# Pseudo Random PWM Techniques Improve Motor Drive Performance

**Nathan John, Strategic Marketing Director**
**Greg Verge, Strategic Marketing Engineer**

**Cypress MicroSystems, Inc.**
**22027 17th Avenue S.E. Bothell, WA 98021**

**Telephone: 877.751.6100**
**Fax: 425.939.0999**
**E-Mail: nwj@cypress.com, gjv@cypress.com**
**Web Address: http://www.cypressmicro.com**

*Pulse width modulation (PWM) waveforms are widely used to implement variable-speed motor drives, especially those based on 8-bit microcontrollers. Pseudo random pulse width modulation (PRPWM) techniques offer significant benefits over deterministic PWMs when implementing motor drive systems. Traditional PWM motor drives produce significant amounts of both audible and electromagnetic noise with high harmonic content during operation.*

*Humans respond negatively to sounds that have high tonal content. Spreading out the noise energy will result in a perceived noise reduction. These techniques will also reduce peaks in the electromagnetic (EM) emissions that occur at the same harmonics, with the potential of easing compliance to regulatory standards*

*A demonstration motor drive system for a brushed DC motor will be presented. This system can operate in two modes utilizing deterministic PWM techniques and pseudo random PWM techniques. The system was tested while operating in both modes, and the resulting analysis shows the benefits of spreading harmonics under typical operating conditions.*

## INTRODUCTION

Pulse Width Modulation (PWM) waveforms are widely used in adjustable speed drives for AC and DC motors. By varying the ratio between high and low states, known as the duty cycle, the user can vary the average voltage applied to the motor, and hence control its speed. The simplicity of this technique, coupled with the wide availability of PWM peripherals on microcontrollers, has made this the standard for implementing adjustable speed drives.

However, there is a drawback to implementing drives in this manner. The switching frequency of the PWM will introduce a high harmonic content into the motor current, resulting in emission of both acoustic and electromagnetic noise. The harmonic structure of this noise, which is a series of tonal ‘combs’, is seen as a set of energy spikes at the fundamental switching frequency of the PWM, and it’s higher order harmonics.

In some end applications containing motors, this noise, and its effects on the people who are within hearing distance, may be an important design criteria. An additional design constraint may result from the regulation of emissions of electromagnetic energy produced by the drive. One straightforward method to mitigate these problems is to increase the switching frequency of the PWM until it is in the ultrasonic range, which will eliminate the effects on humans. However, there are three significant drawbacks to this solution. The first drawback is the power loss that results from the switching of the power transistors, which increases linearly with the switching frequency. This reduces the energy efficiency of the system. A second drawback is the scarcity of high power transistors with ultrasonic frequency switching capability[1]. Lastly, increasing the switching frequency will not decrease EMI energy peaks, which present a large hurdle to gaining regulatory approval.

Experiments conducted by Kryter[2] demonstrated that subjects exposed to sound with narrow energy bands perceive this sound to be more annoying than broadband sound, even if the total energy level in both cases was the same. This is especially true for the frequencies between 1-6KHz, which are optimal for many power transistors but particularly bothersome for humans. We can thus divide the problem of noise, and its effects on people into two components, these being the total sound energy, and the tonal content of this sound.

Changing the tonal content of the noise may represent an approach to solving a noise problem in a product where the reduction of overall sound energy is not an option. The goal of this technique is not to reduce the amount of noise produced by the motor, but rather change the tonal content so that it is less offensive to people, as well as broaden the spectrum of EMI energy.

Adding a pseudo-random component to the PWM signal will break up the narrow band tonal energy, and spread it out over the entire frequency range. There are many common techniques for implementing Pseudo Random Pulse Width Modulators (PRPWMs). This paper presents a simple implementation of a variable speed drive based on PRPWM principles. This implementation will function in either a traditional PWM manner, or as a PRPWM, allowing for the side-by-side performance evaluation of the two techniques regarding noise generation in both audible and EMI form.

