Scalable Advisory System for Structural Health

Final Report

Rustom Jehangir,
Josh Villbrandt,
and Abhinav Chhabra

AME-441a
Submitted: December 8, 2010

Abstract: A Structural Health Monitoring (SHM) System, relying on identification of changes to a structures resonant frequency to recognize potential structural damage and provide alerts, is designed and implemented. A module, using a 3-axis MEMS-based digital accelerometer, capable of measuring and wirelessly transmitting frequency data to a remote computer is made. A test apparatus, consisting of a cantilever-beam and a mechanical device capable of producing up to 20 Hz frequency vibrations in the beam, is constructed to verify the modules readings. The beam is subjected to ambient vibrations and the module monitors its tip acceleration. The remote computer calculates a Fast Fourier Transform to distinguish resonant frequencies on a continuous time basis, and a user interface provides alerts when natural frequency changes considerably.

Introduction

Resonance

Resonance is the tendency of an object to vibrate with higher magnitude at certain frequencies than others. At resonant frequencies, the natural damping forces of a structure are much less which allows the amplitude of a vibration to constructively interfere over successive impulses from the driving force. If ignored, constructive resonant frequencies can lead to damage or destruction of the structure. Even small driving forces, at resonant frequency, can lead to large amplitude vibrations and severe structural damage.

When damping is small, the resonant frequency of an object is the same as that object's natural frequency. The natural frequency of an object is directly dependent on the structural composition of that object. When there is an abrupt change in this natural frequency, one can assume that there has been an abrupt change in the composition of the structure. This project, the Scalable Advisory System for Structural Health (SASSH), attempts to design and implement a Structural Health Monitoring (SHM) system to measure natural frequency changes in an effort to warn users of any possible damage to a structure. This system will also detect dangerous, high amplitude vibrations of any frequency and warn the user.

Objectives

The projects key objectives are as follows:

- To monitor the tip acceleration of a clamped cantilever beam subjected to ambient vibrations
- To transmit the recorded data to a digital storage device wirelessly
- To take the data over a moving time window and find Fourier Transform to distinguish resonant frequencies on a continuous time basis
- To monitor high amplitude vibrations and produce real-time warnings of possible damage
- To track changes in the natural frequencies and provide a damage-alert in case of a considerable change

Concept Product

To meet the depends of the project, a product has been envisioned which consists of multiple wireless accelerometers and host computer to receive and analyze data in real time. The wireless accelerometers, or SASSH modules, will be small be enough to be placed anywhere on a structure. Figure 1 shows how the SASSH modules could be installed on a structure. Once installed, the host computer can determine if the structural integrity of the building is at risk by detecting changes in the natural frequency of the structure at each module. If it is determined that the building is at risk, the system will alert the user.

The device will be designed as a commercial product. A user friendly command interface for the host computer will simplify wireless communication with the modules, and a product manual will provide instructions for advanced settings. Modules will be packaged in a clean, weather-resistant enclosure. The entire system will be contained in a dedicated box for transportation and storage. The final product should be a reliable and economically viable solution for vibration monitoring.

Testing and Verification

The natural frequency of a complex structure, like a building, can be difficult to determine. On the other hand, a simple object, like a metal beam, vibrates in a predictable pattern. The project will rely on a cantilever-beam based test apparatus to physically verify the readings from the SASSH Module. The test model will be comprised of a 36 inch cantilever beam connected by a vertical connecting rod to a DC-motor capable of producing medium-high frequency vibrations of up to 20 Hz (1200 RPM) in the beam. This motor will allow the accelerometers to be at frequencies other than the natural frequency of the beam.

Figure 1: Diagram of the concept use of the SASSH system.

Theory

Vibrations of Buildings

Buildings can vibrate due to a number of reasons including earthquakes, road traffic, and normal daily use. In order to detect these vibrations, which are typically of low magnitude, a high sensitivity accelerometer is necessary. According to the design requirements, the SASSH module was required to measure vibrations with displacements as small as one micron at frequencies between 20-30 Hz. The required sensitivity was calculated as follows. The equation for displacement during a simple oscillation is of the following form.

$$x(t) = Asin(\omega t) \tag{1}$$

where $x(t)$ is the position, A is the amplitude of the vibration, and ω is the frequency in radians/second. The rate of change of displacement, or velocity of the vibrating object is represented by the first derivative as shown below.

$$\frac{dx(t)}{dt} = A\omega cos(\omega t) \tag{2}$$

Similarly, the rate of change of velocity, or the acceleration of the vibrating object, is represented as follows.

$$\frac{d^2x(t)}{dt^2} = -A\omega^2 sin(\omega t) \tag{3}$$

Thus, the magnitude of acceleration of a simple vibration with magnitude, A, and frequency, ω, is:

$$|a(t)| = A\omega^2 \tag{4}$$

Therefore, in order to measure vibrations of amplitude of 1 micron and frequency of 20 Hz, the accelerometer must be sensitive to accelerations of:

$$|a(t)| = (10^{-6})(20 \times 2\pi)^2 = .0158\ m/s^2 \tag{5}$$

The sensitivity of most accelerometers is classified in terms of milligravities, or mg. Nondimensionalizing the above results with gravity provides the following necessary sensitivity:

$$|a(t)| = .0016\ g = 1.6\ mg \tag{6}$$

Module Validation - Cantilever Beam Bending

The theoretical natural frequency of a building is very difficult to calculate accurately so a simple cantilever beam is used for testing and verification. This provided a test apparatus for initial testing of the SASSH module. The simple vibrations produced provide clean data that can be analyzed and verified easily. The following section describes the important aspects of analyzing the beam theoretically.

The natural frequency of a structure, ω_n, is the frequency at which it naturally vibrates once it has been set into motion. The natural frequency of every structure is unique and depends on its material and structural properties and its dimensions. The natural frequency of a single degree of freedom cantilever beam is given as follows,

$$\omega_n = \alpha_n^2 \sqrt{\frac{EI}{mL^4}} \text{ where } \alpha_n = 1.875, 4.694, 7.855 \tag{7}$$

where E is the elastic modulus of the beam, L is the length of the beam, I is the moment of inertia of the beam, and m is the mass of the beam. The beam itself is assumed to be massless but due to the small mass of our beam relative to the weight on the end. It provides an accurate estimate of the beam's theoretical frequency.

The damping ratio, ζ, expresses the amount of damping force in the system. The damping force acts against the vibration causing the magnitude of the vibration to be reduced. Damping is the force that allows vibrations to disappear over time instead of continuing endlessly.

The damping ratio is a characteristic value of the vibrations of a structure and therefore, any change in the damping ratio can be used as an indication that the characteristics of the building have changed. The damping ratio can be found from the Fourier Transfer by measuring the width of the fundamental frequency's peak at the point where the magnitude is equal to $\sqrt{2}$ times the peak's height [4].

The value of the damping ratio is dependent on the properties of the material and is hard to calculate analytically however published experimental results are available for many materials. Professor Udwadia, a vibrations professor at USC, provided a range that the value could fall between, 0.01-0.10.

Realtime Data Processing

SASSH seeks to monitors changes in the amplitude, frequency and damping ratio of vibrations. In order detect changes in these quantities, they must first be measured. Before anything can be measured, however, the time domain data taken from the module accelerometers must first be converted into the frequency domain. A Fourier Transform is used to process this conversion. Many proven implementation of the Fourier Transform exist [3]. The algorithm itself is complicated and will not be discussed here as an in depth knowledge of this algorithm was not needed to complete the product.

Once the data has been converted into the frequency domain, the aforementioned quantities can now me measured. The amplitude of the vibrations is easily measured by taking the average of the frequency domain data. This provides a general measure of the excitement in the system. The frequencies of vibration can be isolated by using a peak finding algorithm. Most peak finding algorithms require a minimum peak height to function. Throughout the development of this system, testing has shown that a reliable method method of calculating a minimum peak height can be found in relation to the average amplitude. For example, peaks that are 200% of the average amplitude are sufficiently above the ambient noise of the system.

