Low Power Design of DCT and IDCT for Low Bit Rate Video Codecs

Nathaniel J. August and Dong Sam Ha, *Senior Member, IEEE*

Abstract—This paper examines low power design techniques for discrete cosine transform (DCT) and inverse discrete cosine transform (IDCT) circuits applicable for low bit rate wireless video systems. The techniques include skipping DCT computation of low energy macroblocks, skipping IDCT computation of blocks with all coefficients equal to zero, using lower precision constant multipliers, gating the clock, and reducing transitions in the data path. The proposed DCT and IDCT circuits reduce power dissipation by, on average, 94% over baseline reference circuits.

Index Terms—Discrete cosine transform (DCT), H.263, inverse discrete cosine transform (IDCT), low power, video codec.

I. INTRODUCTION

LOW BIT RATE wireless video systems have applications in cellular videophones, wireless surveillance systems, and mobile patrols. The ITU-T H.263 [1] video codec standard is suitable for low bit rate wireless video systems. A critical requirement for portable wireless video systems is low power dissipation. This paper examines several low power design techniques for discrete cosine transform (DCT) and inverse DCT (IDCT) hardware design in an H.263 codec.

Fig. 1 shows a block diagram of the major operations performed in an H.263 encoder and decoder. Starting at the encoder, the motion estimation block (MEC) performs temporal compression by computing the difference between the current frame and the previous frame. The temporally compressed data is sent to the DCT, which is a key component in the spatial compression of the data. The DCT block transforms the data into spatial frequency coefficients. The quantization (Quant) block divides each DCT coefficient by a quantization parameter, setting insignificant DCT coefficients to zero. The variable length coder (VLC) performs run length coding on the quantized coefficients, compressing long runs of DCT coefficients.

The coder sends the compressed data to the decoder, which applies the inverse process to reconstruct a frame. The variable length decoder (VLC^{-1}) reconstructs the quantized data, and the inverse quantization (Quant^{-1}) block multiplies by the quantization parameter to recreate the DCT coefficients. Next, the IDCT block transforms the DCT coefficients back to the spatial domain. Finally, the inverse motion estimation block
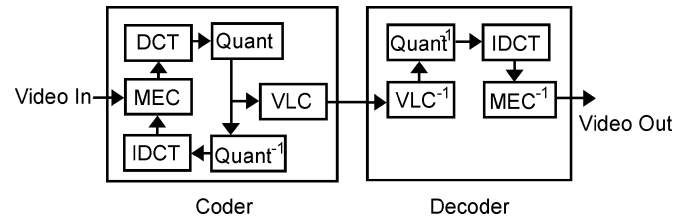
Fig. 1.　H.263 coder and decoder block diagram.

(MEC^{-1}) reconstructs the original frame by adding the transmitted difference to the previous frame. These reconstruction steps also occur in the encoder; both the encoder and the decoder have an identical reference copy of the previous frame.

The DCT and the IDCT are computationally intensive in H.263. The combined computational complexity of the DCT and IDCT in the coder surpasses that of any other unit, consuming 21% of the total computations [2]. The IDCT in the decoder also incurs the largest computational cost. The high computational complexity of the DCT and IDCT leads to high power dissipation of the blocks, so low power design of DCT and IDCT units are essential for portable wireless video systems based on H.263.

II. PRELIMINARIES

In this section, we explain the concepts and terms necessary to understand the paper, and we review previous low power designs for DCT/ IDCT.

A. Terms

In H.263, a **macroblock** is a basic unit of data that represents a 16×16 pixel area of a video frame. The motion estimation unit operates on macroblocks. Macroblocks conveniently represent data in YCbCr format, which contains a luminance component (Y), a blue chrominance component (Cb), and a red chrominance component (Cr). Luminance blocks describe the intensity or brightness of pixels, whereas chrominance blocks contain information about the coloration of pixels. A macroblock contains six 8×8 blocks: four blocks contain luminance values; one block contains blue chrominance values; and one block contains red chrominance values. The DCT, IDCT, and VLC units operate on blocks. Since the human eye is less sensitive to color than to intensity, each chrominance block is downsampled by a factor of two in both the x and y directions. Each luminance value corresponds to one pixel, whereas each chrominance value is shared by four pixels.