## DESCRIPTION OF SYSTEM

### ***Switching Technique Options***

Trzynadlowski et al.[3] discussed three basic strategies for creating pseudo random waveforms for use in motor drives. These strategies are shown in Figure 1, including a deterministic PWM shown in Figure 1A. Figure 2 compares the benefits of the these four different types of PWMs:

1. **Randomized Switching Frequency**

   There are two methods to implement random switching frequency PRPWMs. The first method is to switch between different periods with a traditional PWM. This results in transitions of switching frequency that always occur at the beginning of the PWM cycle, and therefore cause no disruption in the duty cycle. In this method, the period lengths should be switched on a random basis in each output cycle. This is illustrated in Figure 1B.

   Another method of implementing random switching intervals is to use a triangular signal of randomly varying slope, and compare this to a reference that determines the duty cycle. The variation in the slope results in variations in the length of individual cycles, while the duty cycle remains constant.

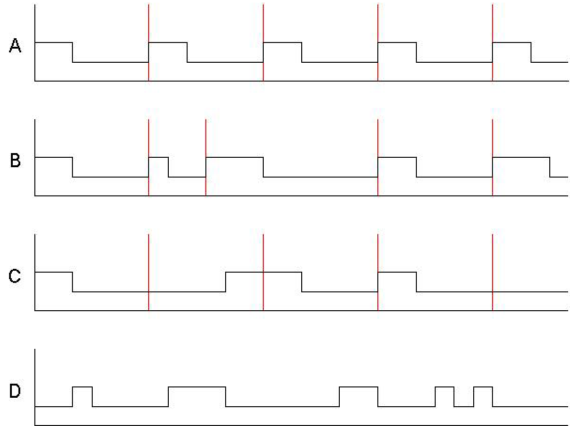
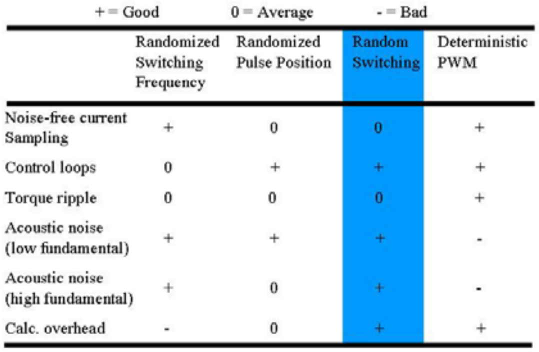


Figure 1

2. **Randomized Pulse Position**
In this approach, the length of each cycle is constant. The length of the high time and the sum of the lengths of the low times are also constant. What varies on a random basis is the position of the high pulse in respect to the beginning of any cycle. This can result in either one or two low times during any two cycles. However, because their length remains constant when summed together, this insures a constant duty cycle. In its simplest form, the high pulse is varied to either come at the beginning of the cycle, or at the end. This is known as a “Lead-Lag” PRPWM and is illustrated in Figure 1C.



+ = Good 0 = Average - = Bad

| | Randomized Switching Frequency | Randomized Pulse Position | Random Switching | Deterministic PWM |
|---|---|---|---|---|
| Noise-free current Sampling | + | 0 | 0 | + |
| Control loops | 0 | + | + | + |
| Torque ripple | 0 | 0 | 0 | + |
| Acoustic noise (low fundamental) | + | + | + | - |
| Acoustic noise (high fundamental) | + | 0 | + | - |
| Calc. overhead | - | 0 | + | + |

Figure 2

3. **Random Switching**
To achieve random switching, both the high and low times are varied. Because this effect is derived by some deterministic technique, there will be a period in which the pattern of high and low times will repeat. If the ratio of this period relative to the shortest switching period is very high, this will spread any harmonics resulting from this period over a broad range of frequencies. This is illustrated in Figure 1D.

## SYSTEM ARCHITECTURE

The method selected for this experiment is a variation of the random switching technique. The particular implementation chosen utilizes a feedback shift register and a digital comparison to the selected duty cycle as discussed by Hui[4]. This shift register is clocked at the desired maximum switching frequency, and will output a pseudo random bit stream with the desired average duty cycle.