The damping ratio is by far the most difficult quantity to measure. Unlike the other two quantities, standard programming functions are not available to calculate the damping ratio of a peak. An algorithm for measuring the damping ratio can be derived from the Q factor. Like the damping ratio, the Q factor is a dimensionless quantity that depends on the damping of a system. The Q factor is defined as

$$Q = \frac{f_0}{\delta f} \tag{8}$$

where f_0 is the frequency of the peak and δf is the width of the peak [5]. The width of the peak is measured where the amplitude of the peak is $A_max\sqrt{2}$ where A_max is the maximum amplitude of the of the entire peak. With the Q factor measure, the damping ratio for a given peak can be found. A conversion is given as

$$\zeta = \frac{1}{2Q} \tag{9}$$

With this, all of the necessary quantities have been measured.

Product Development

Module Requirements

The SASSH module was designed from the ground up to fulfill the specific requirements of the task. It was designed to be simple, easy to use, and fully capable of performing necessary measurements and computations. The device was built to the following specifications:

- Utilizes a 3-axis MEMS-based accelerometer sensitive to at least 1 mg with a measurement bandwidth of at least 100 Hz
- Wireless chip capable of transmitting through the walls of the building to a host computer. The wireless chip also functions in parallel with others so that large sensor networks can be built. Up to 10 modules can be used simultaneously
- Operates on battery or wall-transformer power
- Contains enough memory to store data while the host computer is talking to other modules
- Contained in a enclosure that resists dangers of everyday use and light weather

Module Development Process

Design of the SASSH module device consumed the first half of the semester. The device design went through several iterations before it was actually built. As this was the first microcontroller board designed by the team, there was a learning curve and subsequent designs were dramatic improvements over the previous design.

The board was initially designed with an analog accelerometer similar to some that the team members had used in past projects. The PCB board design was completed in EagleCAD, a free software package available online. The board was sized to fit into a chosen electronics enclosure including mounting holes that allow direct bolting to the enclosure. The first design was shown to Ph.D student Randolph Voorhies who has significant board design experience. Upon his recommendation, the board was completely redesigned to fix problems with the layout of the board. The board was revised several more times afterwards until an acceptable design was produced.

After discussions with Professor Udwadia, the team realized that the sensitivity of the device needed to be much higher. Additionally, even with a sensitive analog accelerometer, much of the data would be lost due to electromagnetic interference and the low resolution of the microcontroller's analog-to-digital converter (10-bit). Many other accelerometer options were researched until one was found with very high sensitivity, a noise-free digital connection, and high resolution measurements (16 bit). The PCB board design was changed to include this accelerometer.

The PCB board design was sent to and manufactured by Advanced Circuits, Inc. Once the bare PCB boards were received, the team hand-soldered the components onto the boards using a hot air soldering station. Unfortunately, the first board manufactured had a design error where two connections were flipped, rendering the accelerometer useless. Everything else on the board functioned correctly. The board was immediately redesigned (including some additional improvements) and was reordered. It was reassembled and tested and everything worked correctly. An additional module was assembled so that multiple modules could be tested simultaneously.

The microcontroller's software had to be written to read the accelerometer at a precise time interval and send the data through the wireless connection. The accelerometer communicates through I^2C, a high

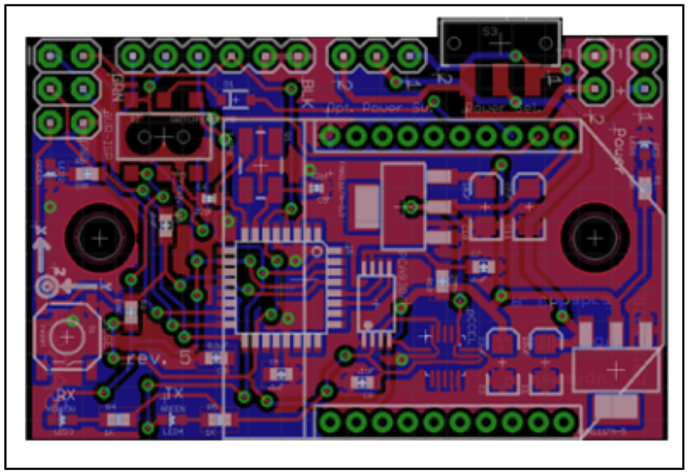


Figure 2: CAD design of the SASSH module.

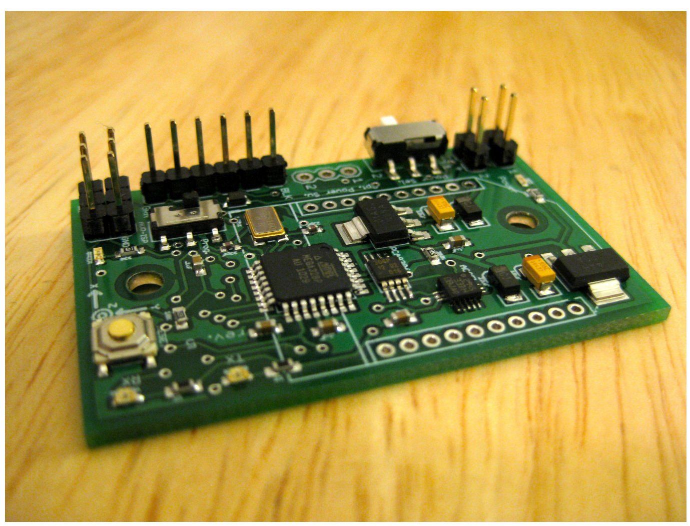


Figure 3: Assembled and functional module (without XBee wireless modem attached or plastic enclosure.)

speed serial communication protocol for digital communication. An Arduino software library was written to interface with the accelerometer and was used in future software development [1].

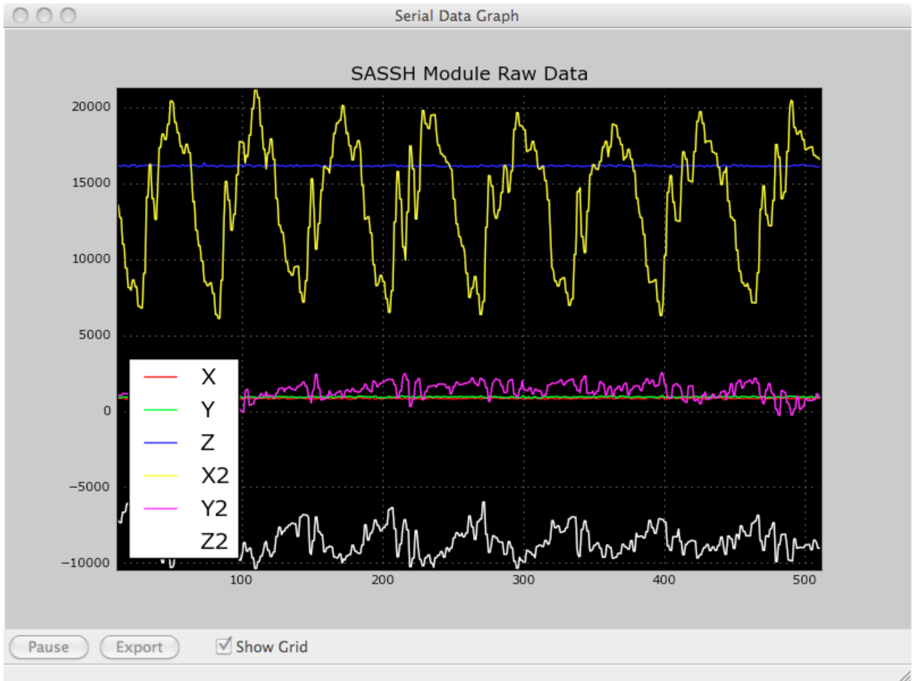


Figure 4: Python application written to test raw data output of SASSH modules.