In motion estimation, the **sum of absolute differences** (SAD) measures how well a macroblock in the current frame matches a nearby macroblock in the previous reference frame. SAD values are obtained with the following equation:

$$SAD = \sum_{k=1}^{16} \sum_{l=1}^{16} |Y_{i,j}(k,l) - Y_{i-u,i-v}(k,l)|. \quad (1)$$

In (1), the magnitude of each luminance sample, $Y_{i-u,j-v}(k,l)$, in a candidate (i.e., reference) macroblock that is offset by (u,v) in the previous frame is subtracted from the magnitude of each luminance pixel, $Y_{i,j}(k,l)$, in the current macroblock at position (i,j). The motion estimation unit chooses the candidate macroblock with the lowest SAD as the most likely match for the current macroblock. This aids the compression process, since only the small difference between the chosen macroblock and the current macroblock is sent to the DCT. Sequences with large amounts of motion tend to produce larger magnitude SAD values, whereas sequences with less motion tend to produce lower magnitude SAD values.

The average **peak signal to noise ratio** (PSNR) of a frame in a video sequence is measured as

$$PSNR = 10 \log \left[x \cdot y \sum_{rows} \sum_{cols} \left(\frac{255}{Y_1 - Y_2} \right)^2 \right] \quad (2)$$

where x is the number of rows, y is the number of columns, and Y_1 and Y_2 are the luminance values in the original and the reconstructed pictures. We use the average PSNR over all frames as a quantitative measure of the quality of our H.263 implementations.

Embedded in the H.263 bit stream, the **coded block pattern** (CBP) is set to "1" for a block with at least one nonzero DCT coefficient excluding the DC coefficient position at (0, 0). Otherwise, the CBP is set to "0". An INTER block (which is temporally compressed) produces IDCT input with all coefficients equal to zero if its DC coefficient is zero and its CBP is "0". Additionally, if the **coded macroblock indication** (COD) bit is set, the entire macroblock has all coefficients equal to zero.

B. DCT/IDCT

The DCT and IDCT are important components in many compression and decompression standards, including H.263, MPEG, and JPEG. The two-dimensional (2-D) DCT in (3) transforms an 8×8 block of picture samples $x(m,n)$, into spatial frequency components $Y(k,l)$ for $0 \leq k, l \leq 7$. The IDCT in (4) performs the inverse transform for $0 \leq m, n \leq 7$. In (3) and (4), $\alpha(0) = 1/\sqrt{2}$ and $\alpha(j) = 1, j \neq 0$:

$$Y(k,l)$$
$$= \frac{1}{4} \alpha(k)\alpha(l) \sum_{n=0}^{7} \sum_{m=0}^{7} x(m,n) \cos\left(\frac{(2m+1)\pi k}{16}\right) \cos\left(\frac{(2n+1)\pi l}{16}\right) \quad (3)$$

$$x(m,n)$$
$$= \frac{1}{4} \alpha(k)\alpha(l) \sum_{k=0}^{7} \sum_{l=0}^{7} Y(k,l) \cos\left(\frac{(2m+1)\pi k}{16}\right) \cos\left(\frac{(2n+1)\pi l}{16}\right). \quad (4)$$

In the matrix of spatial frequency components, the low frequency coefficients correspond to low indices (at the top and left of the matrix); higher frequency coefficients correspond to higher indices (at bottom and the right of the matrix). High frequency coefficients have small magnitudes for typical video data, and the human eye is less sensitive to high frequencies as to low frequencies. In compression schemes, the quantizer block (Quant in Fig. 1) forces the insignificant high frequency coefficients to zero. The IDCT performs the inverse of DCT, transforming spatial frequency components to the spatial domain.