Two examples of the feedback shift registers used are shown in Figure 3, an 8-bit shift register (3B) and a 16-bit shift register (3C). In each case, the input to the selected shift register taps is the output of the most significant bit of the shift register. The shift registers in Figure 3B and 3C are the modular PRS form used by PSoC and correspond to the more common, “simple form” in Figure 3A. These taps are selected specifically for a desired cycle of patterns shifting through the register. The desired patterns for a feedback shift register of length n will be a cycle through $2^n$-1 states, with each state in the cycle being a unique n-bit pattern of zero’s and one’s. For each value of n, there is a unique set of tap values that produce an output pattern that meets this criterion.

The one excluded state is all zero’s, which would result in the shift register being “stuck” in that state forever. The initial state of the shift register does not matter - as long as at least one ‘1’ is present, it will cycle through all the possible states.

Figure 3

The native output of the feedback shift register is a pseudo random pattern of $2^n$-1 bits, of which $2^{n-1}$ will be zeros, and $2^{n-1}$-1, which will be ones. For large values of n, this will closely approximate a 50 percent probability for either output. This output must be modified, because there is no mechanism to vary the duty cycle in this sequence. The probability that the n-bit value inside the shift register is less than an n-bit duty cycle value is equal to the duty cycle itself. Making duty cycle comparisons on a periodic basis will result in a stream of bits where the proportion of one’s will be ratio-metric to the duty cycle, and the stream will also be pseudo random.

This implementation was picked for its simplicity, and level of hardware support available on the host device. This host offers hardware to implement either an eight bit or a sixteen bit feedback shift register, and allows the user to select tap values to match the required output characteristics. Generation of the pseudo random values is therefore done completely in hardware. The only requirement of the software is to make the comparison with the duty cycle register.

## HARDWARE DESIGN

### ***Motor and Driver Description***

The motor chosen for this implementation is a standard direct current (DC) brushed gear motor, as found in many consumer and industrial appliances. Sign magnitude H-bridge control was utilized although locked antiphase is equally applicable to this modulation technique.

## ***Controller Overview***

The device used to implement this system features an array of digital circuits and an array of analog circuits, which can be used to create a wide variety of mixed signal functions. This device also includes an 8-bit CPU core, flash memory and SRAM, so that it can perform all the necessary digital and analog functions required by a variable speed motor drive. Figure 4 illustrates this system in demonstration board form.

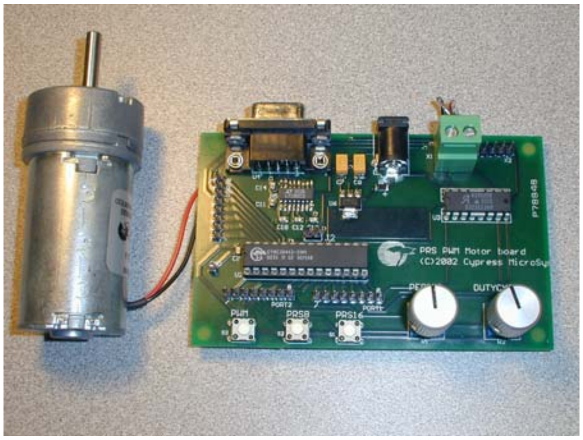
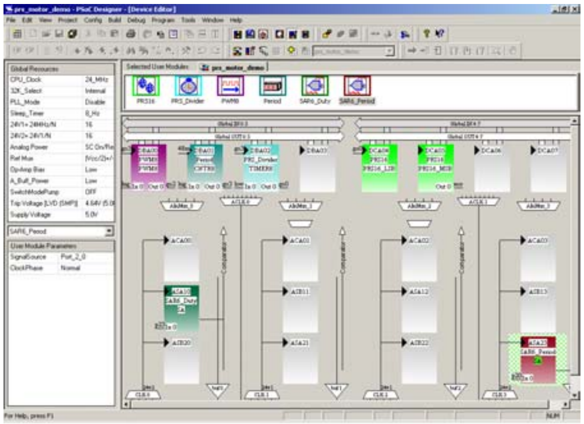