Initial data gathering on the host computer was done in a custom program written in the Python programming language. The simple libraries available in the language allowed rapid development of a test interface to show that the module was working and transmitting data correctly. All later host computer software was written in LabVIEW.

Module Final Design

Early in the development process, a significant effort went into choosing hardware components to ensure that the modules could perform to all of the necessary design specifications. The following is a summary of the important components and the argument for using them on the SASSH module.

Accelerometer. The LIS331 accelerometer was chosen as the measurement device for the SASSH module. This accelerometer was chosen for its high publish sensitivity (1 mg) and digital interface. The sensitivity is sufficient to measure the small vibrations experienced in buildings. The digital interface is unique as most MEMS accelerometers have an analog output. Analog outputs are susceptible to electromagnetic noise before being read by the analog-to-digital converter. A digital accelerometer avoid this problem.

Microcontroller. The SASSH module also contains an Atmel ATmega328 microcontroller that reads the accelerometer measurement at the desired sampling rate and forwards the data to the host computer when requested. The most important aspect of this particular microcontroller is its support of the Arduino programming IDE. Arduino is an open source attempt to simplify microcontroller programming to allow rapid development without extensive knowledge of the electronics. It includes libraries that make it simple to perform many of the necessary actions such as serial communication and digital communication with the accelerometer. Additionally, although the FFT was calculated on the host computer in this project, the microcontroller is powerful enough to allow onboard FFT calculations in the future.

Wireless Module. An XBee wireless module was used to provide a wireless link to the host computer. The module was chosen due to its simple use and the team's experience with it. It provides out-of-the-box serial line replacement with support for multiple devices sending data to one host.

Figure 5: A microcontroller, MEMS-based accelerometer, and XBee wireless modem.

Enclosure and External Connections. The circuit board is mounted in a small plastic enclosure to protect it. The enclosure is approximately 3”x3”x2” and an external antennae for the wireless signal, an external 7-12V DC power connection, an on/off switch, and an on/off LED indicator. The external connections allow the SASSH module to be a fully enclosed product with no need to access the internal components.

Communication Protocol Requirements

In order for SASSH to monitor the module accelerometers, the data must be somehow transmitted from each SASSH module to the host computer. The XBees allow the modules to send data through standard serial commands. However, the data transfers quickly become difficult with multiple modules on the same serial line. The module software (where the communication protocol is defined) must meet the following specifications:

- Allows communicate of multiple SASSH modules simultaneously
- Reliably transfer the x, y and z acceleration components for each module
- Transfer fast enough support the above data being sampled at 100 Hz from each module

Communication Protocol Development

The first implement of the of the low level SASSH module architecture simply dumped accelerometer data on the serial line. That is to say that the data did not identify which module it was coming from and did not wait until the host computer was ready to receive it. This was useful for initial testing of the SASSH module hardware. With this system, our first data sets where collected and stored for MATLAB analysis.

The next iteration of the communication protocol focused on allowing multiple modules to communicate at the same time. Because of the nature of serial lines, multiple pieces of data cannot actually be sent simultaneously. To overcome this difficulty, the SASSH command interface was programmed to request data from each module in sequence at a rate of 100 Hz (the minimum sample frequency required to acquire the data needed). This method also had the benefit of not needing to store any data on each module, as the module would just send back the most recent data when it received a request.

This solution, however, proved to be problematic. The relatively fast update rate (of 100 Hz) did not provide enough time for the serial communication to be initialized and to send the accelerometer data for multiple modules. In other words, more than 10ms had elapsed by the time the data was received. This means that data could never actually be measured as high as 100 Hz. The debug fields of the command interface, shown in Figure 6, where used to diagnose this problem. Once this problem was understood, the protocol was rewritten to save multiple samples and to send them in bulk when requested. This method proved to allow enough time to transfer the data through serial between request periods.

However, there was an additional problem with this new method. The time to transfer multiple samples was still greater than 10ms. These means that although data was sampling effectively at 100Hz, the modules were not sampling at all during a data transfer. The maximum physical transfer rate of our system is set

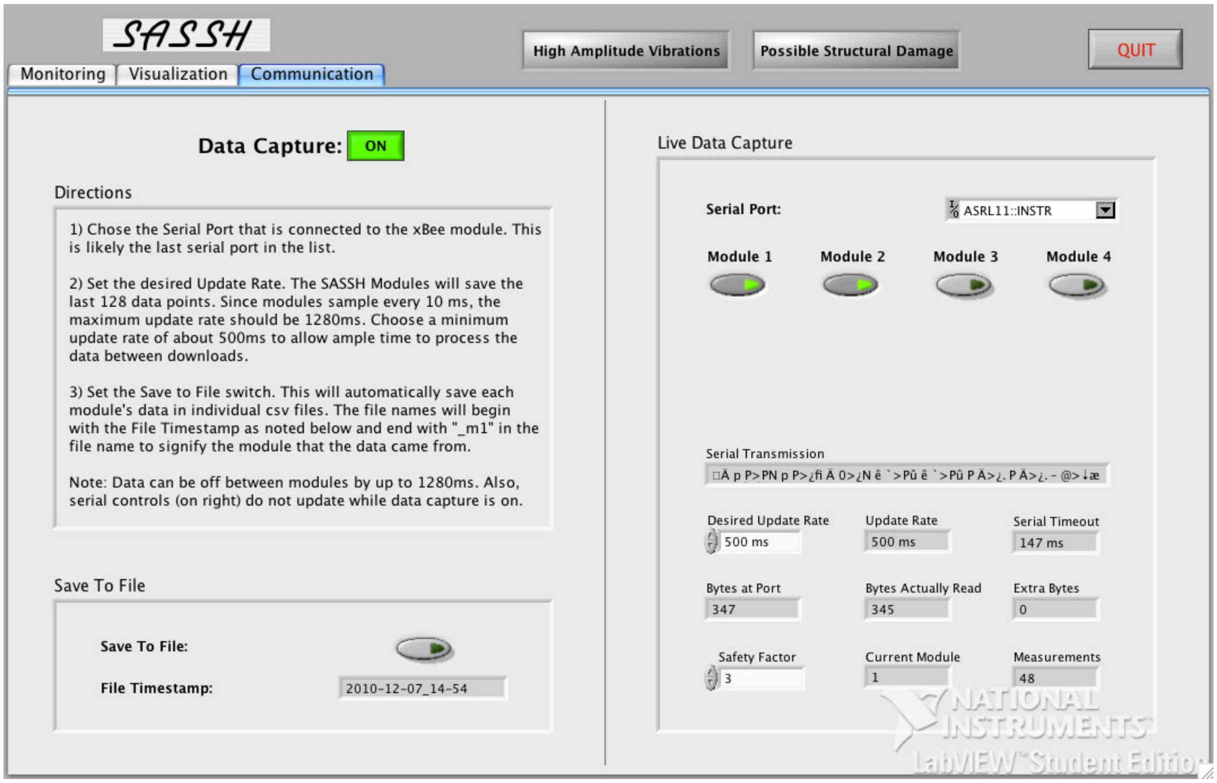


Figure 6: The Communication tab of SASSH. This tab controls which modules are active, whether or not to save data to files, and whether or not the entire system is running. Additional fields at the bottom right provide debugging information.

to 7200 bytes per second and each sample is seven bytes. This means that theoretical shortest time to transfer one sample is 1 ms. If the modules were to transfer data once every second (or 100 samples at once), for example, this would take at least 100 ms just to transfer the data. This means that about 10 samples would not actually be taken, so really only about 90 samples were being collected every second.

