

AN47778

Author: Raj Kumar Singh Parihar, Anshul Gulati

Associated Project: Yes

Associated Part Family: CY8CLED04, CY8CLED08, CY8CLED16

[GET FREE SAMPLES HERE](#)

Software Version: PSoC Designer™ 4.4 or PSoC Express™ 3.0

Associated Application Notes: No

Application Note Abstract

This application note presents an overview of Delta Sigma Modulation (DSM) used in LED dimming applications. It also describes variants of the existing DSM technique. Variable DSM, an improvement on basic DSM, is also discussed extensively in this application note.

Introduction

Delta Sigma Modulation (DSM) is one of the several techniques used to dim LEDs. This application note focuses on DSM and its variants. Any modulation technique that maintains average duty cycle or desired signal density in a specified time period can be used to dim LEDs. These modulation schemes are popularly known as Pulse Density Modulation (PDM).

Unlike Pulse Width Modulation (PWM), where the pulse width is variable, in DSM the time duration between two consecutive pulses is variable. This is also interpreted as variable frequency or modulated time period scheme. In DSM, when duty cycle or signal density is close to maximum, the pulses of fixed widths appear close to each other. This increases the signal density. Similarly, for low dimming values the pulses are far apart from each other, thus lowering the average signal density.

Assumptions

This document assumes that the reader is familiar with LED dimming concepts. Prior knowledge and understanding of PSoC® architecture, C programming language, and switching regulator design is also useful.

EZ Color: Overview

Cypress' EZ-Color™ family of devices offers an ideal control solution for HB LED applications requiring intelligent dimming control. EZ-Color devices combine the power and flexibility of PSoC (Programmable System-on-Chip™) with Cypress' PrISM™ modulation technology to provide lighting designers a fully customizable and integrated lighting solution platform.

PSoC Express™ software, with lighting specific drivers, significantly cuts development time and simplifies implementation of fixed color points through temperature and LED binning compensation. EZ-Color's virtually limitless analog and digital customization allows simple integration of features in addition to intelligent lighting, such as battery charging, image stabilization, and motor control during the development process.

For more information about EZ-Color solutions and associated technologies, visit to <http://www.cypress.com/ez-color/>

Delta Sigma Modulation

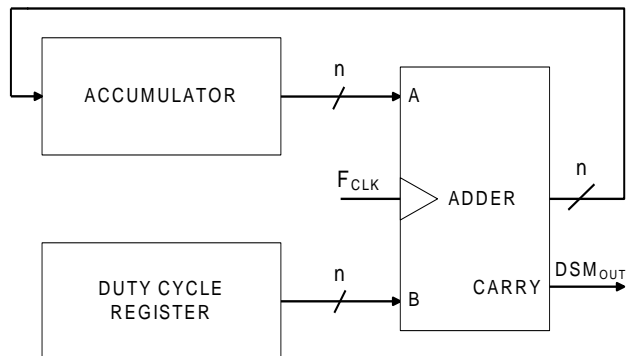
DSM is one among several known modulation techniques employed in LED dimming applications. The average signal density is directly proportional to input voltage or dimming value over a period of time. This feature is useful in brightness control.

In typical Delta Sigma implementation, the interval between pulses is determined by the feedback loop. A low input voltage produces a long interval between pulses and a high input voltage produces a short interval between pulses. Interval between pulses is proportional to the inverse of the mean of input voltage.

DSM Operation

This section describes how to generate the pulses of fixed width (DELTA) at desired intervals. Simple DSM hardware consists of an adder and an accumulator as shown in Figure 1. The duty cycle register is loaded with the desired dimming value and the accumulator is initialized with zero value. On each tick of F_{CLK} , duty cycle value is added to the content of accumulator and the sum is again restored in the accumulator. Carry is the DSM output used to control the brightness of LEDs.

Figure 1. Block Diagram of n-bit Delta Sigma Modulator



An example to generate the equivalent DSM for 50% dimming level is described. In 8-bit DSM, the duty cycle register for 50% dimming level is loaded with 128. On first tick of F_{CLK} , the sum is 128, but the carry is still '0'. On the next tick, sum is '0' after rolling back from the maximum value and carry is generated. This way every odd tick generates the carry, which has an equivalent signal density of 50%.

