SAE TECHNICAL PAPER SERIES

2007-01-1509

Knock Signal Conditioning Using the Discrete Fourier Transform and Variable Detection Window Length

Matthew J. Kearney
Delphi Electronics & Safety

Reprinted From: Electronic Engine Controls, 2007 (SP-2087)

ISBN 0-7680-1636-3

9 780768 016369

2007 World Congress
Detroit, Michigan
April 16-19, 2007

SAE International™

400 Commonwealth Drive, Warrendale, PA 15096-0001 U.S.A. Tel: (724) 776-4841 Fax: (724) 776-0790 Web: www.sae.org

By mandate of the Engineering Meetings Board, this paper has been approved for SAE publication upon completion of a peer review process by a minimum of three (3) industry experts under the supervision of the session organizer.

ISSN 0148-7191

Printed in USA

2007-01-1509

Knock Signal Conditioning Using the Discrete Fourier Transform and Variable Detection Window Length

Matthew J. Kearney
Delphi Electronics & Safety

ABSTRACT

As the speed and memory capabilities of microprocessors have increased, it has become more popular for the signal conditioning of knock sensor outputs to be performed entirely within the microprocessor. One method of this signal conditioning process utilizes the Discrete Fourier Transform (DFT). It is common for systems that use this method to limit the knock detection window to one length across all RPM and load points to reduce computation and memory constraints on the processor. This paper will summarize this method and explain a further enhancement, variable window length, that may become more popular as processor capabilities increase.

INTRODUCTION

One of the keys to maximizing the power output of an Otto internal combustion engine is to advance the spark timing as close to Maximum Brake Torque (MBT) spark as possible. MBT spark is the spark advance that produces the most torque at a given operating point (cylinder temperature, RPM, engine load, and air/fuel ratio, etc [1].)

A limiting factor in achieving MBT spark is detonation, or spark knock. Spark knock occurs when rising pressure and heat caused by the advancing flame front in the combustion chamber cause a secondary explosion. The pressure waves created by the explosions collide, causing a rapid spike in both pressure and heat [2]. The early stages of detonation cause a distinctive “ping” as the colliding pressure waves resonate in the cylinder and the surrounding engine block [3]. Left untreated, long term detonation can raise cylinder temperatures high enough to lead to preignition, a dangerous symptom in which the air/fuel mixture in the cylinder is ignited independent of the timed spark event [4]. Preignition can lead to engine damage, particularly to the spark plugs, piston rings, and pistons [5].

If spark knock occurs at a spark advance before MBT spark, the operating point is said to be “knock limited.” At lower RPM and loads, it is often possible to advance the spark to MBT spark without being knock limited. At higher RPM and loads, it is common to be knock limited, especially in environments with high temperatures and low relative humidity. The knock limited spark advance at a given operating point decreases as the ambient temperature increases [4]. Also, the knock limited spark advance decreases as the relative humidity decreases [4].

In order to run as close to MBT spark as possible, given unknown environmental conditions, most vehicles today employ some form of knock detection system. With a knock detection system vehicles can be sparked assuming cooler ambient temperatures and higher humidity. The vehicle can rely on the knock detection system to retard the spark advance as environmental conditions change.

Knock detection systems often have three main sections:

1. Knock Sensor

2. Analog Input/Conditioning Circuitry

3. Knock Detection and Control Algorithm (internal to the engine control unit)

<u>Knock Sensor</u>

Most knock sensors today are piezoelectric sensors. As a general rule, four cylinder engines use one knock sensor. V style engines with six or more cylinders often use two knock sensors, either mounted on the outside of the block or in the V. These sensors output a differential AC signal that is conditioned before being read by the analog to digital converter in the processor of the engine control unit.

Analog Input/Conditioning Circuitry

In some systems, custom analog ICs are used to condition the AC knock sensor output. These ICs, with the aid of the CPU, window the knock signal to regions where knock occurs. They then convert the differential AC signal from the knock sensor to a constant value per cylinder event through integration, peak detection, or other means.

With the increase in processor capability, the signal conditioning has moved progressively more internal to the processor. The focus of this paper will explore one of the methods of knock signal conditioning that is common in the realm of digital signal processing, the single point Discrete Fourier Transform. This paper will describe this method and discuss ways to increase system performance by varying the knock detection window length while keeping memory storage at a minimum.