To overcome this problem, the software was rewritten to simultaneously sample and send data. Since the microcontroller doesn't support multithreading, the process of sampling and sending data had to be manually threaded. Essentially, the microcontroller samples if needed after sending an individual sample (7 bytes) as opposed to the entire sample set (352 for 50 samples) as before. But for this process to occur, the module will have to read and write from the stored sampling data at the same time. To ensure that these two processes do not overlap, dual two-dimensional byte arrays were implemented. This allows one byte array to always be the sampling array and one byte array to always be the sending array. Upon each new sending request, these byte arrays switch. With this fix, SASSH modules were finally able to sample and send accelerometer data in an adequate manor.

Communication Protocol Final Design

The final communication protocol works on as a request/send behavior. The SASSH command interface requests data sequentially from all active modules by sending the ASCII value of the module number. When a SASSH module receives its personal number, it proceeds to dump any data that hasn't already been sent. Only the last 128 values are stored however, so data must be collected at least once every 1.28 seconds.

The format for the data being sent back is first the source address (byte), then the sample count (byte) and then the individual samples. Each sample has the format x axis data (byte pair), y axis data (byte pair), z axis data (byte pair) and a checksum (byte). This checksum is produced by taking the bitwise XOR of each byte in the sample and is useful for ensuring that data was transferred successfully. Samples are send from oldest to newest sequentially.

While every works well at the moment, the system has only been tested with two modules. By keeping the same programming structures on both the modules and the command interface and perfecting the various time intervals, it is estimated the only ten modules could be used simultaneously with SASSH. To increase the number by a factor of ten, the frequency analysis of accelerometer data would have to be done on board. This would eliminate the need to send back the entire data set, significantly reducing the total amount of data transferred for each module.

Command Interface Protocol Requirements

The command interface is the heart of SASSH. This computer program is responsible for all communication between modules and processes data is that is essential for monitoring the health of the structure. The key requirements for this software are:

- Communicates with a variable number SASSH modules through serial
- Has the ability to save the raw accelerometer data to a file for future analysis
- Can calculate the FFT for each accelerometer axis on each SASSH module in real time
- Can locate the peaks and calculate the damping ratios for each FFT in real time
- Is capable of distinguishing high amplitude vibrations and issues a warning
- Is capable of distinguishing changes in natural frequencies and damping ratios and issues a warning

Command Interface Protocol Development

To meet all of the requirements of the command interface, LabVIEW was chosen as the primary programming language. LabVIEW provides easy methods for serial communication and converting the accelerometer data to frequency domain. It also provides methods for easily displaying data. Figure 7 shows an example of LabVIEW graphical output capabilities. Finally, LabVIEW also contains functions to easily write data to a file.

The development the SASSH command interface was the most time consuming part of the project. While LabVIEW's graphical programming language makes some tasks easy, just as routing data from a source to a graph, it makes many tasks that at trivial in standard text-based languages rather difficult. An example of this is modifying persistent data through multiple iterations of a loop. While this is easy in a text-based language where one instantiates a variable and changes it as needed, LabVIEW requires the user to create confusing feedback loops or instantiate a variable to be global, if it this behavior is not desired. With the said, LabVIEW did come through in the end, and was able to produce a sufficient command interface for SASSH.

Command Interface Protocol Final Design

The final version of the command interface successfully implements all of the stated requirements. In its current version, the command interface is able to communicate with up to four SASSH modules simultaneously. Graphs and monitoring fields are provided for the first two. (Only two complete modules were made due to budget constraints, so this is sufficient.) The various functionalities are broken up into three tabs: Monitoring, Visualization and Communication.

The Communication tab, shown in Figure 6, allows different SASSH modules to be interfaced with the system. The timing and other low level options dealing with the serial communication is available. Convenient instructions on the left hand side explain many of the controls to the user. Also on this page is a switch to control data logging and the important Data Capture switch which starts the system.

The Visualization tab, shown in Figure 7, graphically outputs the response of the first two SASSH modules. At the top are time domain graphs of the measured accelerations. All three axes for both

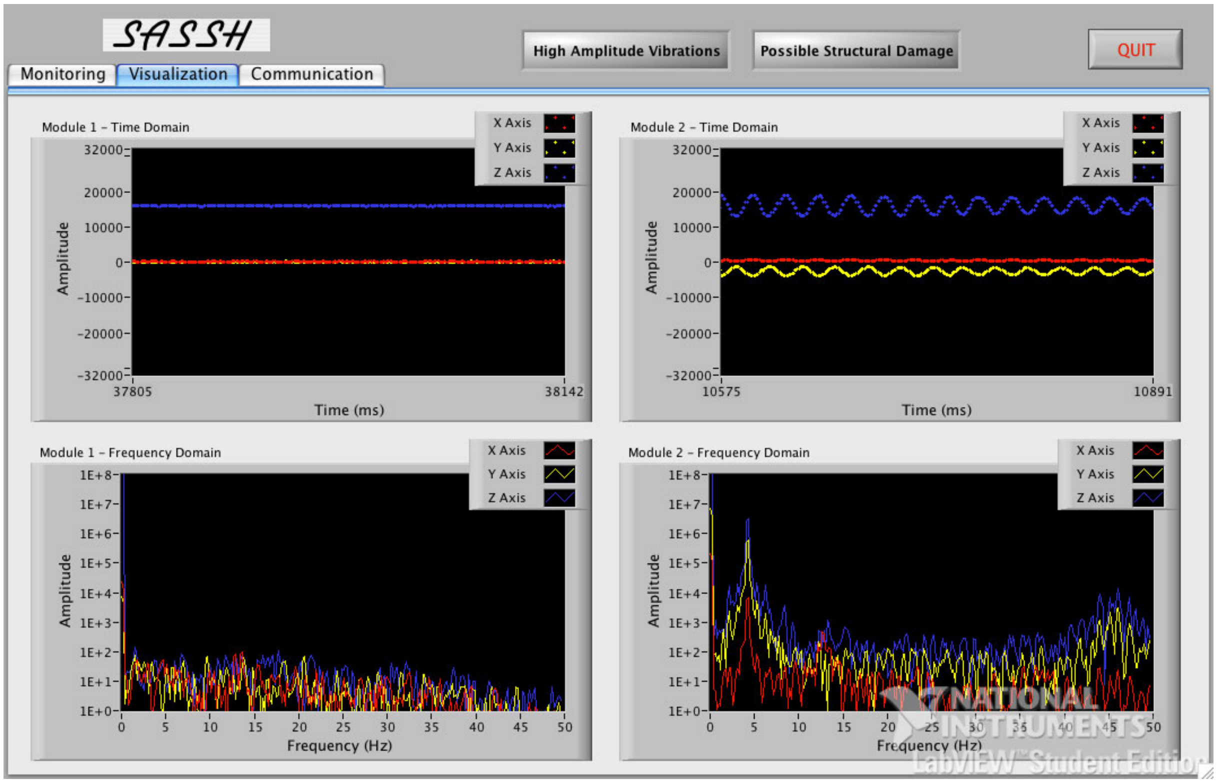


Figure 7: The Visualization tab of SASSH. This tab shows a visual representation of the accelerometer data. Data is shown for the first two modules in both the time domain and frequency domain.

modules are represented here. Currently the modules are set to read accelerations of up to two g's. One can note how the X and Y axes are nominally zero, the but Z axis is nominally half of the positive range as seen in the figure. This is the case when the SASSH module is motionless and oriented with its base parallel to the earth so that only the Z axis is effected by gravity.