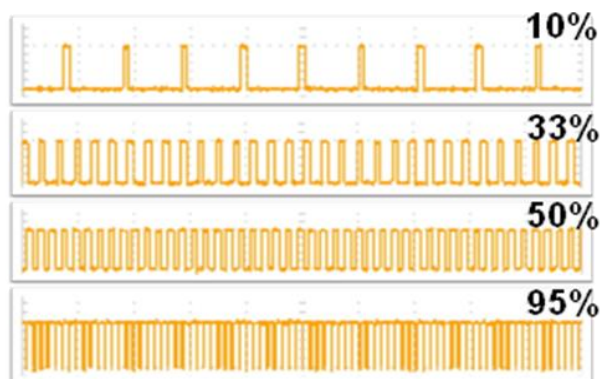
Similarly, for a high dimming value close to 255 there are good chances that carry is always generated after subsequent summation and thus equivalent signal density is close to 100%, if not exact.

This is also true for low dimming values below 5%. In these cases, it takes a longer duration to produce a carry because many subsequent additions are required.

DSM Waveform

Figure 2 shows the sample waveforms for 10%, 33%, 50%, and 95% signal density.

Figure 2. Waveform of DSM for Various Signal Densities



Spectral Plot

FFT plots of DSM wave for 50% and 85% duty cycle are depicted in Figure 3 and Figure 4. Figure 2 indicates that DSM increases the number of transitions in a fixed time period. It maximizes the transitions to the largest possible value. Due to maximum number of switching in DSM, the EMI is worse than equivalent PWM and PrISM.

Figure 3. FFT Plots of DSM for 50% Signal Density

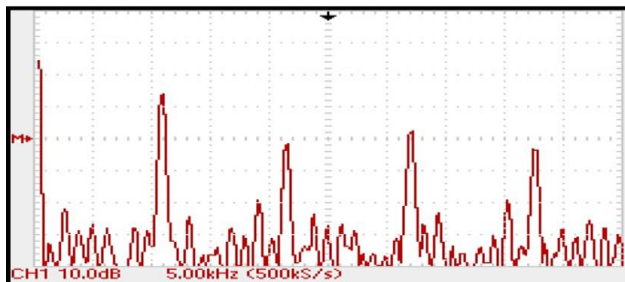
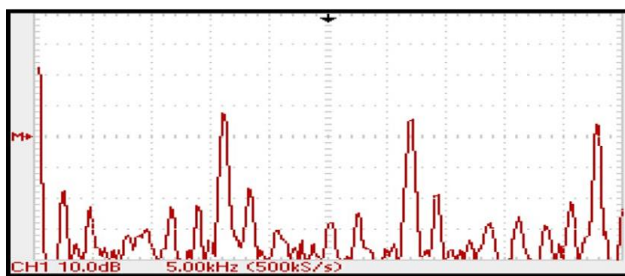


Figure 4. FFT Plots of DSM for 85% Signal Density

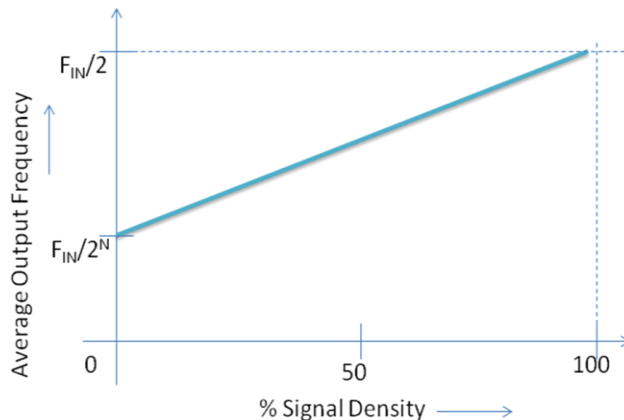


However, for the same clock frequency, the harmonics are pushed farther out making it easier to filter them. The concern is that filtering leads to loss in average signal density. This is not desired in precise brightness control and color mixing applications.

Output Frequency Plot

DSM increases the number of transitions to the largest possible value. Maximum switching occurs at one of the extreme signal density just before 100% dimming value.

Figure 5. Average Output Frequency Plot of DSM



Implementation with EZ-Color

There are two possible ways to implement DSM in EZ-Color.

- In software.** In this implementation, perform the addition using CPU. The accumulation and addition are done in the ISR of timer. When a timer interrupt occurs, perform the addition and accumulate the dimming values. Output is toggled depending on the carry generation.