Figure 4

The digital resources on this mixed signal device are arranged into logical blocks. These blocks can take on a variety of 8-bit functions. For example, a digital block can be configured into an 8-bit deterministic PWM. This same digital block can also be configured to be a pseudo random sequence generator (PRS) with user selectable tap values. After selecting the correct tap values, the software needs only to make a comparison between the 8-bit value inside the PRS and the duty cycle to convert this PRS into a PRPWM. The resource configuration created inside the development tool for this device is shown in Figure 5. The logical blocks available in this family of products are shown in gray, and those blocks that have been assigned a function such as 8-bit PRS are shown in a darker shade.



| Global Resources | |
|---|---|
| CPU_Clock | 24_MHz |
| 32K_Select | Internal |
| PLL_Mode | Disable |
| Sleep_Timer | 8_Hz |
| 24V1= 24MHz/N | 16 |
| 24V2= 24V1/N | 16 |
| Analog Power | SC On/Re |
| Ref Mux | (Vcc/2)+/ |
| Op-Amp Bias | Low |
| A_Buff_Power | Low |
| SwitchModePump | OFF |
| Trip Voltage [LVD (SMP)] | 4.64V (5.0 |
| Supply Voltage | 5.0V |

SAR6_Period

| User Module Parameters | |
|---|---|
| SignalSource | Port_2_0 |
| ClockPhase | Normal |

Figure 5

The digital blocks can also be chained together. For example, two digital blocks can be used together to implement a 16-bit PWM or a 16-bit PRS. The three basic modulation methods that are used in the demonstration project are 1) 8-bit PWM, 2) 8-bit PRPWM and, 3) 16-bit PRPWM.

These three methods are all implemented in the hardware and software of the demonstration board as shown in the Figure 6 block diagram. The board includes three buttons to select between these modes, and two potentiometers to adjust the duty cycle and switching frequency of the drive.

## SOFTWARE DESIGN

The software design for this test project is very simple. After initialization, the software remains in an infinite loop. The loop continuously sequences between the two tasks of making the software comparison for the PRPWM modulation, and checking for control updates of the modulation technique, desired duty cycle and frequency. The software flow chart is shown in Figure 7.

The software uses two hardware resources to perform the correct modulation scheme. One of the digital hardware blocks is configured to be an 8-bit deterministic PWM that performs this type of modulation when selected. A programmable digital interconnect path carries the modulated signal from the PWM to the motor driver I/O pin. A multiplexor inside the I/O structure chooses between this signal, and the data output register that is used when the system is switched to the PRPWM techniques.

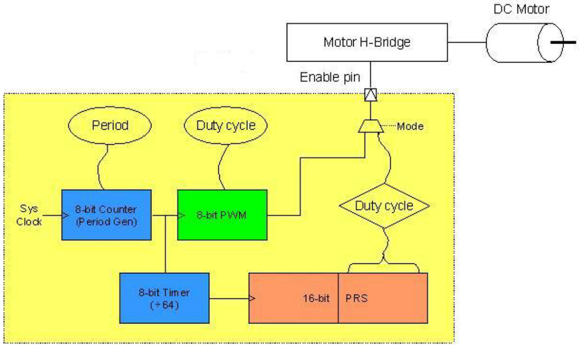


Figure 6

There are two additional digital blocks that implement either the 8-bit PRPWM or the 16-bit PRPWM, depending on the feedback tap values selected. Changing the tap values, (which can be done dynamically) will switch the cycling pattern in the lower byte of the shift register. When the tap values are set for a 16-bit pattern, the entire 16-bit register will cycle through $256^2$-1 states before repeating. The PRPWM must therefore be clocked 65,535 times before the pattern produced by the low byte repeats. The upper trace in Figure 8 displays an 8-bit PRPWM that repeats with the lower trace showing the period boundaries. In each case, the software will read the value from the lower register byte, compare it to the stored duty cycle value, and write out a one or a zero to the motor driver data register bit.