C. DCT/IDCT Algorithms

Since the straightforward implementations of (3) and (4) are computationally expensive (with 4096 multiplications), most implementations employ fast algorithms that reduce the computational cost. Fast algorithms can be broken down into two broad categories: row/column approaches and direct, fast 2-D approaches. The row/column approach results in simple and regular implementations, but it is less computationally efficient than direct, fast 2-D implementations.

For the row/column approach, the one–dimensional (1-D) DCT/IDCT of each row of input data is taken, and these intermediate values are transposed. Then, the 1-D DCT/IDCT of each row of the transposed values results in the 2-D DCT/IDCT. The straightforward implementation of the row/column approach reduces the number of multiplications for a 2-D DCT/IDCT to 1024. Many row/column implementations further reduce computation by using fast 1-D DCT/IDCT algorithms such as the Chen algorithm and similar algorithms [3]–[5]. The Chen algorithm requires only 16 multiplications for an eight point 1-D DCT/IDCT and 256 multiplications for a row/column 2-D DCT/IDCT. Architectures based on the Chen algorithm are a popular choice for implementing a 2-D DCT/IDCT [6]–[14].

The direct, fast 2-D DCT/IDCT approach usually requires about half the computations of the row/column approach at the expense of irregularity in the data paths and more complex control logic. Another advantage to the direct, fast approach is the elimination of the transposition memory, which reduces latency. However, the elimination of the transposition memory is usually offset by the additional memory necessary to reorder inputs and store intermediate values. Three different methods are often used to implement the direct fast 2-D approach: the matrix method, the vector-radix method, and the time-recursive method [15]–[25].

D. Review of Low Power Design Techniques for DCT/IDCT

The following power reduction techniques have previously been explored to enhance implementations of the above DCT/IDCT algorithms. The general techniques apply to most digital circuits, whereas techniques specific to DCT/IDCT take advantage of the characteristics of typical video data or focus on the multipliers.

Some general low power techniques include clock gating, pipelining, and voltage scaling [26]. The low power design in [18] uses a parallel, distributed architecture to reduce the supply voltage. Other low power designs employ parallel processing

units that enable power savings from a reduction in clock speed [22], [23]. Low power libraries reduce power in [13], [18].

To save power, many architectures reduce calculations for visually irrelevant DCT coefficients. Xanthopoulos and Chandrakasan employ arithmetic units in which the precision changes adaptively depending on the visual significance of the data [11]. Another architecture allows fine (1-bit increments) resolution for the precision control on adders that successively approximate toward the final value [27]. The upper limit in the number of approximations is determined from the peak-to-peak pixel difference. The precision can also be determined by the quantization parameter: multiplier units use fewer bits for high quantization parameters and more bits for low quantization parameters [16]. Li and Lu propose skipping the computation of visually insignificant high frequency DCT coefficients altogether [13]. Their method removes the circuit elements that compute high frequency coefficients and sets these coefficients to zero.

By ignoring redundant sign bits, arithmetic units save power in the DCT circuit because of the large amount of data with a small magnitude. Small values occur frequently in INTER frames because the motion estimation unit sends only the difference between the previous frame and the current frame. One architecture reduces power for small coefficients by successively deactivating four adder partitions if they work on redundant sign bits [5]. Another architecture ignores the most significant bits of the inputs if they are common to both addends, thereby reducing addition operations and ROM accesses [11].

Since the majority of input data for the IDCT is comprised of zero-valued coefficients, significant power reduction can come from disabling adders and multipliers for zero-valued operands. Some architectures use a zero detect signal that skips addition and multiplication operations for zero-valued data [11], [16], [27]. The arithmetic units benefit from a standby mode, where they consume a minimum amount of power while idle.

Another popular target for power reduction in DCT/IDCT blocks is the multiplier implementation. The two most popular implementations for multipliers are bit-serial (distributed architecture) and bit-parallel. A comparison between the two architectures reveals that the bit-parallel architecture dissipates less power in a 2-D DCT circuit [28].

