XAPP615 (v1.1) June 25, 2003

Quantization

Author: Latha Pillai

Summary

This application note describes a reference design to do a quantization and inverse quantization of MPEG-2 video signals. After a brief introduction, the process of using JPEG and MPEG-2 standards for quantizing matrices is developed. Finally, implementing the Xilinx solution for quantization or inverse quantization is described.

Introduction to Quantization

Quantization is the process of selectively discarding visual information without a significant loss in the visual effect. Quantization reduces the number of bits needed to store an integer value by reducing the precision of the integer. Each discrete cosine transform (DCT) component is divided by a separate quantization coefficient, and rounded to the nearest integer. The larger the quantization coefficient (i.e., coefficient weighting), the smaller the resulting answer and associated bits needed to express the DCT component. In the reverse process, the fractional bits are "rounded" and are recovered as zeros, constituting a precision loss from the original number. Quantization could be considered as input data binning where the number of bins is less than the number of possible input values. The number of bins is decided by the quantization factor Q. If the input data range is from one to 60, and if Q is 5, then 60/5 is 12 bins (0 to 5, 6 to 10, and so on). A different input data range of 60 is now reduced to 12 possible bins.

A DCT applied to an 8 x 8 block of pixel components has the effect of organizing the components into different spatial frequencies. The lower spatial frequency values or smoother spatial contours appear toward the upper left-hand corner of the coefficient matrix, and the higher spatial frequency values appear in the bottom right-hand corner. The uppermost left-handed value is known as the DC component and expresses a solid color value for the entire block. If the entire block is a solid color, then only this value is needed to recreate the 8 x 8 pixel components.

The human visual system is less sensitive to higher frequency spatial contours. Objects in a scene are recognized by lower frequency spatial contours. Quantization or reduction in the accuracy of the higher spatial frequency contour values does not dramatically affect the image quality, since lower spatial frequencies are unchanged.

The MPEG-2 standard uses different types of quantization. Quantization where parts of the higher frequency coefficients are not transmitted is considered zonal sampling. Scalar Quantization (SQ) is when quantization is performed on each individual coefficient.

Quantization can also be performed on a group of coefficients together. This is known as Vector Quantization (VQ). Both uniform and non-uniform quantizers can be used depending on the problem. In some quantization, there is an increased threshold level around zero, described as a dead-zone. The effect of the increased threshold is to increase the number of coefficients quantized to zero. Small variations in input signals around zero are usually caused by noise. Quantization with a dead-zone around zero actually eliminates the noise around zero and improves the signal quality.

Why Use Quantization?

Quantization is done to achieve better compression. Quantization reduces the number of bits needed to store information by reducing the size of the integers representing the information in the scene. These are details that the human visual system ignores. This step represents one key segment in the multi- compression process. A reduction in the number of bits reduces storage capacity needed, improves bandwidth, and lowers implementation costs.

How to Use Quantization

JPEG

JPEG uses an 8 x 8 quantization matrix, the Q matrix, for each 8 x 8 DCT block to be compressed. For an 8-bit DCT coefficient, each element in the Q matrix can have an integer value between one and 255. The Q matrix for JPEG is designed using two techniques, a Q matrix developed using psycho visual experiments, and a Q matrix based on rate distortion theory and bit rate control. The default Q matrix is derived from the psycho visual experiments for luminance and chrominance components.

$$Q\ Table\ For\ Luminance \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$Q\ Table\ For\ Chrominance \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

The quantization of DCT components in a JPEG picture is shown in Equation 1.

$$Quantized\ Coefficient\ (i,j) = \frac{DCT(i,j)}{Qmatrix(i,j)} \quad \text{Eq. 1}$$

MPEG-2

The MPEG standard consists of three different picture types. The intra-coded "I" picture where an "I" picture is coded with reference to itself (i.e., information from no other frames are used to code an "I" frame.) A predicted "P" picture is coded using information from a previous "P" or "I" picture. The information from a previous picture is used to predict the motion in time of the current P frame. A bidirectional "B" predicted frame uses information from past, and/or future "I" or "P" pictures. Motion compensated prediction is again used for coding B pictures. An intra-picture frame is quantized using the intra-quantizer matrix. The intra-frame AC coefficients (i.e., all coefficients other than (0,0)) are quantized without a dead-zone using an intra-quantizer matrix. The P (predictive) and B (bidirectional) frames are quantized by uniform quantizer with a dead-zone around zero. A dead zone is described as a quantized to zero area larger than the step size with decreasing sensitivity to input noise. A dead zone results in a string of zeroes at

the output. A non-intra quantizer matrix is used in this application note. MPEG-2 also allows the use of a user specified quantization matrix.