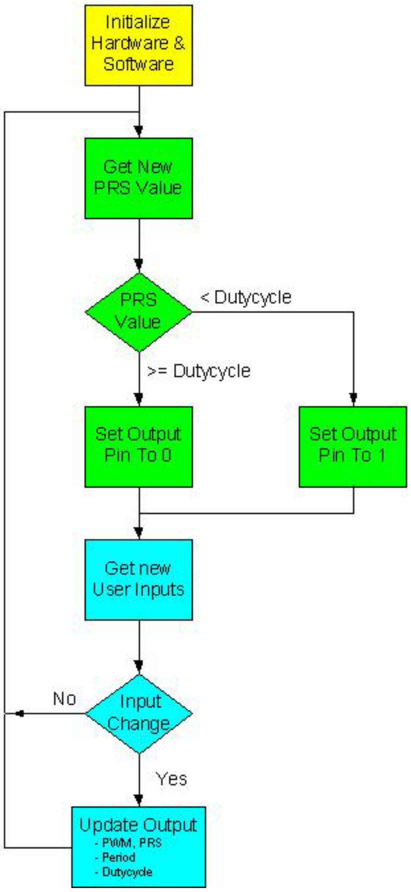


Figure 7

Both the 8-bit PWM, and the two block PRPWM (which can take the form of an 8- or 16-bit PRPWM based on the taps) are free-running, and the only software intervention in the steady state is the PRPWM comparison mentioned in the previous paragraph.

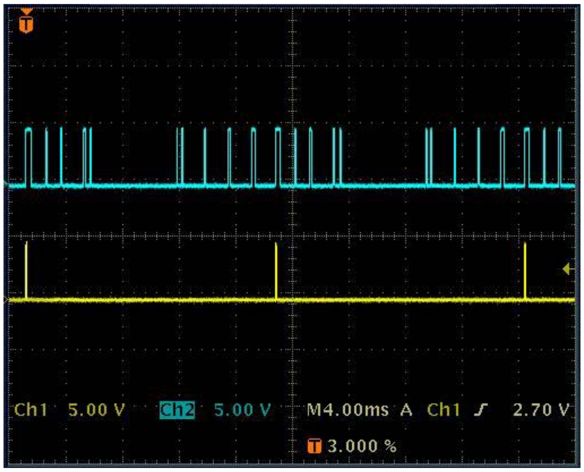


Figure 8

The software also checks for control updates of the modulation technique, desired duty cycle and desired frequency. It updates the controls whenever there is a change detected. Changing the modulation scheme entails changing the mux at the I/O pin to select between PWM and PRPWM modes and possibly the tap values. Selecting the duty cycle on the PRPWM simply requires a write to the duty cycle register. Changing the frequency in any of these modes requires the user to change the pre-scaler in the clock generator, which feeds both the PWM and PRPWM hardware.

## TESTING

The focus of testing was to determine the benefits of PRPWM versus deterministic PWM techniques, based upon both audible noise as well as EMI. All testing was performed under identical conditions that included background noise sources typical of real world environments. Background noise is primarily evident in the base levels of audible noise during motor testing. It is important to note that the acoustic spectrum measured is meant to be a relative comparison only, and is not an absolute measurement. Comparisons were made between an 8-bit deterministic PWM, 16-bit PRPWM and an 8-bit PRPWM while running at a 40 percent duty cycle with a 1.4 MHz input clock. Figure 9 displays both the PWM (top) and PRPWM (bottom) used to generate the line voltage and acoustic noise FFT spectra for each modulation technique.

The deterministic PWM provided a 5.4 kHz output clock to the motor driver. Figure 10 shows significant peaking of the line voltage at the primary switching frequency as well as the odd harmonics. The acoustic noise in electric motors is predominantly proportional to the square of the air-gap-flux density with PWM switching[5], creating the audible tones evident in Figure 11. The tones of the motor were subjectively determined by observers to be very annoying.