It should be noted that, even with the increase in processor capability, a minimum amount of external circuitry is still needed for anti-alias filtering when using digital signal processing techniques such as the Discrete Fourier Transform. The anti-alias filter (AAF) helps attenuate higher frequency signals that can be aliased in the frequency range we wish to observe. An ideal AAF would pass all frequencies up to the Nyquist frequency and block all frequencies above the Nyquist frequency [6]. In general, the complexity of the AAF is determined by the sample rate of the A/D in the processor. For high enough sample rates the AAF can be simplified (in some instances down to a one pole filter) because there is plenty of room for the filter to "roll off" before it reaches the Nyquist frequency. However, as the sample rate decreases, the Nyquist frequency approaches the desired frequency range, and it becomes necessary for the filter to have a steeper "roll off."

Knock Detection and Control

The knock detection algorithm determines whether the conditioned knock sensor signal for the current cylinder event is a knock event. If the detection algorithm determines that there is knock, the knock control algorithm will suppress the knock by retarding spark, increasing air fuel ratio, etc [5]. A simple detection algorithm will be discussed in this paper that is specifically designed for the variable window single point DFT method.

THE DISCRETE FOURIER TRANSFORM

DFT EQUATION SIMPLIFICATION

The Discrete Fourier Transform (DFT) will be used to convert the windowed time-based knock sensor signal to the frequency domain. It can be used to detect mode frequencies of the combustion cylinder that are resonated by knock events. In this section we will simplify the Discrete Fourier Transform equation to meet our needs. The Discrete Fourier Transform is represented by the following equation:

$$X(e^{j\omega}) = \frac{1}{N}\sum_{k=0}^{N-1} x(k)e^{-j\omega k} \tag{1}$$

Substituting the Euler equation into the DFT equation will simplify calculations for our use. Recall the Euler equation below:

$$e^{-j\omega t} = \cos \omega t - j\sin \omega t \tag{2}$$

Also, remember that one period is equal to 1 divided by the sample frequency. The current value of time, *t*, is equal to the current value of k divided by the sample frequency.

$$t = \frac{k}{Fs} \tag{3}$$

These equations can be substituted into the Discrete Fourier Transform equation, which yields:

$$X(e^{j\omega}) = \frac{1}{N}\sum_{k=0}^{N-1} x(k)(\cos(\omega\frac{k}{Fs}) - j\sin(\omega\frac{k}{Fs})) \tag{4}$$

N represents the number of samples that we will be analyzing. We will designate *n* as the bin number. When we calculate the DFT, we will produce a complex number representing the magnitude and phase of the frequency for *N* frequency bins. The current bin number is represented by *n* such that *n* multiplied by Fs/N (the sample frequency divided by the number of samples) equals the frequency represented by bin *n.* This will become clearer in the example in the next section.

$$f = \frac{nFs}{N} \tag{5}$$

If we substitute ω for *f* in Equation 5, we obtain the result we need for our final substitution into the DFT equation.

$$\omega = 2\pi \frac{nFs}{N}$$

(6)

Substituting into the Discrete Fourier Transform equation yields the following revised DFT equation:

$$X(e^{j\omega}) = \frac{1}{N}\sum_{k=0}^{N-1} x(k)(\cos(2\pi \frac{n}{N}k) - j\sin(2\pi \frac{n}{N}k))$$

(7)

The Discrete Fourier Transform equation as represented by Equation 7 is very useful when calculating a single point (bin) for knock detection purposes. The next section will discuss how knock system terminology relates to each variable in Equation 7.

KNOCK DETECTION WINDOW

KNOCK DETECTION WINDOW EXAMPLE

The knock detection window commonly occurs after top dead center (ATDC) and closes before TDC of the next cylinder. The figure below shows a raw knock sensor signal that is windowed by a detection window.

