



## Frequency Domain Concepts for EE Freshman

**Dr. Christina Howe, University of Evansville**

**Dr. Dick Blandford, University of Evansville**

Dick Blandford is the Chair of the Department of Electrical Engineering and Computer Science at the University of Evansville.

# Frequency Domain Concepts for Electrical Engineering First Year Students

## Abstract

A major component in the retention of students for any program is to get the students excited about the field they have chosen to study. At the University of Evansville, EECS is a combined department that has programs in electrical and computer engineering, and in computer science. The first engineering course, required of all freshmen, is a three-credit course called Engr 101. One of the main objectives of this course is to awaken students to the exciting possibilities all three disciplines offer. Another main objective of this course is to assist the students in choosing which discipline best fits their skills and interests. We get a significant number of major changes between the three disciplines during the first year. As a result, we have made the first year nearly identical for all three majors, and in Engr 101, we make an effort to teach all students what each of the three disciplines is about. Engr 101 therefore has three parts: one for electrical engineering, one for computer engineering, and one for computer science. Each part takes one-third of the term and all three parts are taught in parallel with instructors rotating every 12 to 14 class days.

In electrical engineering, the challenge is this: What can you teach freshmen in 12 lectures which will give them a good idea of what electrical engineering is about and allow them to make an intelligent choice of major? This paper presents a solution to this problem that has been well received by students and has good assessment data for the first year. In addition, we present the topical outline for the computer engineering and computer science portions of the course.

The focus of the EE portion of the course is frequency domain concepts in linear systems. Implementation involves the use of MATLAB<sup>®</sup><sup>1</sup>, difference equations, the Fourier transform, and sound files. Since freshmen do not have the math background of our juniors we view the Fourier transform as a correlation between an input signal and a complex sinusoid. MATLAB<sup>®</sup> allows for easy manipulation of difference equations, the z-transform need not be mentioned, and the course has no time for theorems or proofs.

Sound files present an easy way to introduce real data into the course and all students are able to see the frequency spectrum of sounds in their environment. In a final project they determine the speed of a passing automobile by observing its Doppler shift as it passes. Through this project students gain an appreciation of the exciting real-world problems that they can solve with a degree in electrical engineering.

## Introduction

Retention in regard to electrical engineering students can be defined as the percent of the entering class who continue on to get a degree in any major, or, it can be defined as the percent of students who continue on to get a degree in electrical engineering. In this paper we will call the former "university retention" and the latter "electrical engineering retention". We are going to further limit our definition of retention to the first year – that is, we are interested in the number of students who successfully complete their freshman year and go on to the sophomore year.

At the University of Evansville the university retention rate is above average. The latest data indicates that, for electrical and computer engineering students, the freshman to sophomore university retention rate was greater than 95% and has improved in each of the past 5 years. This improvement is due in part to efforts made at the University level. The electrical engineering retention rate is considerably lower. The most recent data shows that the freshman to sophomore electrical engineering retention rates is about 70% which means that roughly a third of the students who enter with an electrical engineering major switch to some other program by the end of the freshman year.

At the University of Evansville, electrical engineering, computer engineering, and computer science programs are part of a combined department. For the past twelve years, all freshmen declaring a major in the department have been required to take Engr 101 – a three hour course taught by faculty in the major. For the past six years this course has had three sections: one in computer science, one in electrical engineering, and one in computer engineering. Each section teaches a topic for only 13 lecture periods after which students are traded and the 13 lectures are repeated. At the end of the term, all students in EECS who complete the course have in effect a short course on computer science, a short course on computer engineering, and a short course on electrical engineering.

The objective is to provide students enough information to allow them to make an informed decision as to their major with the hope that the retention rate within the major can be improved.

This paper discusses the structure of the electrical engineering section of Engr 101, which focuses on frequency domain concepts as representative of the electrical engineering program.