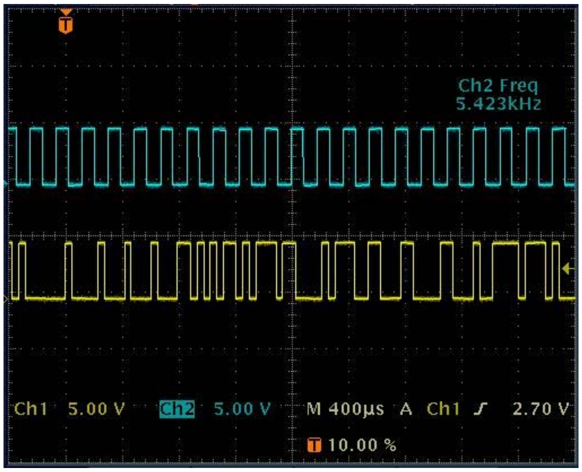


Figure 9

To allow a consistent comparison point, the 8- and 16-bit PRPRS implementations divide the 1.4 MHz clock by 64, resulting in the same average maximum number of transitions as the PWM in a given period. In an PRPWM, the maximum density of switching events occurs at a 50 percent duty cycle with the switch rate and associated losses reducing at both higher and lower duty cycle values as shown in Figure 12. The only significant difference between 8-bit and 16-bit PRPWMs is the number of cycles that take place before the pattern repeats. At high clock rates the 8-bit PRPWM generates cycles that can have an audibly noticeable pattern. The 16-bit PRPWM alleviates this by increasing the cycle length by a factor of 257. The subtle deference results in almost identical spectra by both PRPWM widths. Figure 13 shows that the switching energy of the line voltage has been spread across the spectrum, while Figure 14 shows the reduction of tonal components in the audible range. The PRPWM motor drive produces a white noise emission that is difficult to differentiate from normal motor operational noise.

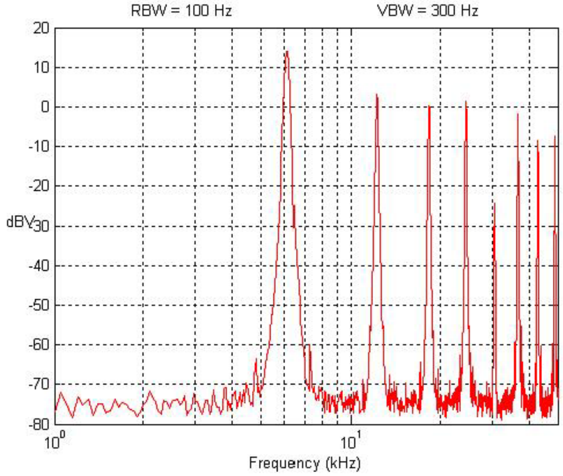


Figure 10

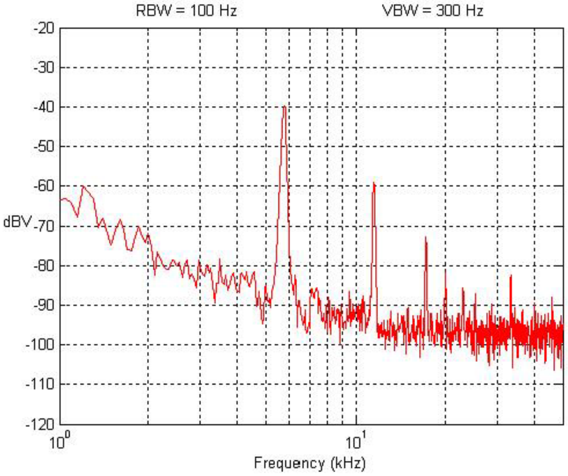


Figure 11

From these test results, we have shown that switching energy has been spread resulting in a 15 dB reduction at peak frequencies. This reduction has removed the offensive tonal components emanating from the motor, as seen in the 15 dB reduction in peak sound intensity and the more uniform spreading of noise over the audible range. The total sound energy has not been reduced but spread to create a more pleasant sound. We can see that the spreading of noise has dramatic benefits for motor driven applications and can be accomplished with relatively simple modifications to traditional methods.