Code 1. Firmware Template

```

//Global Variables for Delta Sigma Modulation
char DeltaSigmaAccum = 0;
char SgnlDensity;

//Timer ISR
_CLK_DSM_ISR:
    mov A, [_DeltaSigmaAccum];
    add A, [_SgnlDensity];
    mov [_DeltaSigmaAccum], A;

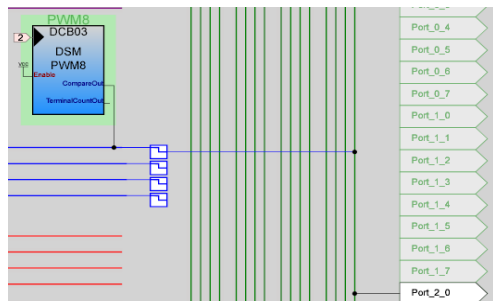
    jc Skip
    //Reset PORT_1_0 if carry is not generated
    and PRT1DR, 0xFE;
    jmp EndISR
Skip:
    //Set PORT_1_0 if carry is generated
    or PRT1DR, 0x01;
EndISR:
    reti
    
```

- With existing PWM module.** An existing PWM module can be used to perform the DSM. Keep the pulse width of an individual pulse fixed and vary the time period depending upon dimming value. Here, time period is inversely proportional to dimming value.

Steps to perform DSM using a PWM module:

- Choose an 8-bit PWM module and place it.
- Provide a clock of approximately 100 KHz as input clock. In this case it is VC2.

Figure 6. DSM Implementation using PWM Module



- Choose a desired, non-zero DELTA. In other words, it is also the pulse width.

- Apply the following transformation for time period values for various dimming levels.

If compare type is set to less then,

$$\text{Duty Cycle} = \text{Compare Value} / (\text{Time period} + 1)$$

In this case compare value is DELTA.

$$\text{Duty Cycle} = \text{DELTA} / (\text{Time period} + 1)$$

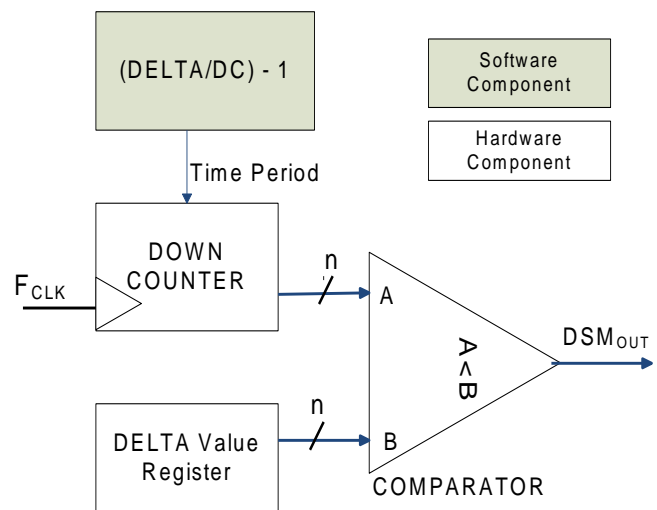
$$(\text{Time period} + 1) = \text{DELTA} / \text{Duty Cycle}$$

$$\text{Time period} = (\text{DELTA} / \text{Duty Cycle}) - 1$$

$$\text{Time Period} = (\text{DELTA} * 100 / \text{Duty Cycle}\%) - 1$$

Equation 1

Figure 7. HW/SW Implementation of DSM using Existing PWM Module



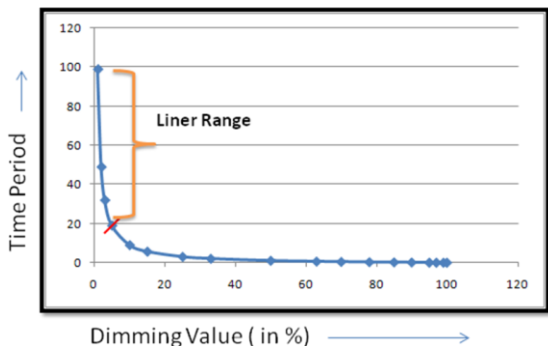
- Feed the calculated time period in step 4 to PWM modules to generate the equivalent dimming value.

Issues and Challenges in DSM

- Maximum Switching:** DSM increases the switching to maximum possible value. It is $F_{IN}/2$ in worst case condition. Due to this switching, a lot of spikes are seen in the spectral plot (refer Figure 3 on page 2). This causes significant EMI generation. Another disadvantage of this type of modulation is that it reduces the efficiency of switching supply by 10 to 15 percent. This is because high frequency components comparable to switching frequency of regulator circuitry are filtered out.