The process of DCT coefficient quantization is described in this section. The output of a 2D-DCT is read out one value at a time as defined in the MPEG-2 standard. Each DCT coefficient is divided by a corresponding quantization matrix value supplied by the quantization matrix. The encoder can use the default matrix or it can substitute a new user defined quantization matrix at a picture level and download it to the decoder via the bitstream.

In a 4:4:4 format there is no decimation of chrominance components. For every one Y or luma component, there is one Cb and one Cr component. In 4:2:2 format, both chrominance components are decimated by two horizontally, making the number of chrominance components along a given line be half the number of luminance components. In a 4:2:0 format, both chrominance components are decimated horizontally as well as vertically where the number of Cr or Cb components horizontally and vertically is half the number of luminance (Y) component. For every four Y samples, there is one Cr and one Cb sample.

With 4:2:0 data, two Q matrices are used, one for intra-macroblocks and one for non-intra-macroblocks. For 4:2:2 and 4:4:4 data, four matrices are used, two sets (intra and non-intra) for luminance and two sets (intra and non-intra) for chrominance. The default matrices corresponding to the luminance portion are loaded first. For 4:2:2 and 4:4:4 data, a different user-defined chrominance matrix can be loaded later.

$$\textit{Intra Quantizer Matrix} \begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix}$$

$$\textit{Non Intra Quantizer Matrix} \begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

When a sequence header code in the video bitstream is decoded, all quantization matrices are reset to the default matrices. User defined matrices can be downloaded in the sequence header or the "quant matrix" extension. The Q matrix has 64, 8-bit unsigned values. Value zero is not used since division by zero is not applicable.

The quantization of AC coefficients are given in Equation 2.

$$QDCT(i,j) = \frac{\left[\frac{32 \times DCT(i,j)}{Qmatrix(i,j) \times Quantizer\ Scale} + k\right]}{2} \quad \text{Eq. 2}$$

Where k = 0 for intra blocks

k = sign of DCT coefficient for non-intra blocks

1 if $DCT_{(i,j)} > 0$

0 if $DCT_{(i,j)} == 0$

–1 if $DCT_{(i,j)} < 0$

$Qmatrix_{(i,j)}$ is the $(i,j)^{TH}$ element of the corresponding quantizer matrix given. Quantized (i,j) is limited to the range [–2048, 2047].

For DC coefficients of intra-coded blocks, the quantizer step-size is determined by the selected DCT precision. The quantization of DCT components in an MPEG-2 picture is shown in Equation 3.

$$Quantized\ Coefficient\ (0,0) = \frac{DCT(0,0)}{k} \quad \text{Eq. 3}$$

Where k is a constant determined by the DCT precision selected: k = 8 for 8-bit precision, k = 4 for 9-bit precision, k = 2 for 10-bit precision, and k = 1 for 11-bit precision. MPEG-2 allows changing the quantization scale factor on a macroblock basis to promote achieving better compression ratios.

Each value in the quantization matrix is pre-scaled when multiplied by a single integer value, known as the quantizer scale code. The quantizer scale code is a 5-bit unsigned integer in the one to 31 range. The encoder/decoder uses this value until another quantizer scale code is encountered either in a slice or a macroblock. The value zero is forbidden. Each of the 31 values of the quantizer scale code are mapped to two sets of integers, called "quantizer_scale", ranging from one to 112. The quantizer scale type is used to indicate if either the quantizer scale code or the quantizer_scale applies.

The value of the quantizer scale code is modifiable on a macroblock basis, making it useful as a fine-tuning parameter for bit-rate control, since it is not economical to send an entirely new matrix on a macroblock basis. This operation forces as many of the DCT coefficients to zero, or near zero, as possible within the boundaries of the prescribed bit-rate and video quality parameters.