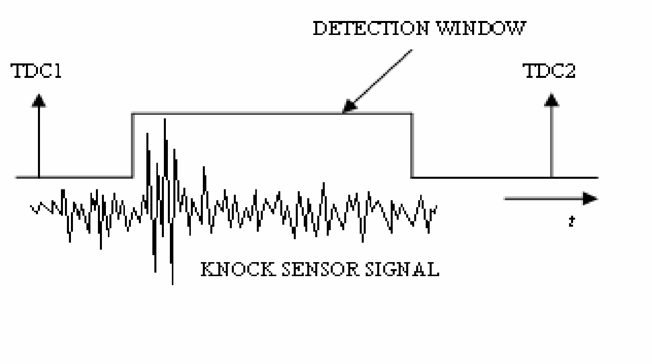

Figure 1

For our example, we will assume a 40 kHz sample rate and a 3 ms knock window. As mentioned in the previous section, *N* equals the number of samples. For our case, this is the number of samples collected in one detection window, which is equal to 120 samples. Substituting this into Equation 7 results in the following:

$$X(e^{j\omega}) = \frac{1}{120}\sum_{k=0}^{119} x(k)(\cos(2\pi \frac{n}{120}k) - j\sin(2\pi \frac{n}{120}k))$$

(8)

To determine the frequency content across the entire spectrum (in this example, 0 to 40 kHz), the DFT equation would be calculated for each value of *n* from *n* = 0 to *n* = 119. Computation time can be saved if we can isolate frequencies for which to test for the presence of knock. Many systems may select multiple frequencies for which to test for knock, but for simplicity of the example we will choose one frequency.

Knock frequencies are dependent upon several factors, such as bore diameter, combustion temperature, etc. As mentioned in the previous section, *n* equals the *n*th bin, which is equivalent to the frequency that we wish to observe. For our purposes, this frequency, or bin number, is the frequency of the knock vibration. The sample rate divided by the number of samples equals the bin size.

$$\frac{40kHz}{120samples} = 333Hz/bin$$

(9)

Calculating the full DFT would give us 120 bins, each 333 Hz wide, from 0 to 40 kHz. We will assume that the frequency we wish to observe is a first circumferential knock frequency of 6 kHz. If we wish to test for 6 kHz knock, we need to observe the 18^{th} bin (*n* = 18.)

$$\frac{6kHz}{333Hz/bin} = 18^{th} bin$$

(10)

It should be noted from sampling theory that the result of the DFT above the Nyquist frequency will mirror the result below. Since the sample frequency of our example is 40 kHz, our effective Nyquist frequency is 20 kHz. Therefore, nothing is gained from calculating the DFT for *n* = 61 to *n* = 120 as *n* = 61 will produce the same DFT result as *n* = 59 and so on.

Returning to our equation substitution, the 18^{th} bin, or *n* = 18, is representative of 6 kHz. Our DFT equation is now in a manageable form for software computation.

$$X(e^{j\omega}) = \frac{1}{120}\sum_{k=0}^{119} x(k)(\cos(2\pi \frac{18}{120}k) - j\sin(2\pi \frac{18}{120}k))$$

(11)

The cosine terms for k = 0 to k = 119 can now be calculated ahead of time and stored in a 1x120 lookup table. The sine terms can be acquired from the same table as the cosine terms as they are shifted by 90 degrees.

Calculating the DFT yields a complex number from which the magnitude and phase of the knock signal at 6 kHz can be determined. For our purposes, the phase is not important. The actual magnitude can be calculated by taking the square root of the squares of the real and imaginary portion of the result. There are several approximations for magnitude, however, that can be employed by software to save calculation time.

WINDOW LENGTH LIMITATION

For time based knock detection and control, the knock detection window is traditionally calibratable in both start angle and length. A common limitation with the Discrete Fourier Transform method is the limiting of the knock window length to one value over the entire operating range.

Why would most DFT systems limit the detection window to one length over the entire operating range? This is best explained if we consider our calculations of both *n* and *N* in the previous section. If we change the length of the knock detection window and the sample rate remains constant, we change the number of samples in the knock detection window, *N*. Also, if we examine our calculation of *n*, changing the value of *N* effects the bin size, since the number of samples is changing and the sample rate remains a constant 40 kHz. This means that the 18^{th} bin is no longer representative of 6 kHz if the window length is not equal to 120 samples. Changing the detection window length changes both the *N* and *n* term in the equation, and thus each new window length will require a new 1xN lookup table. A fully calibratable window length would use much memory in table storage.

The next section will propose a method for varying the knock detection window length while using the single point DFT detection method. The section will propose methods for limiting table storage while slightly increasing calculations.

VARIABLE WINDOW LENGTH PROPOSAL

Our goal is to propose a method for knock detection that allows the varying of the window length without varying the values of *n* and *N*. One way to do this is to make the detection window a combination of two smaller overlapping sub-windows. The two sub-windows would be of equal length; the detection window length would be modified by varying the overlap of the two smaller sub-windows.