Previous papers have shown a positive impact on gaining student interest through projects done in freshman engineering course<sup>5, 6, 7</sup>. However, none of these use the 3 section model used in UE's Engr 101 with DSP the lead topic of the EE section.

### **Course Objectives and Outline**

The overall objective of Engr 101 is to improve the retention in the major for each of the three programs. Each section of the course attempts to do this by providing students with objective information sufficient to make an informed decision about the major and by providing a mentoring environment for freshman.

There are thirteen lecture slots for each major. Faculty who teach each section fill twelve of these slots with lectures, demos, projects, and testing. The thirteenth day is reserved for a "mingle". The mingle concept is that following each section, one day is allotted to allow students and faculty to get together for informal discussion over refreshments. All department faculty are invited to the mingles and considerable effort is made to ensure that all of the students meet the faculty on an informal basis. Students are encouraged by faculty to "drop by" the office and talk about the types of work done by professionals in the major to get a better feel for what the major is about.

Computer science section

The computer science section uses C# to teach the fundamentals of program structure. In the four-week short class students typically write a program which has loop and if structures and an introduction to graphics. The class typically writes a game or animation program for a project. At the end of this session students are expected to understand how computer programs work, what a compiler does, and how an algorithm might be developed to do something useful on a computer.

#### Computer engineering section

The computer engineering section provides a mix of hardware and software and is by far the busiest section. Student teams typically spend several hours in the labs outside of class assembling and testing their project. In a typical project, students are provided with a platform having wheels and motors. Students add their own sensors with the objective of building an autonomous vehicle which follows a black line of tape around the lobby and in and out of labs. This section also uses upper-classmen to provide additional assistance in the labs outside of class. At the end of this session, students are expected to understand the relationship between hardware and software.

#### Electrical engineering section

Since the mid-1990s there have been several attempts to introduce digital signal processing to electrical engineering students at the sophomore year.<sup>2,3,4</sup> This effort has met with some success. The Infinity Project<sup>8</sup> has demonstrated that it is possible to introduce high school students and university freshmen to DSP concepts. As a result we considered introducing DSP, or linear systems concepts, in the freshman year as a way to introduce students to the discipline of electrical engineering. While freshmen do not have the math skills necessary to DSP implementation, they can be taught some of the concepts such as sampling, aliasing, filtering, etc.

The electrical engineering section uses linear systems as representative of electrical engineering upper level coursework and attempts to introduce students to the frequency domain. This is a challenge because about 60% to 80% of the class is taking a first course in calculus and is being introduced to differentiation and integration at the college level. (The remainder of the class has had sufficient calculus in high school to be beyond the first calculus class.) MATLAB<sup>®</sup> is used to get into the frequency domain, and the spectrum analyzer and oscilloscope are used as demos.

Students use a recorder to record various sounds which are then analyzed in MATLAB<sup>®</sup>. For example, students record a passing automobile on the street and the file is analyzed in the frequency domain and the Doppler shift is used to estimate the speed of the car as it passes.

The class has assignments in MATLAB<sup>®</sup> and Fourier analysis is used throughout.

### **Frequency Domain Concepts**

#### *MATLAB<sup>®</sup>*

The first three lectures are on the use of MATLAB<sup>®</sup>. Students have a daily assignment that reinforces the instruction in the classroom. In addition, a summary of each lecture is made available as a fifteen-minute video online and students are expected to view this video *before* they come to class. There is no textbook and the short online summaries provide links to

examples and tutorial resources for those needing more help. Most have never seen MATLAB<sup>®</sup> before and many have not done any computer programming in high school. By the third day, students are adept at creating m-files, using vectors for calculations, and doing plots of data.

### *Sound Files*

MATLAB<sup>®</sup> provides many functions for the manipulation of sound files with a `wav` extension. The class includes a short discussion of sound files and how they are encoded in the `wav` format. This introduces sampling frequency and recording resolution concepts. In an in-class demo, students listen to a sound file with spoken words that is played back using 16-bit recording resolution. MATLAB<sup>®</sup> is used to reduce the quantization level of the file to successively lower quantization numbers. Students are amazed to learn that you can clearly understand speech even if it is recorded using only a single bit. With this insight into the frequency content of signals, we introduce the Fourier series and the Fourier transform. This introduction focuses on the concepts and not on the mathematical manipulation necessary to carry out the operations.