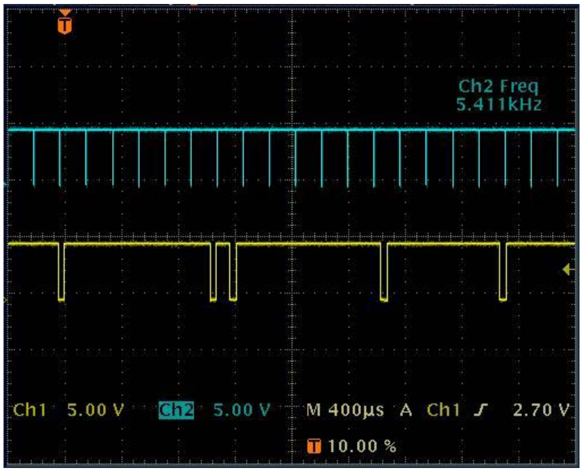


Figure 12

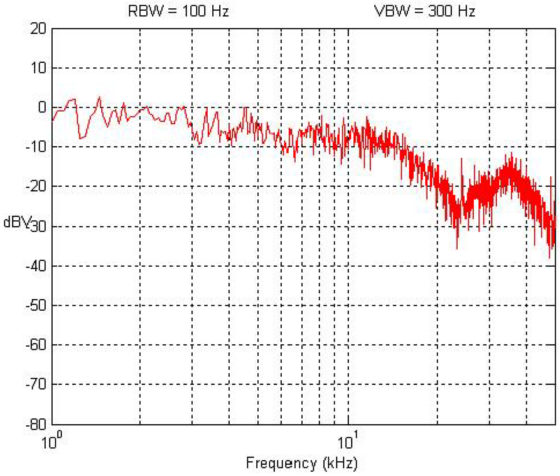


Figure 13

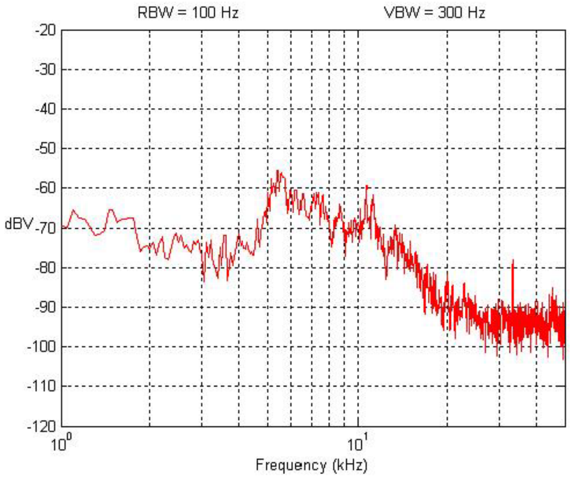


Figure 14

## EXTENSIBILITY TO OTHER MOTOR TYPES

While the data presented here shows a clear advantage for fractional horsepower-brushed DC motors, similar improvements for other topologies are expected based on the experiments by Boys and Handley[1], Trzynadlowski et al.[3] and Handley et al.[6]. Brushless DC and stepper motors can benefit from RPWM by reducing EMI and audible effects of switching in either their basic mode of operation, or when using chopper drives to control current.

The most significant improvements can be expected in variable frequency AC motor drives. Although there are several AC motor topologies, the most common used in appliances have their speed controlled by varying the AC frequency. AC motors are most often used at larger power levels, where tonal components of sufficient amplitude to cause discomfort and temporary hearing loss have been documented by Wallace et al.[7]. While the switching frequency is most easily increased into the ultrasonic range, high power motors often require IGBT-based inverters in which the effective switching speed is limited to the audible range. Figure 15 depicts the sinusoid to be approximated with the random values generated, the PRPWM inputs to the three inverter phases (a, b, c), and the voltage across each phase winding (Va, Vb, Vc).