- **Loss of Linearity:** Current implementation of DSM which uses hardware PWM suffers from some significant non-linearity. Time period must always be inversely proportional to dimming value. However, there is a problem with single constant DELTA value.
1. If DELTA is low, the duty cycle and time period relationship is no longer linear at higher dimming values and it approaches zero. As shown in Equation 1, if value in time period register is 0 it generates 100% duty cycle.
 2. In [Figure 8](#), the DELTA value is 1. For DELTA = 1, period value saturates after 50% dimming value which causes full brightness at 50% dimming level itself. In other words, there is no control when requested signal density level is more than 50% in this implementation.

Figure 8. Plot of Time Period and Dimming Value for DELTA_1



3. If DELTA is high, the minimum signal density value cannot be produced below DELTA. For example, if DELTA is 5 then the minimum signal density levels represented are either 0 or 5/255. The range from 0 to DELTA is not covered. This is because the time period to produce signal density below DELTA is larger than n-bits and cannot be used in n-bit DSM scheme.

This illustrates that a single DELTA is not sufficient for the entire range of 0 to 2ⁿ. One possible improvement is to choose various DELTA values for various ranges. The next section discusses the variable DSM technique and addresses the downsides of DSM.

Variable Delta Sigma Modulation

The variable delta sigma modulation (vDSM) technique is an improved version of the existing DSM. It varies the minimum pulse width and time period of DSM to achieve dimming control. This is similar to delta sigma based PWM because its duty cycle and the time duration between two pulses are variable.

Duty cycle is defined as follows:

$$\text{Duty Cycle} = \frac{\text{High Time}}{\text{High Time} + \text{Low Time}} \tag{Equation 2}$$

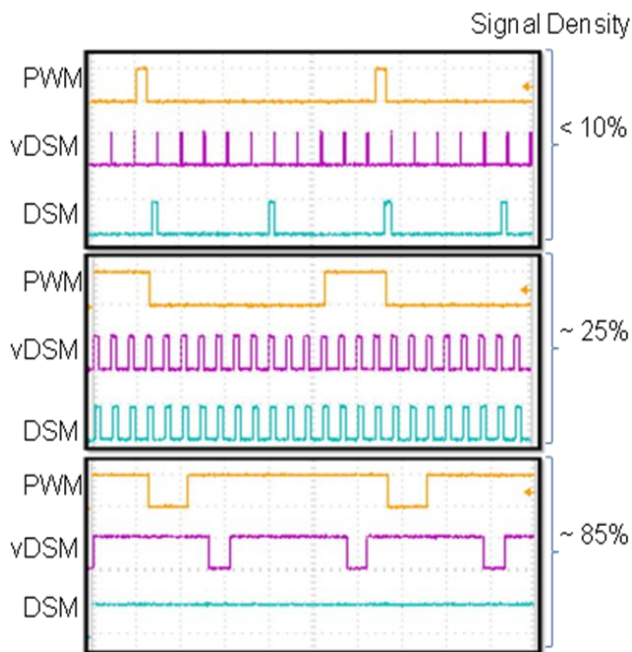
When High time of any pulse is changed and the denominator and summation of Low and High time is constant, it becomes PWM. Alternatively, Low time can also be varied to achieve dimming; this becomes DSM. However, by varying low and high time, there is better control over the range in which brightness control is achieved.

Sample Waveform of Variable DSM

The waveform in [Figure 9](#) depicts three modulation techniques for the same dimming values. The dimming value in the first case is less than 10%. The second and third set depicts the signal density of approximately 25% and 85%.

For variable DSM, DELTA values of 1/255, 5/255, and 125/255 respectively are selected after studying the relationship between time period and signal density for various DELTAs.

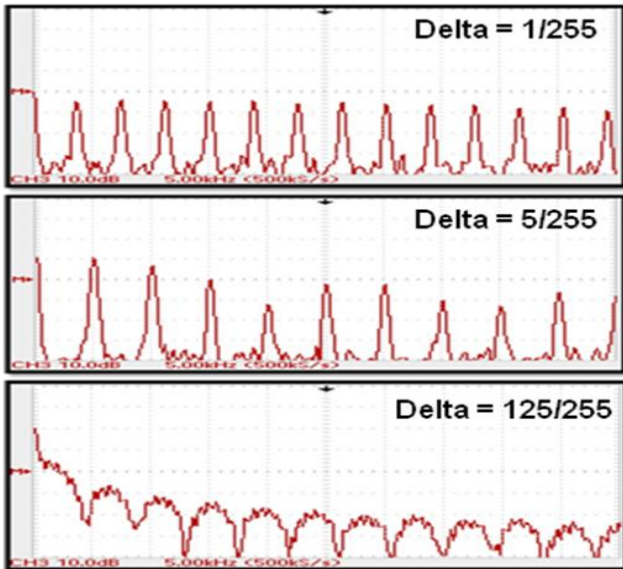
Figure 9. Comparison of PWM, VDSM, and DSM