Table 1: **Relationship Between quantizer_scale and quantizer_scale_code**

quantizer_scale_code	quantizer_scale[q_scale_type]	
	q_scale_type = 0	q_scale_type = 1
0	(forbidden)	
1	2	1
2	4	2
3	6	3
4	8	4
5	10	5
6	12	6
7	14	7

Table 1: **Relationship Between quantizer_scale and quantizer_scale_code *(Continued)***

quantizer_scale_code	quantizer_scale[q_scale_type]	
	q_scale_type = 0	q_scale_type = 1
8	16	8
9	18	10
10	20	12
11	22	14
12	24	16
13	26	18
14	28	20
15	30	22
16	32	24
17	34	28
18	36	32
19	38	36
20	40	40
21	42	44
22	44	48
23	46	52
24	48	56
25	50	64
26	52	72
27	54	80
28	56	88
29	58	96
30	60	104
31	62	112

Notes:

1. MPEG Video Compression standard by Mitchell, Pennebaker et. al., page 218

Inverse Quantization

The inverse quantizer output for an MPEG-2 system is shown in Equation 4.

$$DCT = \frac{(2 \times QDCTf(i,j) + k) \times Qmatrix(i,j) \times quantiser\ scale}{32} \quad \text{Eq. 4}$$

Where k = 0 for intra blocks

k = sign of quantized coefficient for non-intra blocks

(1 if quantized (i,j) > 0, 0 if quantized (i,j) == 0, –1 if quantized (i,j) < 0)

The results from inverse quantization are limited: –2048 to 2047. A mismatch control is applied to the output of the inverse quantization. The final inverse quantized output is the same as the outputs of Equation 4 for all elements except coefficient [7,7]. All the coefficients resulting from Equation 4 are added together.

Final[7,7] = Equation 4 [7,7] if sum is odd

Final[7,7] = Equation 4 [7,7] – 1 if sum is even and Equation 4 [7,7] is odd

Final[7,7] = Equation 4 [7,7] + 1 if sum is even and Equation 4 [7,7] is even

Mismatch error occurs due to difference in the DCT at the encoder and IDCT at the decoder. Ideally, an 8 x 8 block passed through a DCT then passed through an IDCT gives back the original 8 x 8 block values. Because the quantization is done after DCT, and the subsequent errors are introduced during the quantization division process, the quantized DCT values after IDCT could be different from the original 8 x 8 block values. The error caused due to this mismatch is propagated until an intra-frame coding is done. Mismatch control attempts to error by eliminating bit patterns which statistically have the greatest contribution towards mismatches. Mismatch control in MPEG-2 is called "LSB Toggling". Toggling affects only the least significant bit (LSB) of the sixty-third AC DCT coefficient (the highest frequency in the DCT matrix). In MPEG-1 the mismatch control, called oddification, is performed on the quantized DCT coefficients. Whereas, in MPEG-2 toggling is performed on the DCT coefficients after inverse quantization.

Implementation in a Xilinx FPGA

Quantization

When implementing Quantization in a Xilinx FPGA, start with Equation 2. Rewriting Equation 2 gives Equation 5.

$$Quantized\ Coefficient(i,j) = \frac{16 \times DCT(i,j)}{Qmatrix(i,j) \times quantiser\ scale} + \frac{k}{2} \quad \text{Eq. 5}$$

All the possible values for 16/quantiser_scale are stored in ROM q_scale_mem. With two possible quantizer scale types, and with each type having 32 possible quantizer scale codes, the quantizer scale can have up to 64 different values. Since the Qmatrix coefficients are up to 8-bits wide, these coefficients can have up to 256 possible values. 1/Qmatrix values are calculated for all the 256 values and are stored in a second ROM, q_value_mem.

The 64 values of the Qmatrix, either default, user-defined intra, or user-defined non-intra, are stored in RAMs called def_q_mem, q_value_new_intra, and q_value_new_non_intra respectively. Each of the 64 Qmatrix values are used to choose the corresponding 1/Qmatrix value from the q_value_mem.

The input DCT value is multiplied with the corresponding value from q_value_mem. The result is then multiplied with the output of q_scale_mem, selected by the input quantizer scale code. Figure 1 shows the quantizer implementation in the decoder.