Revisiting our example in the previous section, our full detection window was 3 ms (120 samples.) If we wish to have two smaller windows with 3 ms of coverage, we could use two windows that are 1.5 ms (60 samples) in length. This changes the bin size. The bin size is now twice the size that it was before. The increase in bin size is an added benefit of using the smaller amount of samples in the sub windows. Often times, especially when calculating the full DFT, it is desirable to use more samples to decrease the bin size, and thus increase the resolution across the effective frequency range. But, since we are only monitoring one frequency (one bin), it is desirable for us to use a larger bin. Because of the increase in bin size, the bin number for 6 kHz is now bin 9.

$$\frac{6kHz}{667Hz/bin} = 9^{th}\,bin$$

(12)

The updated DFT equation with the new bin number and number of samples is listed below.

$$X(e^{j\omega}) = \frac{1}{60}\sum_{k=0}^{59} x(k)(\cos(2\pi\frac{9}{60}k) - j\sin(2\pi\frac{9}{60}k))$$

(13)

The cosine terms for k = 0 to 59 will still be calculated ahead of time and stored in a 1x60 lookup table, which is half the size as our previous example. The sine terms can still be acquired from the same table as the cosine terms, but the sine terms are shifted by 90 degrees.

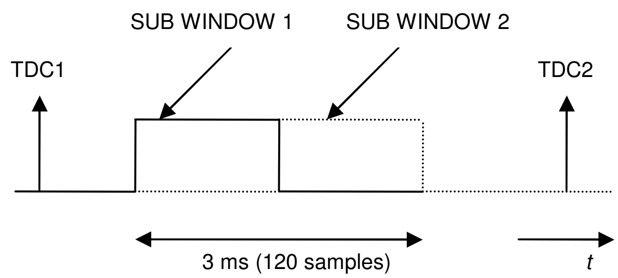

Figure 2

With no overlap, the detection window remains 3 ms (the sub windows are 1.5 ms). Equation 13 must be calculated for the first sub-window and the second sub-window. Magnitudes for each sub-window would be stored and maintained separately.

If we wish to use a 2.6 ms window, the sub windows would need to over lap. The overlap would be 16 samples.

(3 ms – 2.6 ms)(40000samp/sec) = 16 samp

(14)

The total length of the window in samples is now 104 samples.

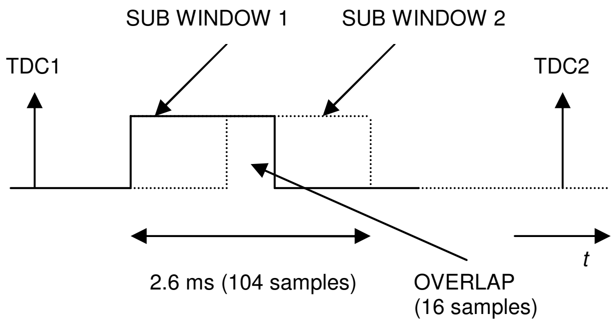

Figure 3

Examining Figure 3, we see that the sub windows now overlap by 16 samples. The samples in the overlapping section are included in the DFT calculations for both sub windows. The calculation of sub window 1 includes window samples 1 to 60. The calculation of sub window 2 includes window samples 45 to 104.

ALGORITHM PROPOSALS

A recommended knock detection algorithm for the single window, single point DFT is to calculate the magnitude at the desired frequency, which, in the case of our example, is 6 kHz. A knock event would be determined by comparing the magnitude calculation of the current cylinder event to the most recent non-knocking magnitude calculation plus a threshold offset.

One simple method for dealing with the separate sub window DFT calculations is to average the magnitudes of the DFT point for both sub-windows in the knock detection algorithm. Again, a knock event would be determined by comparing the present average magnitude calculation to the most recent non-knocking average magnitude value plus a threshold offset.

Once knock is detected, a control algorithm would suppress the knock (usually by retarding spark.) The actual control algorithm is beyond the scope of this paper. However, it is recommended that weighting techniques be examined as a varying percentage of the knock signal is present in both sub-windows.

The next section of this paper contains windowed cylinder events from actual engine data for both knocking and non-knocking conditions. The data was post processed using MATLAB to calculate the Discrete Fourier Transform using both the traditional single window technique as well as the proposed overlapping window technique. The DFT magnitude is provided at 6 kHz for all plots for comparison purposes.