Spectral Plot

FFT plots of variable DSM wave for various DELTA values are shown in Figure 10. Due to reduction in switching, the EMI is better for increased DELTA. The first plot is FFT of DSM when DELTA is 1/255. Due to high switching, there are a lot of spikes in this; the spikes are comparable to signal. The second plot is for DELTA 5/255 and EMI spikes are slightly reduced. The final plot depicts the spectral content of DSM when DELTA is 125/255. In this case, the EMI reduction is significant.

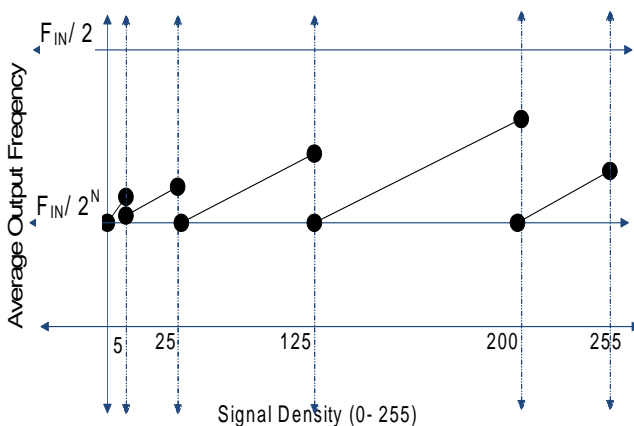
Figure 10. FFT Plots of vDSM for Various DELTAs



Output Frequency Plot

The problem with fixed pulse width DSM is that it increases the number of transitions to the largest possible value especially for high dimming value. However, by choosing different DELTAs for different operating ranges, the thinner pulses are clubbed together and the transitions are reduced. Maximum switching occurs at one of the extreme signal densities in the particular DELTA operating range. The good thing about variable DSM is that output frequency never reaches $F_{IN}/2$ as it does in SSDM and DSM.

Figure 11. Average Frequency Plot of vDSM

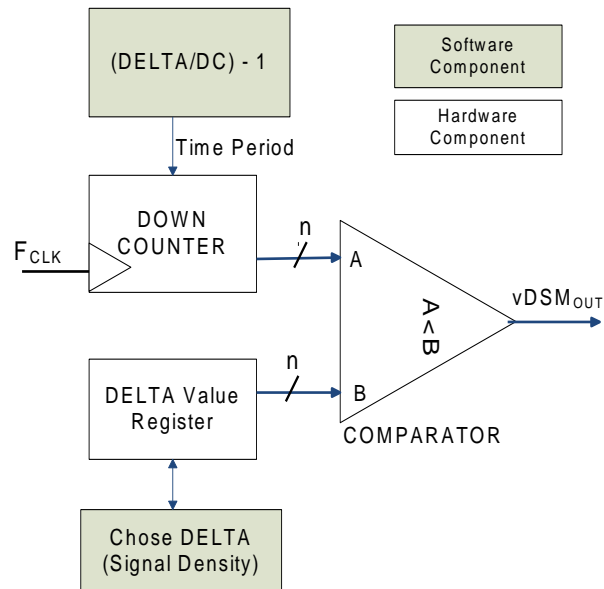


Implementation and Issues

This technique is implemented partly in software and partly in hardware. The basis is 8-bit PWM user module. Unlike fixed width DSM or PWM here both time period and duty cycle of output wave are changed.

Various signal density levels are chosen for different DELTAs. The pulse width is selected depending on desired signal density value.

Figure 12. HW/SW Implementation of vDSM using Existing PWM Module



When PWM module works in PWM mode, its time period is always fixed whereas pulse width changes in accordance to dimming value. In DSM, fix the pulse width register and vary time period to generate a particular duty cycle. If both are combined, there is better control over the dimming because there are two variables to control the dimming.