MATLAB<sup>®</sup> functions are available to read multi-channel sound files into a vector, create a new vector and write it to a new sound file, view sound files in both the time and frequency domains, and apply filters to modify the frequency content of sound files.

### *Concepts*

There are numerous concepts introduced in this class, with the most difficult being the transformation to the frequency domain which is done by the Fourier series and the Fourier transform.

The equations for the Fourier series are given to the students with some historical insight into how and why they were developed. The instructor finds the coefficients for the Fourier series for a square wave and a triangular wave (since these are commonly available on a function generator). The spectrum analyzer and oscilloscope are introduced and students can view the time and frequency domain presentations for the two signals. In MATLAB<sup>®</sup> students add sinusoidal vectors from the Fourier series calculations and show that they can approximate a square wave. Figure 1 illustrates how this is done.

The Fourier transform is easier to understand from a conceptual point of view than is the Fourier series since it can be done discretely (without integrals) and it can be viewed as the cross-correlation between a signal and a complex sinusoid at various frequencies.

The equation for the discrete Fourier transform is:

$$F(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j \frac{2\pi kn}{N}}$$

Using Euler's identity this can be written as:

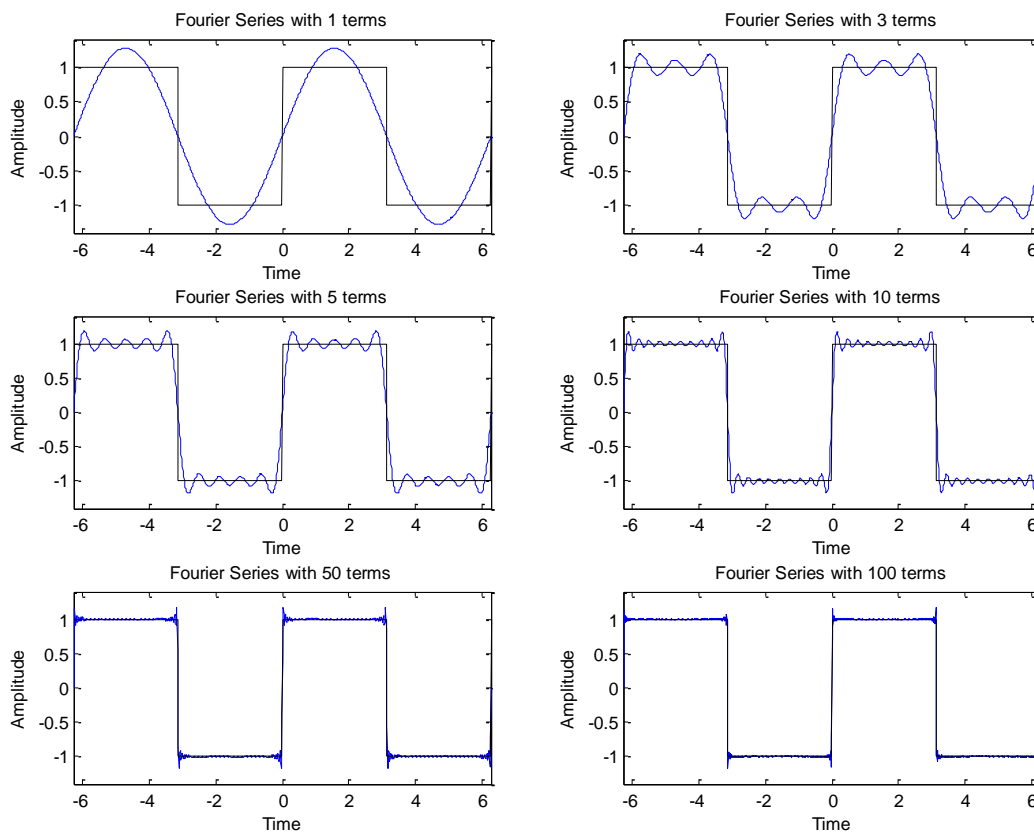
$$F(k) = \sum_{n=0}^{\infty} x(n) \cos(2\pi kn / N) - j \sum_{n=0}^{\infty} x(n) \sin(2\pi kn / N)$$

