, displacement, force, temperature…) can be measured with suitable sensor devices. Plots on figure 5.17 are an example of the kind of measurements that can be performed. These measurements were taken on the sample actuator while the impedance of a mass (m=500g) was simulated using the modified adaptive feedforward controller.

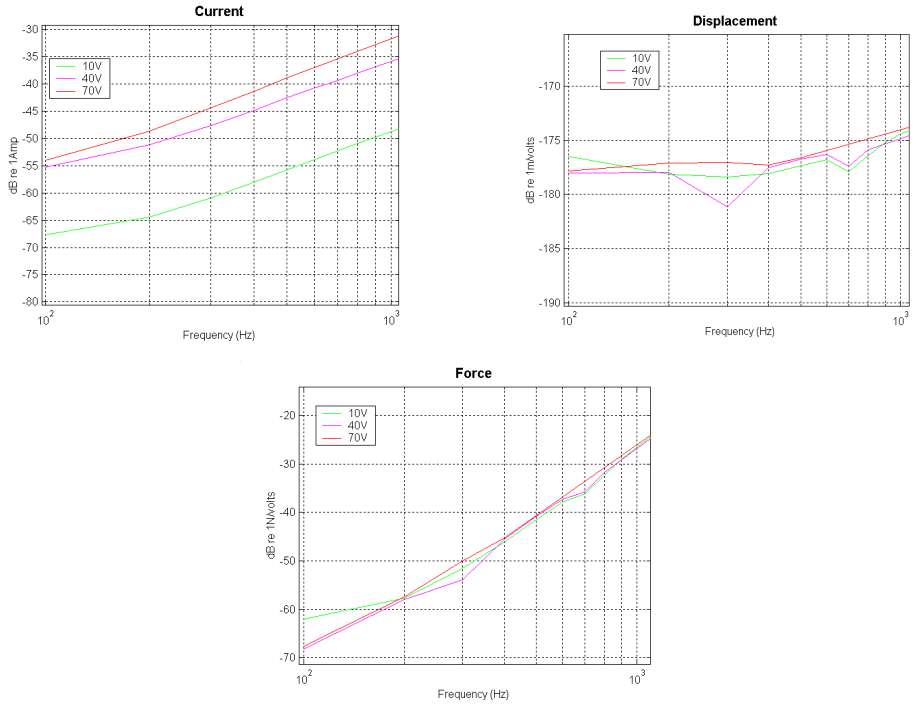


Figure 5.17: Plots of the current, displacement and force of the sample while it is controlled to see the impedance of a 500 g mass