If compare type is set to “less than” for PWM module then

$$\text{Duty Cycle} = \text{Compare Value} / (\text{Period Value} + 1) \tag{Equation 3}$$

Variable ranges are:

Duty Cycle (DC) → 0 to 1

Compare Value (CV) → 0 to 255

Period Value (PV) → 0 to 255

$PV + 1 = CV / DC$

$PV = CV / DC - 1$

$PV = (CV * 100) / DC\% - 1 \rightarrow (4)$

CV is DELTA which is a variable value as dimming value changes.

Code 2. Firmware of vDSM

```

//Variable DSM Parameters
#define DELTA_1      1
#define DELTA_5      5
#define DELTA_25     25
#define DELTA_125   125
#define DELTA_200   200

void main()
{
    int temp;;
    char delta;
    /* start the Variable DSM Module */
    vDSM_Start ();

    // Apply power to the SC Block
    // Use POT to vary the dimming value
    PGA_Start(PGA_HIGHPOWER);
    ADC_Start(ADC_HIGHPOWER);

    // Enable Global Interrupts
    M8C_EnableGInt;
    for(;;)
    {
        wRawADCAccum = 0;
        //Get 8 samples and average them.
        for(i=0; i<8; i++)
        {
            ADC_GetSamples(1);
            while(ADC_fIsDataAvailable() == 0);
            wRawADCAccum += ADC_iClearFlagGetData();
        }
        // average the ADC value
        wRawADCAccum >>= 3;
        ADCValue = wRawADCAccum;

        SgnlDensity = ADCValue;

        // vDSM implementation
        // Signal Density below 5
        if(SgnlDensity < 5)
            delta = DELTA_1;

        // Signal Density between 5 to 24
        else if((SgnlDensity >= 5) && (SgnlDensity < 25))
            delta = DELTA_5;

        // Signal Density between 25 to 124
        else if((SgnlDensity >= 25) && (SgnlDensity < 125))
            delta = DELTA_25;

        // Signal Density between 125 to 199
        else if((SgnlDensity >= 125) && (SgnlDensity < 200))
            delta = DELTA_125;
        // Above 200 Signal Density
        else
            delta = DELTA_200;

        // Select the appropriate delta
        vDSM_WritePulseWidth(delta);

        //Calculate the time period
        temp = (SgnlDensity*2)/5;
        temp = (delta*100)/temp;
        vDSM_WritePeriod(temp - 1);
    }
}

```

Recommendation

The Microsoft Excel based utility, provided with this application note assists users to customize the firmware to their requirements. This utility helps to decide various DELTAs and their ranges.

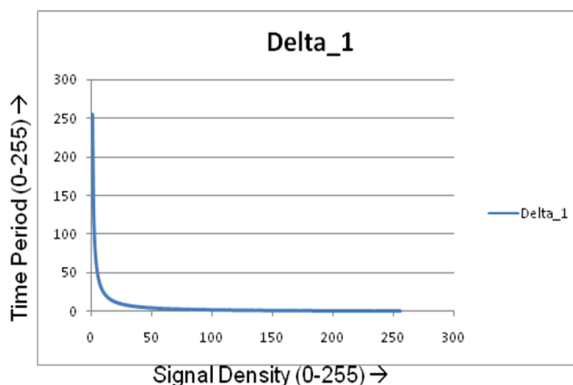
There are two problems in selecting various DELTAs.

Maintaining Linearity

For a fixed DELTA to control brightness there must be sufficient granularity in possible time period values. However, for smaller DELTA, above some level of signal density control over time period is lost because it has already reached the minimum possible value.

For example, for DELTA = 1, the plot between signal density and desired time period is shown in Figure 13.

Figure 13. Plot of Signal Density vs Desired Time Period for DELTA = 1

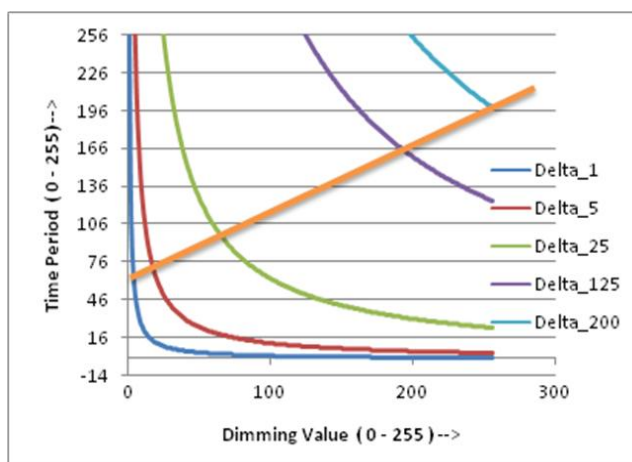


This plot illustrates that after a dimming value of 50, the time period approaches zero which results in 100% brightness of LED as per Equation 3.