ENGINE DATA

The following plots were recorded on an engine dynamometer operating at 1600 RPM with a load of 90 kPa. The knock detection window was calibrated to be active 20 degrees after top dead center with a length of 3 milliseconds at a sample rate of 40 kHz. Figure 4 shows a windowed knock event with the DFT magnitude calculated at 6 kHz (bin 19.)

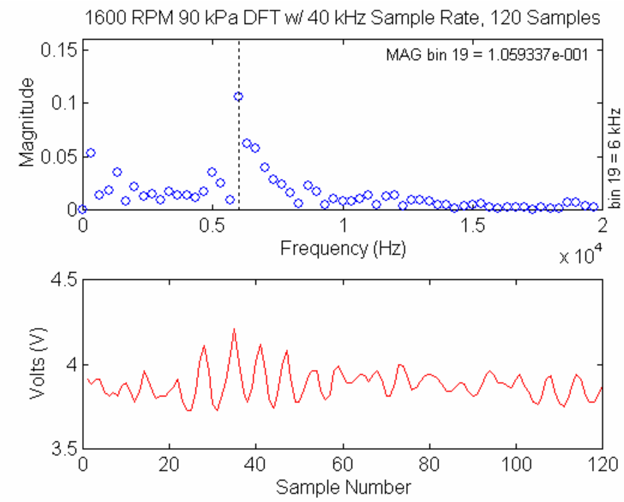

Figure 4

Figure 5 shows a non-knocking cylinder event with the DFT magnitude calculated at 6 kHz (bin 19.) Note the difference in the magnitude calculation at 6 kHz for both plots. Algorithmically, it should be very easy to differentiate the knock event from the non-knocking event.

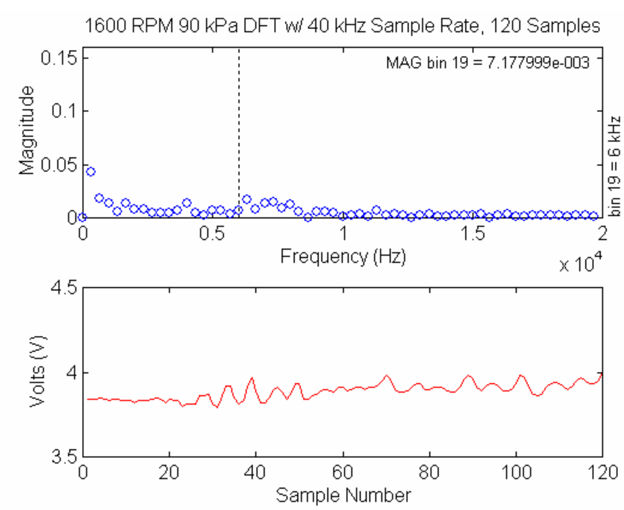

Figure 5

Figure 6 shows the same windowed knock event using overlapping sub-windows of 80 samples. The overlapping region is 40 samples. The average magnitude is calculated at 6 kHz, which is now bin 13.

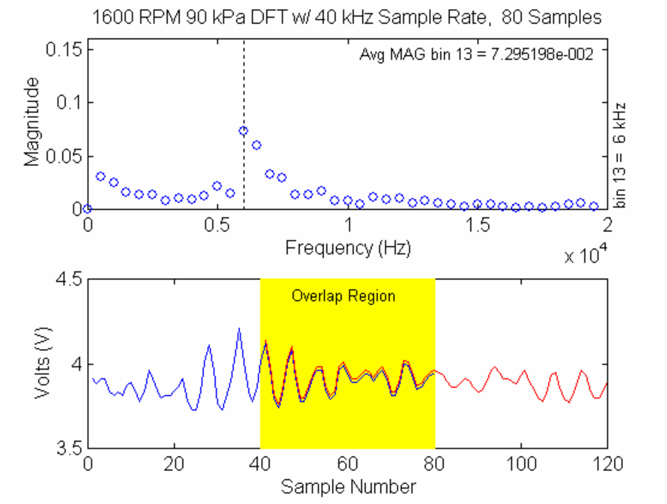

Figure 6

Figure 7 below shows the same non-knocking cylinder event processed with the overlapping sub-window method. The average magnitude is again calculated at 6 kHz (bin 13.)