Bit-serial architectures require more power due to the high internal frequency, the serialized operation, and the high capacitance of the ROM address and bit lines. However, bit-serial architectures have the advantage of easily and finely partitioning data for use in architectures that rely on variable precision arithmetic to save power.

Several designs of parallel multipliers, which compromise speed, area, and power dissipation, have been proposed. An array multiplier is a straightforward implementation of the bit parallel architecture; it is easily implemented from library cells [14]. Other choices include ROM-based multipliers [6] and PLA-based multipliers [27]. The fact that one multiplicand is known *a priori* in the DCT/IDCT can be exploited for optimization of multipliers that use a shift-and-add approach [8], [12], [13], [27], [29]. One interesting multiplier uses rotation-based arithmetic, which reduces shift-and-add operations by 28% [22], [23].
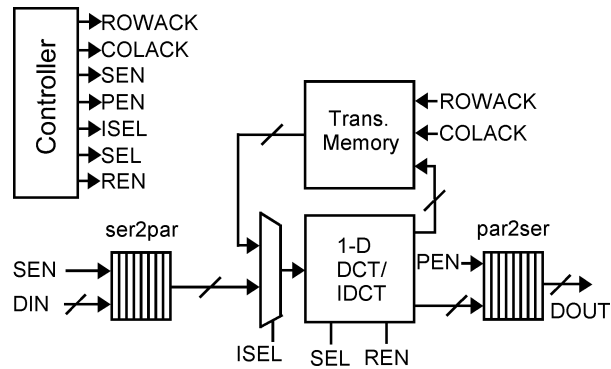

Fig. 2. Two-dimensional DCT/IDCT architecture.

III. BASELINE DCT/IDCT

The baseline DCT/IDCT provides a reference design for application of our low power techniques. To evaluate the effectiveness of these techniques, we compare our low power DCT/IDCT designs with the baseline design. To ensure that the final architecture will be power-efficient, the baseline design features the power-efficient Chen algorithm from Section II-C. To fairly and independently assess our power savings techniques, the baseline models avoid any power savings techniques from Section II-D.

Since the lower latency of the direct 2-D approach does not benefit low bit rate video, the baseline design employs a row/column approach (which is simple and regular) based on Chen's algorithm. The row/column approach requires three steps: eight 1-D DCT/IDCTs along the rows, a memory transposition, and another eight 1-D DCT/IDCTs along the transposed columns. A block diagram of the baseline architecture for the 2-D DCT/IDCT block is shown in Fig. 2.

The controller enables input of the first row of data (DIN) through the ser2par unit under the SEN signal. It then activates the 1-D DCT/IDCT unit with the SEL and REN signals determining the data path. The first row of the transposition memory stores the results under ROWACK enabled. This process repeats for the remaining seven rows of the input block. Next, the ISEL and COLACK signals enable the 1-D DCT/IDCT unit to receive input data from the columns of the transposition memory. The results (DOUT) of the column-wise 1-D DCT/IDCT are available through the par2ser unit under PEN enabled.

The 1-D DCT/IDCT includes two multipliers, two adders, and two subtractors. The multipliers are array multipliers, which are fast and readily available library components that consume only 5% of the total power dissipation in the baseline design. To conform to IEEE 1180–1990 accuracy specifications, the multiplier constants in Chen's algorithm require a 12-bit representation. The DCT uses 16 internal registers to store intermediate values, whereas the IDCT requires 15 internal registers to store intermediate values. The internal registers have 14-bit internal width, which is necessary to avoid overflow. The arithmetic units and registers use multiplexers to select inputs from internal and external registers. With these resources, a 1-D DCT/IDCT operation completes in 16 clock cycles, and the overall 2-D DCT/IDCT process concludes in 392 clock cycles.

The transposition memory is a bank of 64 registers that holds the intermediate values from the first eight 1-D DCT/IDCTs.

The transposition memory receives inputs in a row-wise fashion and provides outputs in a column-wise fashion, thus performing a matrix transposition. Each row of the transposition memory is enabled for input from the 1-D DCT/IDCT unit after the first eight 1-D DCT/IDCTs. For the next eight 1-D DCT/IDCTs, the columns of the transposition memory output their data to the 1-D DCT/IDCT unit.