The Monitoring tab, shown in Figure 8, shows the calculated values of natural frequency, damping ratio and average amplitude for each axis of each module. Present in the global pane of the program are two warning lights and a quit button. The two warning lights correspond to high amplitude vibrations or large changes in natural frequencies and damping ratios. Controls are present in the Monitoring tab to augment the sensitivity of the warning functions. Again, directions are provided in the upper right to inform the user of each feature.

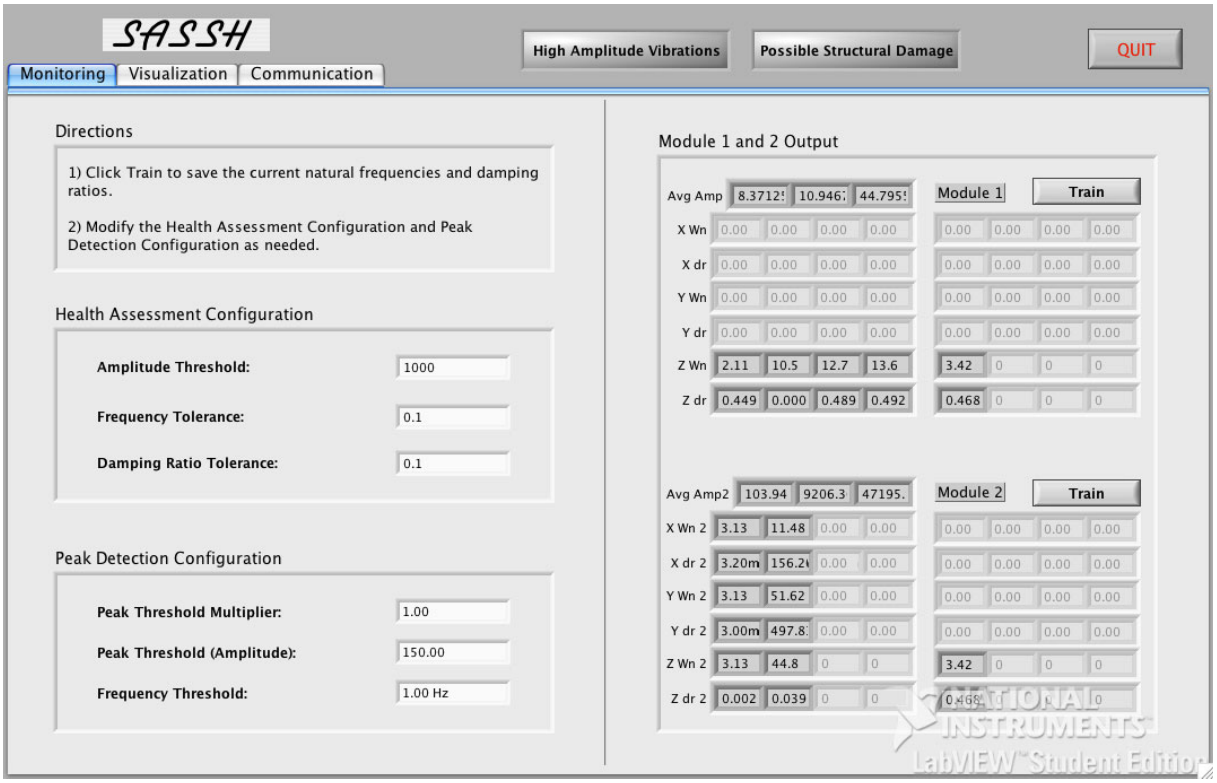


Figure 8: The Monitoring tab of SASSH. This tab shows the natural frequencies and damping ratios as processed from the accelerometer data This tab also provides methods for configuring the high amplitude and damage warnings.

Test Apparatus Requirements and Development

The team used a simple cantilever beam test apparatus as a test bed for the SASSH modules. The test apparatus was designed to produce a simple oscillation in the frequency range that was expected on a building. Unlike a building, the single simple frequency was easy to test and validate. A motor was also included that allowed the beam to be forced by an input frequency other than its natural frequency. This was designed to allow testing of more complex vibrations.

The projects final test apparatus (Figure (9)) has been developed after several rounds of experimental rearrangements (As shown in Appendix). Presently, the test apparatus constitutes a 36 long aluminum cantilever beam installed on an 18 high wooden block. The final test model uses a 12-volt DC motor along with suitable gears to produce up to 1200 RPM, which may produce vibrations up to 20 Hertz in the Cantilever Beam.

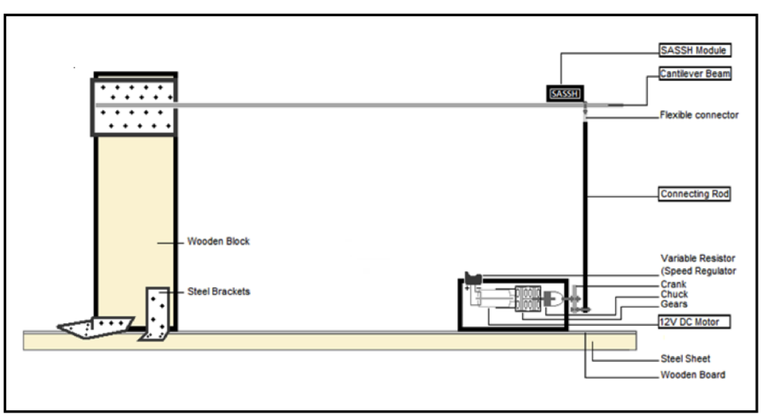


Figure 9: Detailed diagram of the final cantilever beam test apparatus.

As shown in the above figure, during testing, the SASSH Chip Module is attached to the free-end of the Cantilever Beam to measure the frequency of vibration.

The SASSH module transmits the recorded frequency of vibrations in the beam wirelessly, as well by using a wire if required, to the computer system. Then, the computer calculates a Fast Fourier Transform to distinguish resonant frequencies on a continuous time basis, and the custom-designed software interface provides alerts when natural frequency changes considerably.

Test Apparatus Results. A lab bench power supply is used to power as well as control the speed of the DC-motor. At lower voltage (non-resonance frequency), displacement in the beam increases from the fixed end to the free-end (Figure (10)). As the power is gradually increased (towards resonance frequency), the displacement in the free-end of the beam constantly increases and then, it sharply changes - with the maximum displacement in the beam shifting to its center and gradually decreasing towards both the ends (Figure (11)). The team believes that at time, the beam is vibrating at its resonant frequency (first-mode).

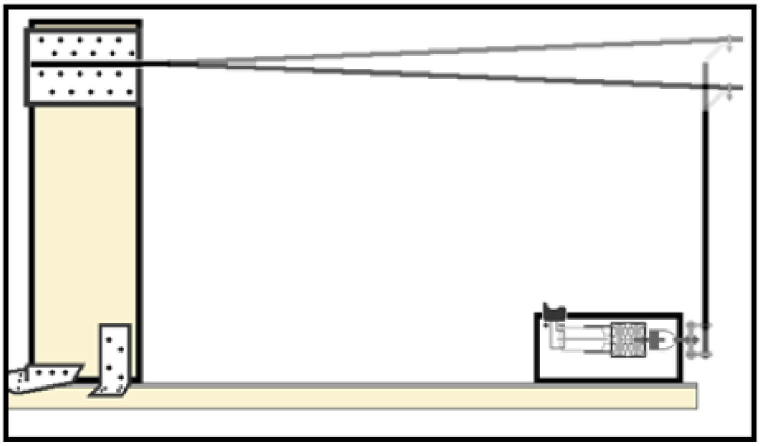


Figure 10: Vibration pattern below resonance.