Students use MATLAB<sup>®</sup> to evaluate this equation for several common signals and plot the result. For simplicity we look only at the magnitude of the result – not the phase. For example, Figure 2

shows the result of applying this equation to a signal consisting of a sinusoid at 100 Hz which is truncated in time.

MATLAB's `fft` function is introduced. A demonstration shows that it produces the same results as a spectrum analyzer and the correlation equation. The class spends a full day doing frequency plots of various waveforms.

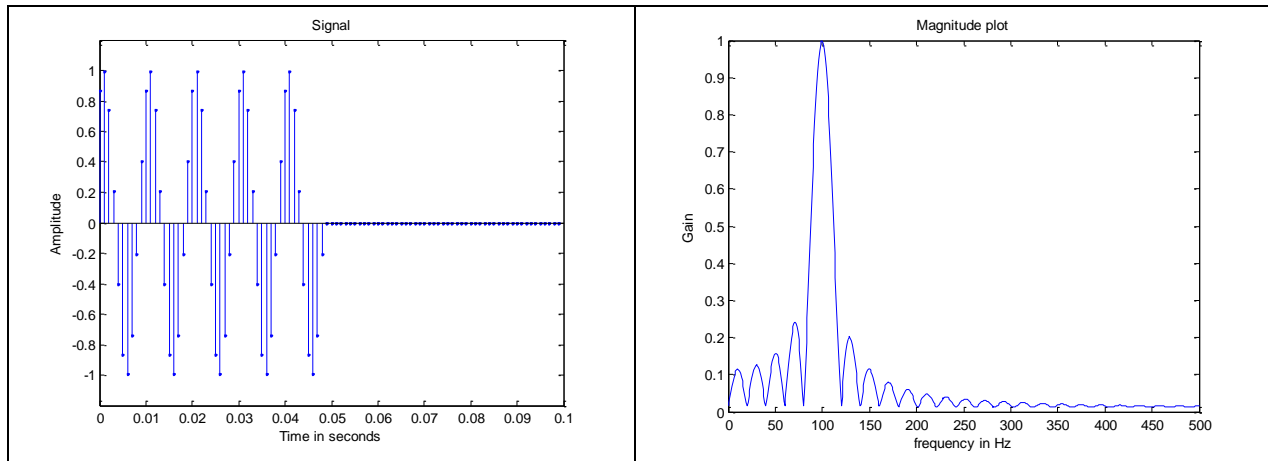
To make these concepts more vivid, students are assigned to sketch and approximate frequency spectrum for several signals, to apply the discrete Fourier transform and to verify their estimates. More than 80% of the class becomes adept at accurately estimating the frequency content of simple signals based upon their best guess at which sinusoids correlate with the signals.



**Figure 1**

A square wave created by adding sinusoids specified by the Fourier series.