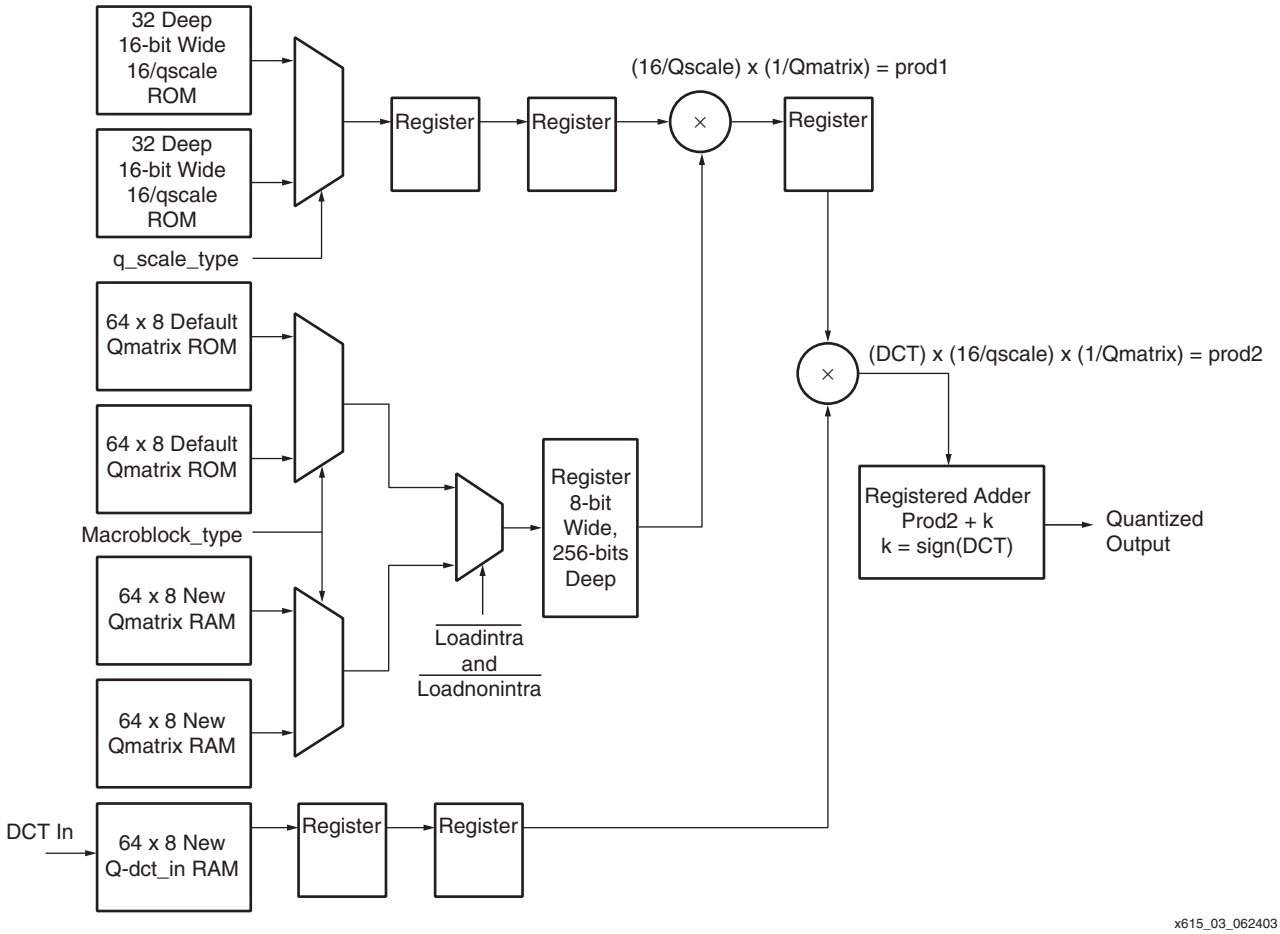

Figure 1: **Quantizer Implementation in the Decoder**

Inverse Quantization

The quantized Discrete Cosine transform coded coefficients are fed into the quantizer. The quantized coefficients are taken through an inverse quantizer to get back the original DCT coefficients. Since quantizing is a lossy process where certain DCT coefficients are thrown away, the inverse quantization will not given back all of the original 64 DCT coefficients. The non-recovered coefficients are have the least visual effect on the picture. The DC value or the DCT coefficient at (0,0) is quantized using Equation 3. The inverse quantization is done as in Equation 4. Figure 2 shows the inverse quantizer implementation in the decoder.