It is important to choose DELTAs and their operating ranges in such a way that the time period never reaches a minimum possible value for that range.

The plot in Figure 14 shows the range of various DELTAs and their operating ranges. The DELTA time period above the orange line are not saturated and thus accepted.

Figure 14. Plot of Signal Density vs Desired Time Period for Various DELTAs



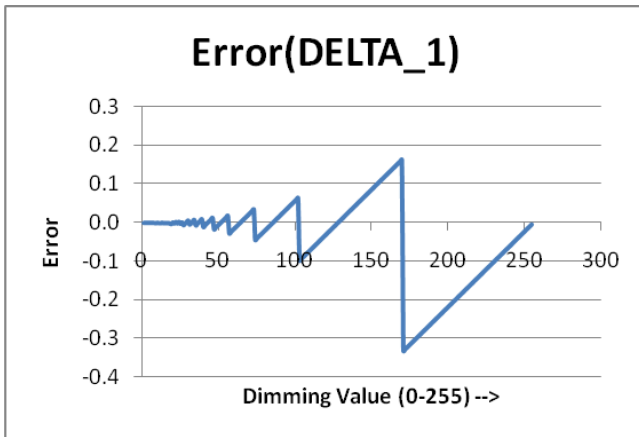
Minimizing Error

Another concern is error in the desired and produced brightness levels. When signal density value increases, try to reduce the time period. There is a lower limit of time period which exists for each DELTA. For example, DELTA of 1/255 or 5/255 has 100% brightness even at 50% signal density.

The time period for desired signal density is calculated from Equation 4. The time period is rounded to the nearest possible integer. This introduces an error between the desired signal density and actual signal density.

For DELTA = 1 the error of plot for various signal densities is shown in Figure 15.

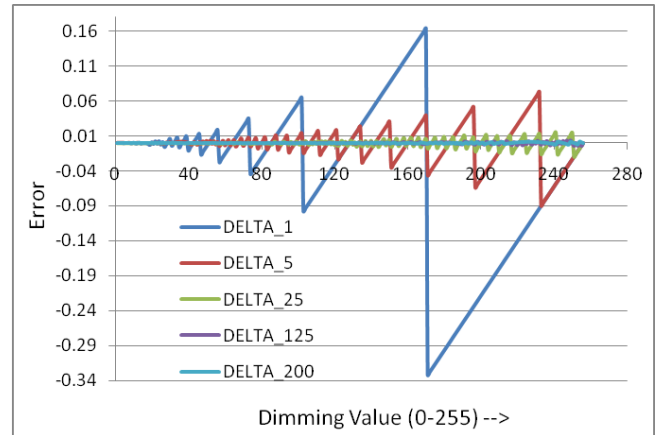
Figure 15. Plot of Signal Density vs Error for DELTA_1



This error increases as signal density increases. This is due to lesser granularities in time period at higher signal density. However, this is minimized by changing the DELTA and selecting various operating ranges.

The plot in Figure 16 confirms this.

Figure 16. Plot of Signal Density vs Error for Various DELTAs



From this plot, the DELTA and its ranges are decided as shown in Table 1. The error is actually quite small and hardly affects the color produced and brightness.

Table 1. DELTA Values: Operating Range and Error in Worst Case Scenario

DELTA	Dimming Value (Minimum)	Dimming Value (Maximum)	Error
1	0	4	< 0.1%
5	5	24	< 0.1%
25	25	124	< 0.3%
125	125	199	< 0.2%
200	200	255	< 0.2%