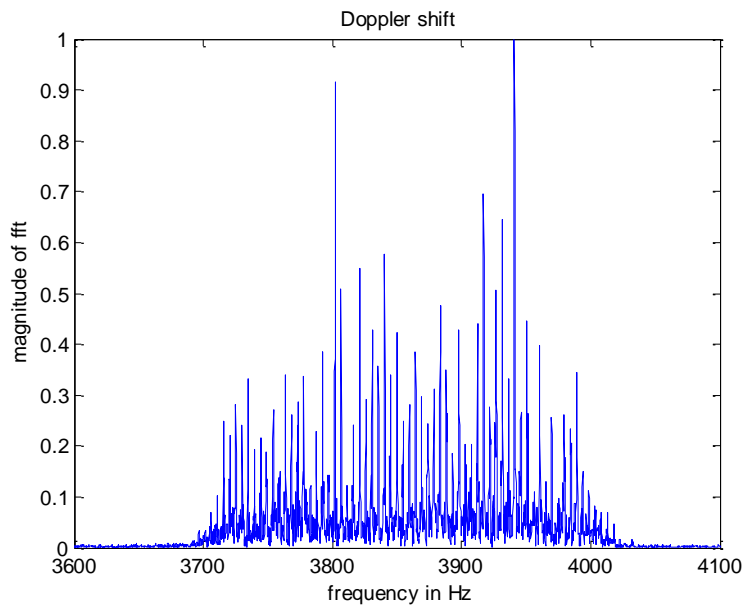
As an in-class exercise, we record several students saying the vowels "a", "e", "i", "o", and "u" as words in some arbitrary sequence. Using frequency analysis alone, students then decipher the correct sequence in which the vowels were spoken. Similar things can be done with the words "yes" and "no" and with some of the digits.



**Figure 2**

A sinusoid truncated in time and its discrete Fourier transform.

In another exercise, a beeper is mounted on a window fan blade and rotated at three different speeds. The beeping sound is recorded and students use the Doppler shift in the tone to calculate the rotational speed of the fan blade. Figure 3 shows the Doppler shift produced by one such beeper experiment.



**Figure 3**

A beeper produces a single frequency tone. When placed on the end of a spinning fan blade, the frequency spectrum is spread out due to a Doppler shift.

In the final three lectures we introduce filtering concepts. Since students don't have the mathematical background to understand the z-transform, we introduce a shifting operator and write filter equations in terms of difference equations. A classic example that is easy to understand is the running average filter. In class we evaluate a difference equation for a running average filter using MATLAB<sup>®</sup> and apply it to a noisy signal. The low-pass frequency effects of

a running average filter can be heard as a sound file is replayed and is evident from the frequency analysis which can be plotted.

Since MATLAB<sup>®</sup> treats difference equations and their evaluation as a sequence of coefficients it is easy to introduce more sophisticated filtering types such as the Butterworth and Chebyshev filters. We spend the last day filtering music files.

#### Assessment

There are two forms of assessment for this class. The first is a set of 15 questions related to concepts in the class such as the sampling theorem, Fourier transform, the Doppler effect, etc. All of these questions are concept based to test understanding rather than mathematical expertise. The second form of assessment is done at the completion of the course. A group of 9 students are chosen at random from the class to participate in a focus group conducted by an administrative assistant not associated with the class. The assistant has a set of questions to stimulate discussion. Results are recorded and summarized and used to improve the course in the following year.

The results of the concept questions and the focus group summary is given below in Tables 1 and 2.

#### Concept Questions Assessment

There were 3 questions on each of five topics. Thirty-six students took the quiz and the average percentage score on each topic is given.

**Table 1**  
Concept questions results

Topic	Average score
MATLAB <sup>®</sup> fundamentals Can they write the code to plot an equation. Can they evaluate an equation with vector variables. Can they read/write a sound file.	84.8%
The Sampling Theorem Can they write the sampling theorem in words. Can they determine the minimum sampling frequency of a signal with known frequencies. Can they explain aliasing.	55.5%
Instrumentation Can they explain what an oscilloscope does. Can they explain what a spectrum analyzer does. Can they read an oscilloscope display in frequency and amplitude.	95.5%
Doppler Shift Can they explain in words what the Doppler shift is. Can they explain in words how to evaluate the Doppler shift in MATLAB <sup>®</sup> . Can they give practical examples of the Doppler effect.	83.9%
Fourier Series and Fourier Transform Can they explain the difference between the Fourier series and the Fourier transform. Can they explain a square wave in terms of its frequency content. Do they understand the Fourier transform plot in MATLAB <sup>®</sup> .	57.6%



## Focus Group Assessment

For the focus group nine students were chosen at random and asked to participate in a focus group conducted by an administrative assistant. To stimulate discussion the assistant asked the group four open-ended questions. She recorded the results, scrubbed them of personal data, and produced a written report in the form of comments and criticisms. A summary relevant to the EE session is given below in Table 2.