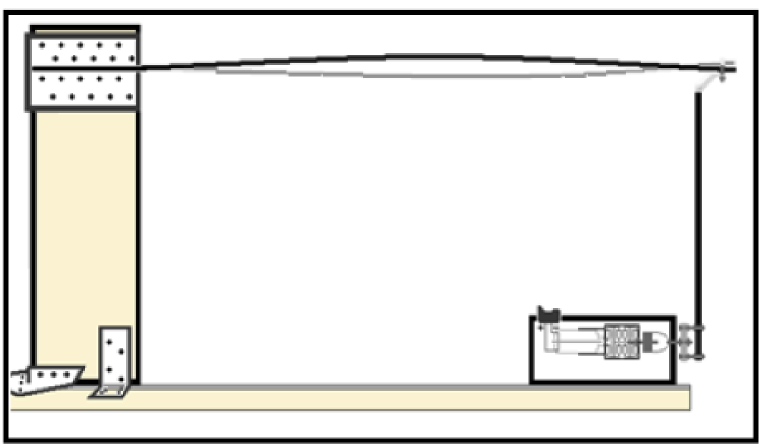


Figure 11: Vibration pattern at resonance (1st mode).

As the voltage supplied to the DC-motor is increased even further, the beam tends to move away from the vibration pattern exhibited at its resonance frequency.

More details of the test apparatus development are offered in the Appendix.

Early Testing Procedure

While the full-fledged command interface was still being developed, a MATLAB script was generated to complete all of the natural frequency and damping ratio calculations that the final program would also have to do. This allowed for the calculation methods to be tested ahead of time. Once the SASSH module was working with its first program, this script allowed for a comparison of values against theoretical data. Finally, a consistent data storage method has been used throughout the development of SASSH, so this MATLAB script can also analyze data saved from the command interface. Figure 14 shows an example plot produced by this MATLAB script.

Results

The SASSH Product

Shown in Figure 12 is the completed SASSH product. The following items are included with SASSH:

- Two complete wireless SASSH modules with triple axis accelerometers
- One incomplete SASSH module with spare parts
- One USB module with wireless modem for host computer
- One CD containing the LabVIEW command interface and all other project related files
- One product manual for the command interface
- One bright red traveling case with separate small box for USB module

Figure 12: The completed SASSH product. Included in the picture are the two completed SASSH modules, wireless USB modem (incased in smaller red box) and CD.

A close up of a SASSH module is shown in Figure 13. This ruggedized case of each module contains features that extend the capabilities of the module. Located on the bottom of the module is the external power plug. This plug fits standard type-N wall adapters. On the right side of the module is a battery power switch, power LED and external antenna connector. The module can also be powered through an internal battery, although this has not been supplied. The power switch on the case is only for the internal battery.

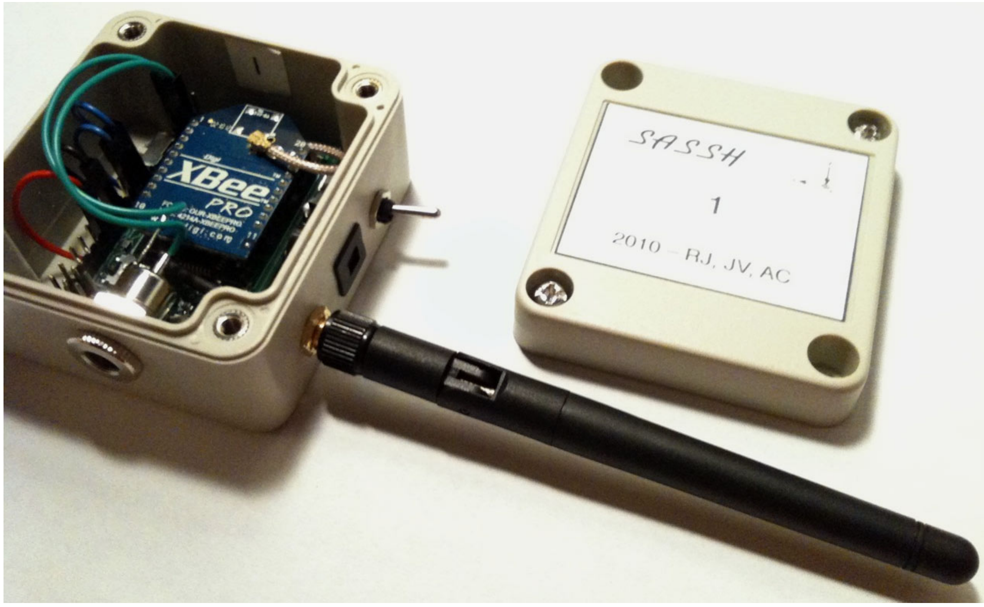


Figure 13: An up-close view of an individual SASSH module where cover removed.

The majority of the SASSH components are relatively easy to use. For example, to active a module, one simply needs to provide it power through the external power plug. However, some of the advanced features of the command interface can be confusing. The included product manual fully explains all of these advanced features.

Validation

During the initial phases of product development, a MATLAB file was created to demonstrate an algorithm capable of detecting the natural frequencies and damping ratios of vibrations from accelerometer data. While the test beam was still being constructed, fake sine wave data with a frequency of 1 Hz and a damping ratio of 0.1 was generated (using MATLAB) and tested against the detection algorithm. Figure 14 shows the output of the MATLAB algorithm. This shows that the natural frequency was calculated to within 0.02 Hz and the damping ratio was calculated to within 0.02. This proved that MATLAB algorithms were working correctly.

The cantilever test beam provided a source of simple, predictable measurements that were used to validate that the module was functioning correctly. The theoretical predictions for the beam's natural frequency were compared with MATLAB analysis of data collected from a SASSH module. Although it is not possible to produce an accurate estimate of the damping ratio from calculations, the results was within the range that Professor Udwadia predicted. The value of the damping ratio is not as critical as changes to the damping ratio, which could signify damage to the structure.

Table 1: Frequency comparison

	Theoretical	Experimental
Frequency	$3.8 \pm .4\ Hz$	$3.73\ Hz$

The frequencies found by the module are well within uncertainty of those predicted by calculations. The predicted values are shown against the measured values in the table below. Uncertainty of the theoretical calculation is based on the uncertainty of the dimensions of the beam as well as the mass of the beam and module. This proved that the accelerometers and MATLAB algorithms were working correctly together.

Finally, the LabVIEW command interface needed to be validated against the original MATLAB pro-

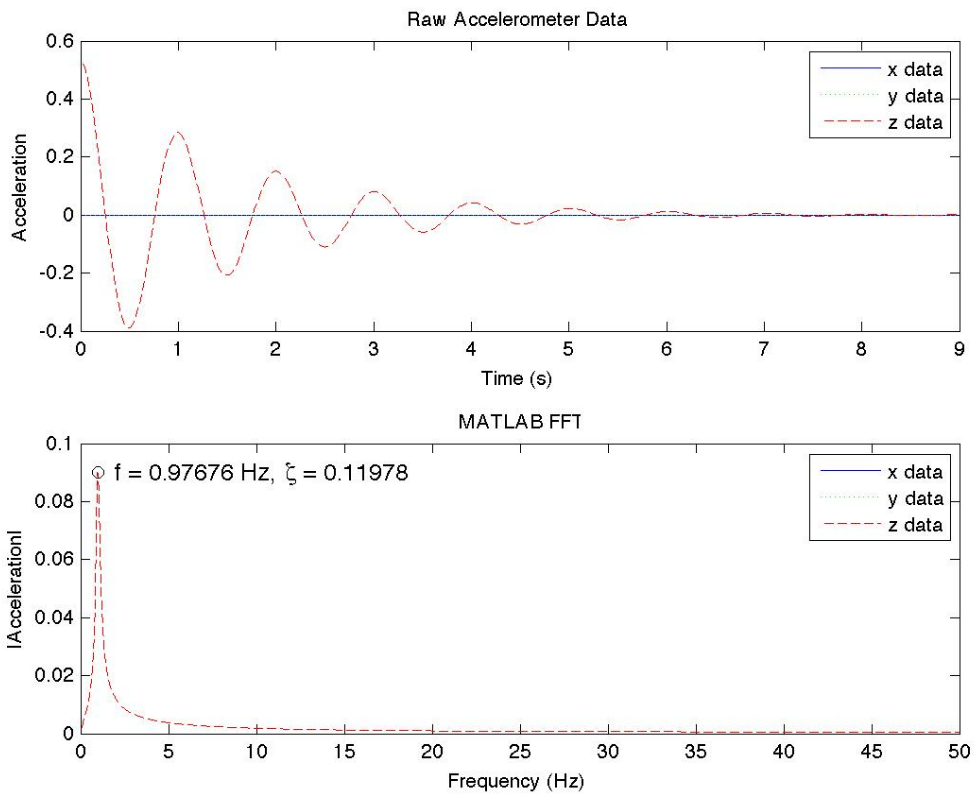
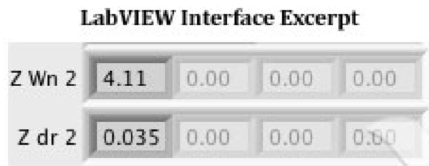