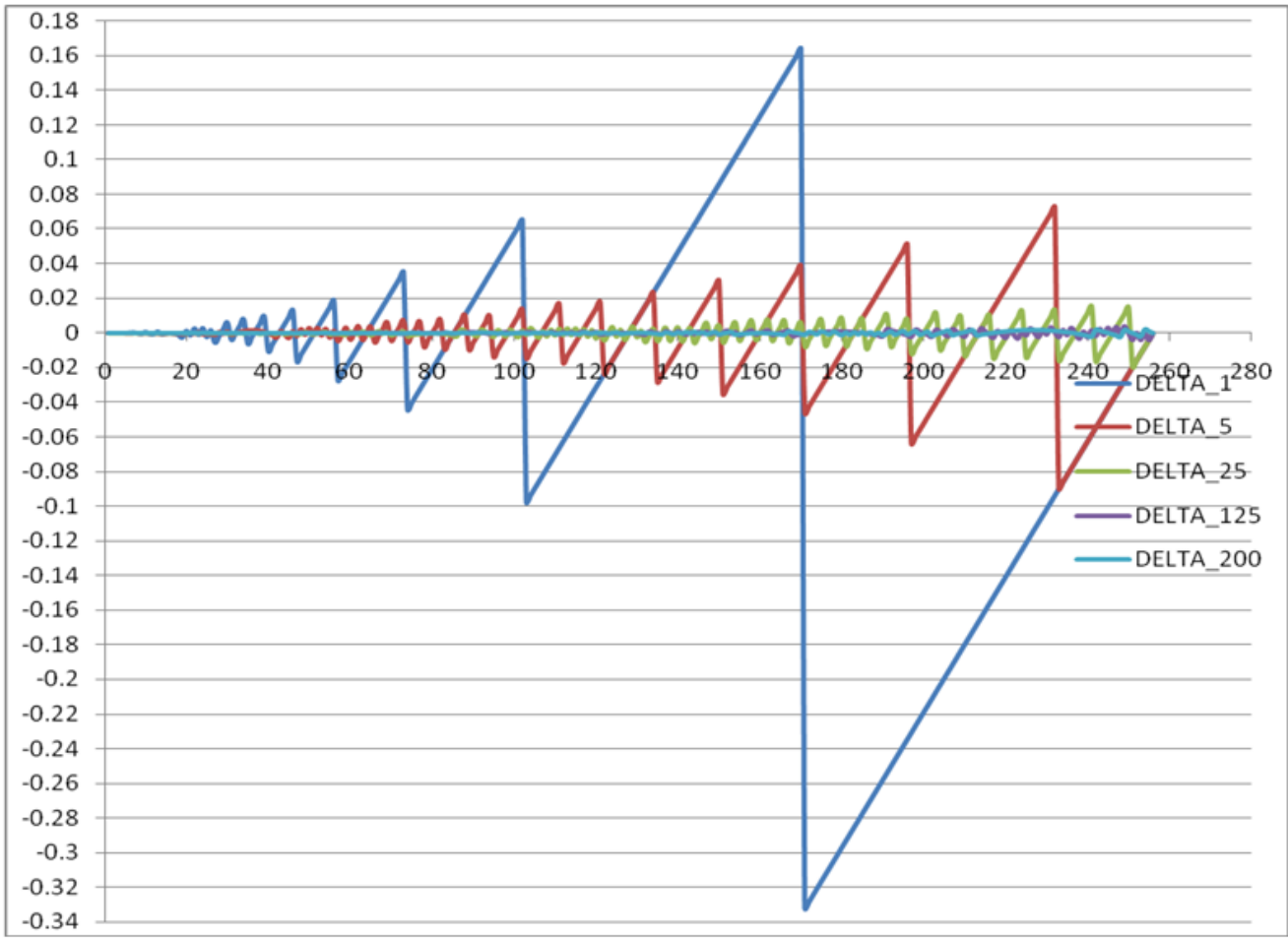
Summary

DSM and its variants are used in LED dimming. Variable DSM has lower EMI than normal DSM due to low switching frequency. It has better EMI than PWM because its frequency is variable and there are no spikes as in case of fixed PWM. It covers larger range of dimming values possible as compared to normal DSM. Variable DSM can be implemented using existing PWM modules.

Appendix

Error Plot for Various DELTAs

(Enlarged view of Figure 16)



About the Authors

Name:	Raj Kumar Singh Parihar	Name:	Anshul Gulati
Title:	Applications Engineer	Title:	Applications Engineer
Background:	Raj Kumar holds a bachelors degree in Electrical and Electronics from BITS – Pilani, India.	Background:	Anshul holds a bachelors degree in Electrical and Electronics from BITS – Pilani, India.
Contact:	Email – rksp@cyress.com Ph: +91 9940242401	Contact:	Email – gula@cyress.com

Document History Page

Document Title: Delta Sigma Modulation Using Cypress' High Brightness–LED Controllers

Document Number: 001-47778

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	2545839	RKSP/GULA	07/31/08	New Application Note

PSoC is a registered trademark of Cypress Semiconductor Corp. Programmable System-on-Chip, PSoC Designer, PSoC Express, and EZ-Color are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
<http://www.cypress.com/>

© Cypress Semiconductor Corporation, 2008. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

This Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.