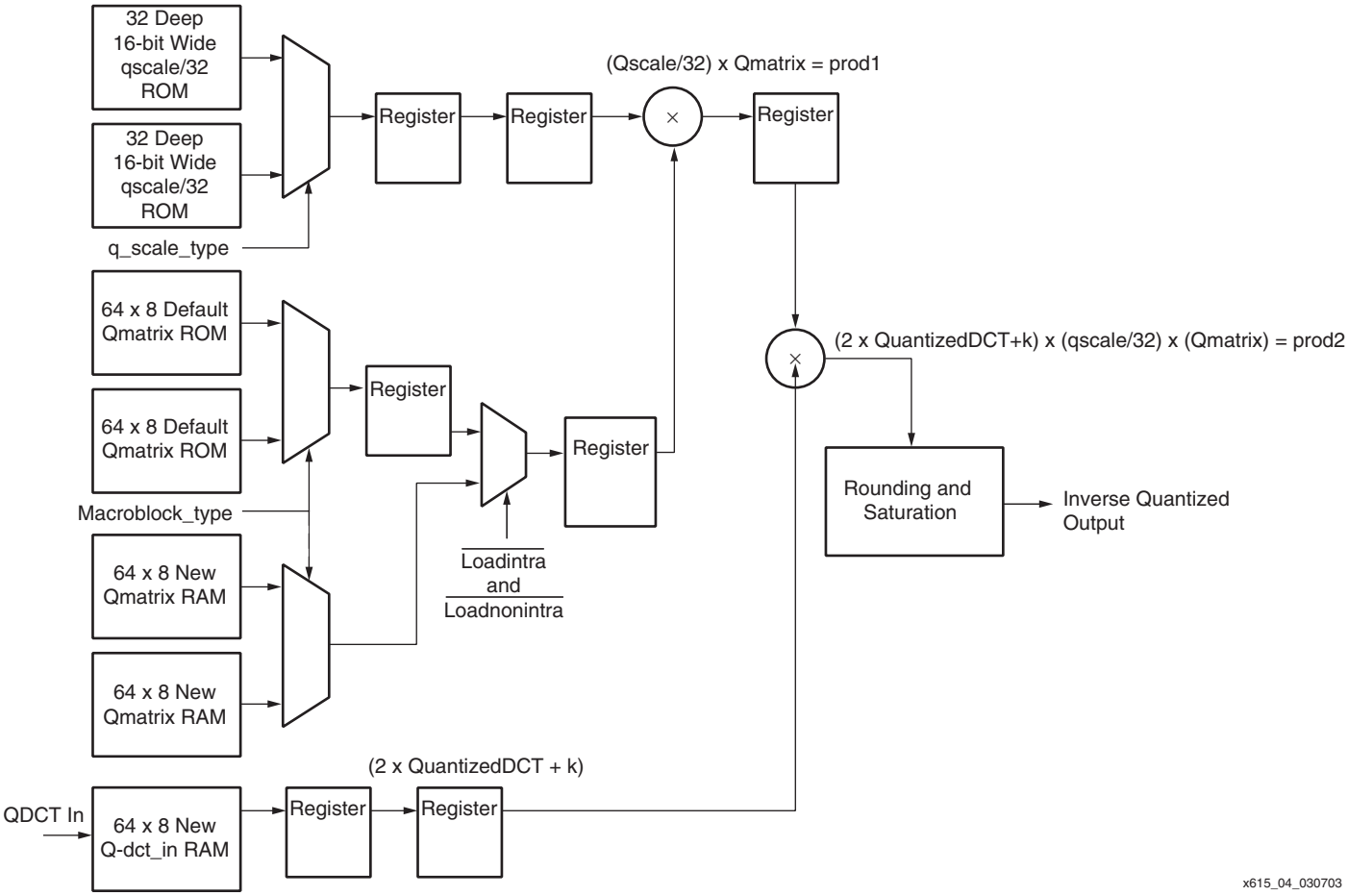

Figure 2: **Inverse Quantizer Implementation in the Decoder**

Resource Utilization

Table 2: **Quantization**

Device	Post-Route (Synthesis Constraint)	LUTs (Flip-Flops)
XCV300E -8 BG352	94.76 (100 MHz)	794 (276)
XC2V250- 6 FG456	121.89 (100 MHz)	538 (203) and two Mult18 x 18s
XCV300 -6 PQ240	75.81 (75 MHz)	794 (276)
XC2S200 -6 FG256	75.56 (75 MHz)	794 (276)