**Table 2**  
Focus group summary.

Topic	Comments, remarks, and criticisms
Objectives	<p>Students believed that all three sessions met their stated objective of giving them a reasonably good idea of what the discipline was about.</p> <ul style="list-style-type: none"> <li>• EE session was the "heaviest".</li> <li>• EE session made it clear that what they were learning in calculus would be critical to their success in EE later on.</li> <li>• MATLAB<sup>®</sup> was a big hit. Students felt that they did enough in MATLAB<sup>®</sup> to be able to continue on their own.</li> <li>• Video summaries of the lectures were very helpful.</li> </ul>
Likes and Dislikes about the course	<p>In general students were very positive about this course.</p> <ul style="list-style-type: none"> <li>• The rotation among disciplines was given high marks.</li> <li>• The mingles at the end of each session allowed students to get to know others in the class as well as all of the department faculty.</li> <li>• Homework in the EE session was very time consuming and difficult to do without the online videos.</li> </ul>
Future changes	<p>It was suggested that instead of rotating each session, we rotate half the class to one session and the other half to the other session. For example, half of session 1 would rotate to session 2 and the other half would rotate to session 3. This method would allow students to get to know more of their classmates. We liked this idea and will implement it next year.</p>
General Comments	<p>It was suggested that we make more of an attempt to integrate student personal computers into the sessions rather than rely on classroom machines.</p>

### Summary

This paper presents one organizational structure for introducing freshmen to the closely related disciplines of electrical and computer engineering and to computer science. Our assessment data indicates that the course was well received. We have been measuring freshman to sophomore retention in electrical and computer engineering and in computer science and it has been improving steadily over the last five years. It remains to be seen whether or not this change in the course content will make any difference in the rate of change. We intend to continue monitoring our retention rate.

After doing the EE session using DSP and sound files, an independent faculty group reviewing this course as part of our annual assessment concluded that this is a viable option provided we stick to concepts. Freshmen have neither the mathematical ability nor the academic maturity to handle the mathematics and rigor of a DSP course.

## References

1. MATLAB® is a product of the Mathworks. Version R2012a
2. Chiang, K.H., Evans, B.L., Huang, W.T., Kova, F., Lee, E. A. Messerschmitt, D.G, Reekie, h.J., and Sastry, S. S., Real-Time DSP For Sophomores Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings, 1996 IEEE International Conference, May, 1996.
3. Hoffbeck, J.P. "Enhance Your DSP Course With These Interesting Projects" ASEE Annual Convention presentation, 2012, San Antonio, Texas.
4. Black, B.A., "Implementing DSP First at Rose-Hulman", Proceedings of IEEE SPE, 2000 - spib.ece.rice.edu
- McClellan, J.H., Schafer, R. W., Yoder, M.A., "Experiences in teaching DSP first in the ECE curriculum", Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference, April, 1997
5. Maher, Robert C., Becker, James, Sharpe, Tia, Peterson, James, Towle, Bradford A., "Development and Implementation of a Robot-based Freshman Engineering Course", ASEE Annual Convention presentation, 2005, Portland, Oregon.
6. Myler, H.R., "Early Electrical Engineering Concepts Engagement in a Freshman Level Introductory Course", ASEE Annual Convention presentation, 2004, Salt Lake City, Utah.
7. Jensen, James N., "A Case Study Approach to Freshman Engineering Courses", ASEE Annual Convention presentation, 2003, Nashville, Tennessee.
8. URL: <http://www.smu.edu/lyle/infinity>