Figure 14: Result of a MATLAB algorithm to detect natural frequencies and damping ratios of accelerometer vibrations. Input sine wave had $w_n = 1$ Hz and $\zeta = 0.1$.

gram. To do this, data was analyzed by the LabVIEW interface in real time with the raw data being saved to a file. A screenshot of the real time analysis was taken and the raw data was analyzed through MATLAB after the fact. Figure 15 shows a side-by-side comparison of these two calculations. The results were nearly identical thus validating the LabVIEW algorithms.



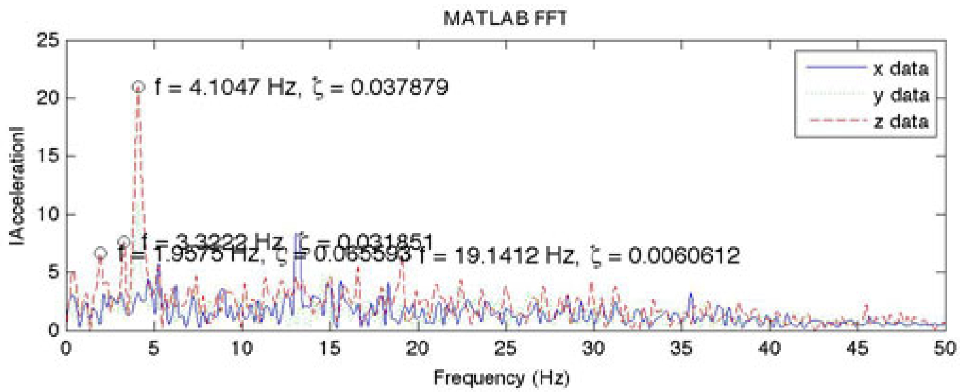


Figure 15: Data analysis of module acceleration. LabVIEW analysis was performed in real time while MATLAB analysis was performed after the fact from the raw data saved by LabVIEW. Note the strong similarities in calculated natural frequency and damping ratio in the z axis.

Discussion

Practical Functionality

The current SASSH system is very good at monitoring changes in a simple cantilever beam. It can successfully detect natural frequencies and damping ratios in realtime within 1% of the MATLAB calculations and theoretical values. It has been tested with two modules running in realtime. Up to four modules can be used in the current version of the software. From the total data bandwidth of the wireless serial line, it is estimated that up to ten modules could be supported without major restructuring of the system. Any more, and the FFT analysis would have to go onboard to decrease the total amount of data that had to be transferred.

Future Improvements

This project has plenty of room to grow. While SASSH is currently very adapt as measuring changes of a simple cantilever beam, an entire second semester could be put towards applying this product to real buildings. It is likely that to be useful on real building, much more specialized software would have to be written. During the development of the module, vibration noise tests where conducted. These showed that what looked like signal noise was actually very small amplitude local vibrations. One skilled in digital signal processing with knowledge of building vibrations would likely have a better idea on how to pick up relevant signals. Also, a skilled programmer could devise an algorithm to automatically pick out the resonant peaks. Currently, one has to manually "train" the command interface for active frequencies, but this would not be ideal for a building where it might be harder for a person to pick out these frequencies. Regardless, this is a tedious task that should be replaced by such an algorithm.

Conclusion

This project successfully produced a structural health monitoring system including the hardware, software, and validation. The system's accelerometer allows the vibration of a structure to be analyzed and recorded. It is sensitive enough to sense very small vibration such as walking and traffic, and has sufficient range to measure large vibration such as earthquakes. By remembering the building's characteristic natural frequency and observing any changes to that occur, structural changes can be noticed. In some situations, this will provide an alert for structural damage that may have been caused by an earthquake or simple wear and aging of the building.

The software interface developed allows a user with minimal experience to operate the modules and collect and analyze data in realtime. The software is extendable and scalable, just like the hardware modules, so the system can be scaled to become a large sensor network. This is clearly a first version of a product that could benefit from many improvements, however it is a great starting place and is capable of all of the requirements for which it was designed.

References

[1] Arduino. 2010. 1 Sept. 2010 <http://www.arduino.cc/>.

[2] "Audio Spectrum Monitor." Electronic Lives Manufacturing. 29 May 2005. 21 Sept. 2010 <http://elm-chan.org/works/akilcd/report_e.html>.

[3] "Fourier Transform." Wolfram MathWorld. 2010. Wolfram Research, Inc. 1 Sept. 2010 <http://mathworld.wolfram.com/FourierTransform.html>.

[4] "Fundamental Frequency." Wikipedia. 19 June 2010. Wikimedia Foundation, Inc. 1 Sept. 2010 <http://en.wikipedia.org/wiki/Fundamental_frequency>.

[5] "Q Factor." Wikipedia. 28 November 2010. Wikimedia Foundation, Inc. 1 Dec. 2010 <http://en.wikipedia.org/wiki

Appendix

Derivation of Uncertainty of Theoretical Natural Frequency

We start with the general equation for the vibration of the beam shown below.

$$\omega_n = \alpha_n^2 \sqrt{\frac{EI}{mL^4}} \text{ where } \alpha_n = 1.875, 4.694, 7.855 \tag{10}$$

We first find the partial derivatives with respect to I, m, and L.

$$\frac{\partial \omega_n}{\partial I} = \frac{\alpha_n^2}{2} \sqrt{\frac{E}{mL^4 I}} \tag{11}$$

$$\frac{\partial \omega_n}{\partial m} = \frac{-\alpha_n^2}{2} \sqrt{\frac{EI}{L^4}} m^{-3/2} \tag{12}$$

$$\frac{\partial \omega_n}{\partial L} = -2\alpha_n^2 \sqrt{\frac{EI}{m}} L^{-3} \tag{13}$$

With the partial derivatives of ω_n with respect to each variable, we can find the uncertainty of ω_n. We first find the Taylor expansion of the uncertainty, $\Delta\omega_n$.

$$\Delta\omega_n = \frac{\alpha_n^2}{2} \sqrt{\frac{E}{mL^4 I}} \Delta I + \frac{-\alpha_n^2}{2} \sqrt{\frac{EI}{L^4}} m^{-3/2} \Delta m + -2\alpha_n^2 \sqrt{\frac{EI}{m}} L^{-3} \Delta L + \textit{(higher order terms)} \tag{14}$$

Because all variables are independent and uncertainties are very small, it is assumed that the higher order terms in the Taylor series are equal to zero. To ensure that the remaining terms are all positive and non-canceling, we square the expression.