Table 3: **Inverse Quantization**

Device	Post-Route (Synthesis Constraint)	LUTs (Flip-Flops)
XCV300E -8 BG352	102.43 (100 MHz)	662 (288)
XC2V250 -6 FG456	120.32 (100 MHz)	411 (232) and two Mult18 x 18s
XCV300 -6 PQ240	80.66 (75 MHz)	660 (278)
XC2S200 -6 FG256	75.51 (75 MHz)	660 (278)

The performance can be improved by adding more pipeline stages in the designs. The place and route effort level was kept at "high". To get an update on performance and utilization, always re-run the designs using the latest Xilinx software. Figure 3 shows the input DCT signal (dct_in), quantized DCT signal (quant_dct_out), and the inverse quantized DCT signal (q_out).

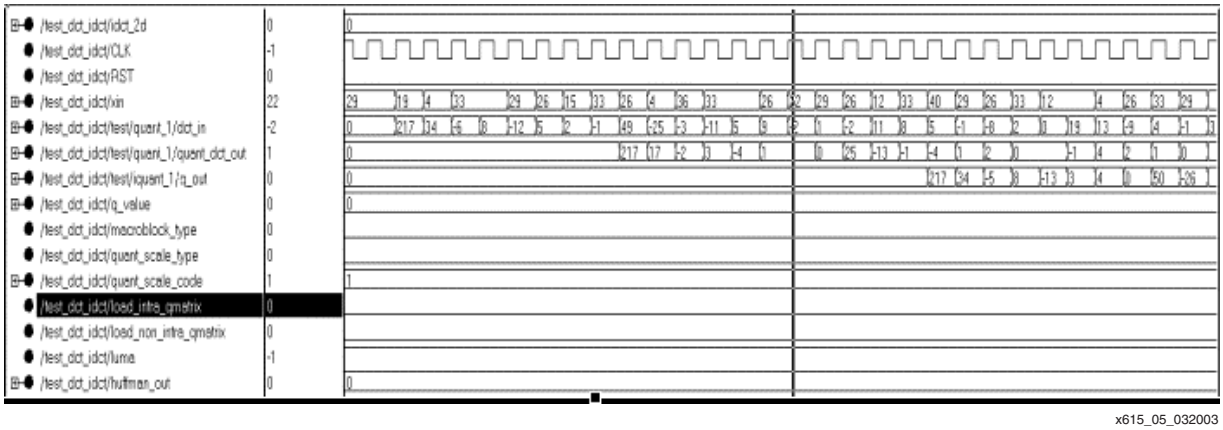

Figure 3: **DCT Simulation Waveform**

Reference Design

Quantization reference design files in both VHDL and Verilog are available on the Xilinx FTP site at:

ftp://ftp.xilinx.com/pub/applications/xapp/xapp615.zip

Conclusion

This Quantization application note describes the quantization and inverse quantization algorithms used in an MPEG-2 encoder and/or decoder. The design files show the efficient implementation of the algorithms on Xilinx devices. The code can be used to target any Xilinx device. Optimize the code by instantiating the adder/subtractor and multiplier units when targeting Virtex devices. The quantization block has an initial latency of 10 clock cycles. The initial latency of the inverse quantization block is 13 clock cycles. After the initial latency, one output value is obtained at every clock for both the blocks. The performance of both blocks can be increased by adding more pipeline stages.

References

1. Image and Video Compression Standards, Second Edition, by Vasudev Bhaskaran and Konstantinos Konstantinides, ISBN 0-7923-9952-8
2. MPEG Video Compression Standard, by Mitchell, Pennebaker, Fogg, and LeGall, ISBN 0-412-08771-5
3. MPEG-2 Video IS document, ISO/IEC 13818-2: 1995(E)

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
04/22/03	1.0	Initial Xilinx release.
06/25/03	1.1	Update constant "k" on page 4 and Figure 1.