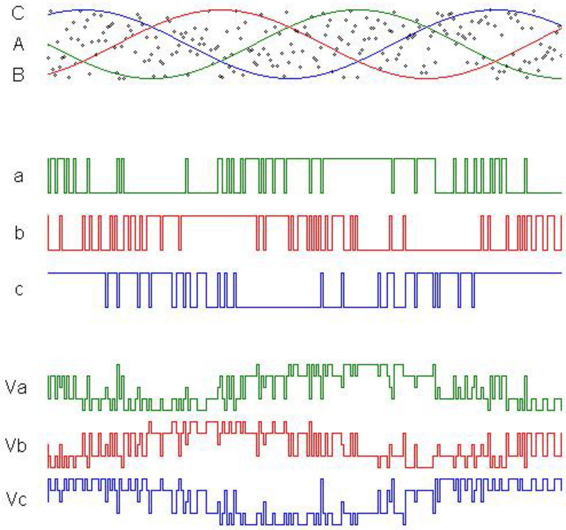


Figure 15

## FUTURE ENHANCEMENTS TO HARDWARE PLATFORM

One limitation in the current implementation presented is the software overhead associated with the comparison between the value stored in the shift register and the desired duty cycle. While the overhead is small for a system with one variable speed drive, the overhead will increase as the number of motors or phases controlled increases. This is especially true if the frequencies for the PWMs are not the same, or if there is some reason in the system that each comparison cannot take place for each PWM in succession, which will add further software overhead.

A solution to this problem is planned for the next generation of this family of devices. By implementing the comparison in hardware inside each logical block, the processor is relieved from any software overhead, with the exception of changing the duty cycle on a particular channel, similar to a standard PWM. This planned family of devices will support eight independent 8-bit Pseudo Random PWMs, each of which is capable of driving a variable speed motor.

## CONCLUSION

Pulse width modulation techniques are widely used in variable speed motor drives of all types, and these techniques have many desirable characteristics. One undesirable characteristic is the significant tonal content resulting from the primary switching frequency of the PWM. An effective and low cost solution to mitigate this issue has been demonstrated utilizing a pseudo random pulse width modulation technique. For the designer and user of motor drives, this represents another tool to improve the performance of your products.

---

[1] Boys, J.T, and Handley, P.G., 'Spread spectrum switching: low noise modulation technique for PWM inverter drives', IEEE Proceedings-B,1992, vol. 139, no. 3, [253-254]

[2] Kryter, K.D.: 'Concepts in perceived noisiness, their implementation and application'. Journal of the Acoustic Society of America, 1985, no.2, [344-361]

[3] Trzynadlowski, A.M., Blaabjerg, F., Pedersen, J.K., Kirlin, R.L., Legowski, S. 'Random Pulse Width Modulation Techniques for Converter-Fed Drive Systems – A Review', IEEE Transactions on Industry Applications, vol.30, no. 6, [1166 –1167]

[4] Hui, S.Y.R., IEEE Transactions on Power Electronics, vol. 12, no. 3, [253 – 258]

[5] Habetler, T.G., Divan, D.M., 'Acoustic noise reduction in sinusoidal PWM drives using a randomly modulated carrier', IEEE Transactions on Power Electronics, vol. 6, no.3, [356 – 363]

[6] Handley, P.G., Johnson, M., Boys, J.T., 'Elimination of Tonal Acoustic Noise in Chopper Controlled DC Drives', Applied Acoustics, no 32, [107 – 119]

[7] Wallace, A.K., Spee, R., Martin, L.G., 'Current Harmonics and Acoustic Noise in AC Adjustable-Speed Drives', IEEE Transactions on Industry Applications, vol. 26, no. 2, [267 – 273]