$$(\Delta\omega_n)^2 = \left[\frac{\alpha_n^2}{2} \sqrt{\frac{E}{mL^4 I}} \Delta I + \frac{-\alpha_n^2}{2} \sqrt{\frac{EI}{L^4}} m^{-3/2} \Delta m + -2\alpha_n^2 \sqrt{\frac{EI}{m}} L^{-3} \Delta L \right]^2 \tag{15}$$

The cross terms produced by squaring the right hand side of Equation (15) are assumed to be zero because the uncertainties are small and the cross terms will be negligible. Thus,

$$(\Delta\omega_n)^2 = \left[\frac{\alpha_n^2}{2} \sqrt{\frac{E}{mL^4 I}} \Delta I \right]^2 + \left[\frac{\alpha_n^2}{2} \sqrt{\frac{EI}{L^4}} m^{-3/2} \Delta m \right]^2 + \left[2\alpha_n^2 \sqrt{\frac{EI}{m}} L^{-3} \Delta L \right]^2 \tag{16}$$

Taking the square root, we find the most likely uncertainty, $\Delta\overline{\omega_n}$:

$$\Delta\overline{\omega_n} = \sqrt{\left[\frac{\alpha_n^2}{2} \sqrt{\frac{E}{mL^4 I}} \Delta I \right]^2 + \left[\frac{\alpha_n^2}{2} \sqrt{\frac{EI}{L^4}} m^{-3/2} \Delta m \right]^2 + \left[2\alpha_n^2 \sqrt{\frac{EI}{m}} L^{-3} \Delta L \right]^2} \tag{17}$$

Design Considerations/Making of the Test Apparatus

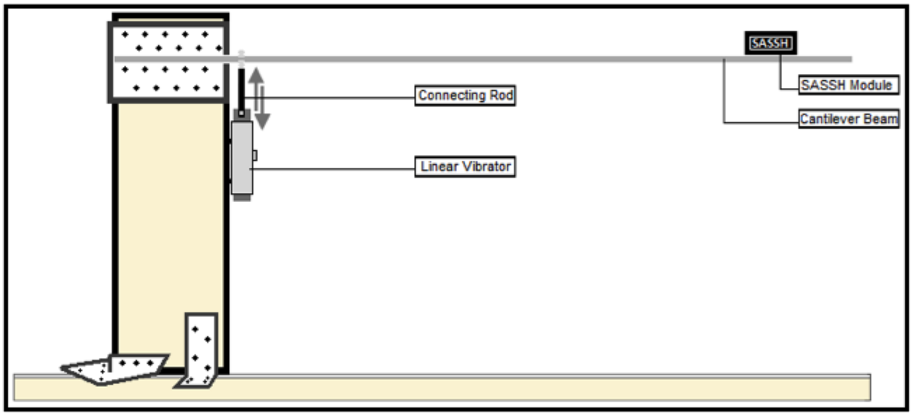


Figure 16: The early version of the test apparatus used a compressed-air powered linear vibrator. However, this design had to be scrapped because (1) compressed- air was not available for testing (2) the team had little prior knowledge to make the difficult mechanism required control the vibration frequency and (3) this component was expensive.

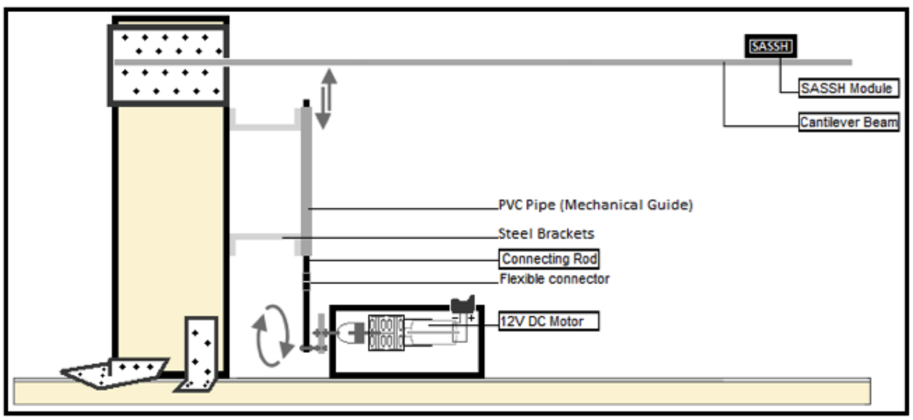


Figure 17: The second version of the test apparatus used a DC motor attached to a vertical connecting rod. When the unit was powered, the vertical rod, which was guided by a vertically fitted PVC pipe, impacted the aluminum beam to produce vibrations. However, this design provided very high noise levels when the aluminum vertical rod impacted the aluminum cantilever beam.

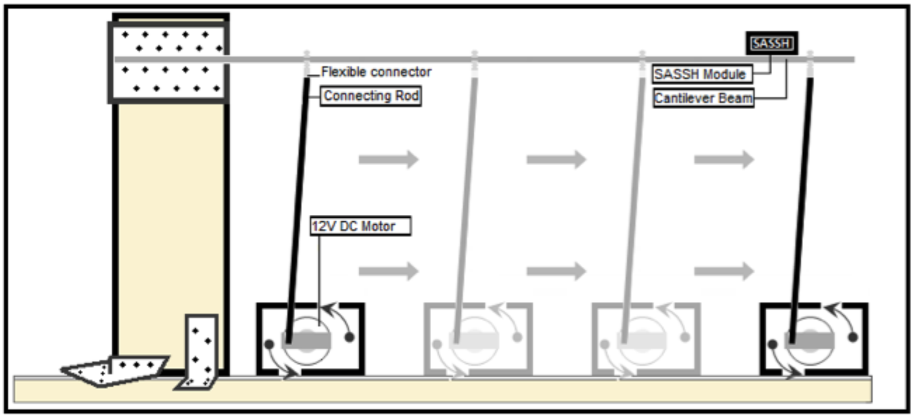


Figure 18: The next major innovation to the test apparatus involved fixing the vertical connecting rod to the cantilever beam by a flexible connector. This non-impacting design reduced the noise-levels produced, as well as ensured that there would be no lag in the cantilever beams movement, as compared to the connecting rods movement.

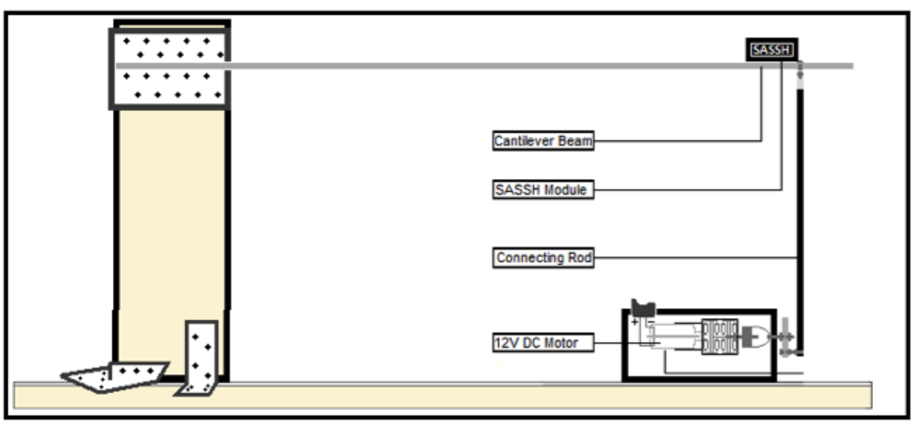


Figure 19: (cont...) Even this design did not result in visible resonance at first. The faculty therefore advised that the DC-motor and the connecting-rod be positioned under the free-end of the cantilever beam to increase chances of achieving resonance. Before permanently fixing the DC-motor and the connecting-rod under the free-end of the cantilever- beam, the team experimented and positioned the motor at various spots (As shown in Figure C, and this confirmed that the final position advised by the faculty was indeed the most appropriate for achieving...