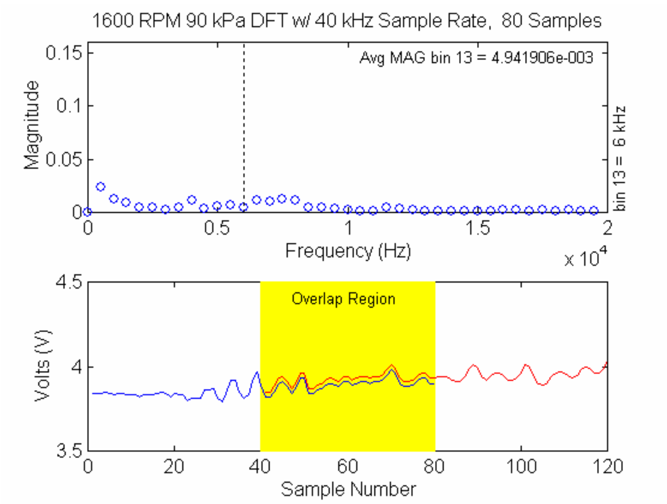

Figure 7

Again, note the difference in the magnitude calculation at 6 kHz for both plots calculated using the overlapping window method. The difference in the magnitude between the knocking and non-knocking data is comparable to that of its single window counterpart. The data shows that it is possible to differentiate the magnitude of a knock event from a non-knocking event at 6 kHz using either the traditional single window method or the overlapping window method.

ADDITIONAL CONCERNS

There are some additional concerns that must be addressed if the overlapping window method is to be applied in a practical application. It should be noted that if we increase the overlap, and thus reduce the total coverage of the window as we increase in RPM, we are reducing the total number of samples that will be analyzed. This increases the variance in the estimate of the knock magnitude.

Because the overlap causes varying number of samples to be included in the DFT calculations of both windows, it is advisable to not radically change the window length over small changes in RPM. Some issues regarding RPM could be reduced by adding hysteresis to the window length calibration versus RPM.

It may also be more practical to raise the base window calibration increment from 1 sample. With our example, the smallest the window can increment or decrement would be .025 ms. For most applications, a knock window would probably not need .025 ms of resolution. If we assume that we need .1 ms of resolution, we can see that that the calibration changes to the window will cause it to shift in blocks of 4 samples.

Also, the DFT method adds error with each window edge. Since the DFT calculation assumes that the knock signal in the window repeats continuously, each window rise and fall adds a “sharp edge” that can cause abnormal and unrealistic frequency content in the calculation of the DFT. With the overlapping window scenario, we have discontinuities at the open and close of both sub-windows. Tapering the knock signal with an apodization function is a method to reduce the effects of these discontinuities. An apodization function reduces the effects of the discontinuities by smoothly rather than abruptly reducing the edges [7]. The apodization function will reduce the resolution of the result.

The apodization function would be calculated ahead of time and stored in software. To save memory, the apodization function can be combined with the cosine table that is already stored for the single point DFT calculation.

CONCLUSION

There are already knock detection and control systems in practice today that make use of the Discrete Fourier Transform, but when conditioning the knock signal many limit the detection window to one length for all operating points. This paper has proposed a way to add variable window length to a knock system that employs the use of the DFT with no increase in memory storage and a minimum increase in processor calculations.

ACKNOWLEDGMENTS

I would like to thank Robert Koseluk for your years of mentoring. I appreciated your time and patience.

REFERENCES

1. omni.cc.purdue.edu/~tlrobert/Combustion%20and%20knock
2. Vehicle Engine Technology / Heinz Heisler – 2nd Edition
3. Automotive Engines : theory and servicing / Herbert E. Ellinger, James D. Haldermann – 2nd ed.
4. The Internal-Combustion Engine in Theory and Practice Volume II: Combustion, Fuels, Materials, Design / Charles Fayette Taylor
5. Automotive Handbook, Bosch 3rd ed / Robert Bosch GmbH, 1993
6. Signals and Systems / Alan V. Oppenheim, Alan S. Willsky, with S. Hamid Nawab – 2nd ed.
7. mathworld.wolfram.com/ApodizationFunction.html

CONTACT

Matthew (Matt) J. Kearney
Knock Advanced Development Group
Delphi E & S
matthew.j.kearney@delphi.com

Mr. Kearney has a BSEE from Purdue University. He has spent much of the last seven years traveling the country providing on-site calibration development for customers of Delphi's knock detection and control systems. He currently works to develop new knock detection strategies for Delphi's Knock Advanced Development group.