The ser2par unit in Fig. 2 converts serial input to parallel input for the 1-D DCT/IDCT unit. Prior to each of the first eight 1-D DCT/IDCTs, the ser2par shifts in eight values through its eight serially connected registers. The ser2par registers are 9 bits wide for the DCT and 12 bits wide for the IDCT. The par2ser unit is active after each of the last eight 1-D DCT/IDCT operations. It latches the results of each 1-D DCT/IDCT into its eight registers and shifts the results out serially. The par2ser unit is 12 bits wide for the DCT and 9 bits wide for the IDCT.

IV. PROPOSED LOW POWER DCT/IDCT DESIGN

Previous low power DCT/IDCT designs concentrate on a single low power design technique; we examine the individual and cumulative effects of several low power design techniques. Our techniques modify only the register transfer level (RTL) code—not processing techniques or standard cell libraries, which generally save power on most circuits. For consideration in the low power design, a low power technique must not significantly degrade the performance of the circuit (in area and speed) or the picture quality (in terms of PSNR or subjective metrics).

Before considering the impact of a low power design method on circuit performance, we ensure that the method does not significantly degrade picture quality. A method is considered for our design only if the degradation of the picture quality is unnoticeable to human eyes and the degradation of the PSNR is small. For the PSNR measurement and the visual examination, we use a prototype H.263 codec implemented in the C language. Three video clips—Claire, Miss America, and Foreman—are commonly used for such benchmarking. Claire and Miss America have little motion with a stable camera, whereas Foreman has frequent motion and camera movement. The three video clips are in QCIF format (176×144 pixels per frame), which is a small size suitable for a wireless device.

The following low power design techniques are candidates for our design: skipping input macroblocks for the DCT unit, skipping input blocks for the IDCT unit, gating the clock for disabled registers, using constant shift-and-add multipliers with reduced precision, and reducing transitions in the data path.

A. Skipping Input Macroblocks for the DCT Unit

Many input macroblocks, after motion compensation, contain little new information; this results in small magnitude DCT coefficients that are likely quantized to zero. The DCT (and the quantization operation) can be skipped for such macroblocks, and all DCT coefficients are set to zero as an approximation. In fact, this method is suggested to speed up DCT operations in software [30], [31], and we propose employment of this method to save power in hardware by disabling the DCT unit. To disable

TABLE I
THRESHOLD VERSUS SKIPPED MACROBLOCKS AND PSNR DEGRADATION

THRESH-HOLD	Claire		Foreman		Miss Am.	
	PSNR	Skipped MB's	PSNR	Skipped MB's	PSNR	Skipped MB's
0	40.53	0%	29.64	0%	40.52	0%
16	40.53	0.70%	29.64	0.50%	40.51	3.3%
32	40.51	8.30%	29.65	2.60%	40.44	23.8%
64	40.39	53.9%	29.56	15.5%	40.20	51.4%
128	39.41	79.8%	29.41	57.8%	40.08	76.4%
256	38.12	91.4%	29.40	85.7%	38.13	91.7%
512	36.72	96.1%	29.40	91.2%	37.77	95.9%

the DCT unit, the control unit should gate the clock signal, deactivate the enable signal, and activate the reset signal. In the disabled state, the DCT produces output with all coefficients equal to zero, which means the current block matches the candidate block from the previous frame. In typical video sequences, such matches occur often.

For efficient power savings, the method of predicting macroblocks with all output coefficients equal to zero should be simple and quick. The method in [31] requires complex calculations, as it calculates the DC coefficient. A better method of predicting macroblocks with all output coefficients equal to zero is to examine the SAD value of incoming macroblocks [30]. The SAD value provides a good measure of the energy of the incoming pixels and is readily available from the motion estimation unit. Macroblocks with low SAD values tend to produce little new information and are more likely to be quantized to zero.

Since a higher quantization parameter, QUANT, forces more coefficients to zero, it is better to consider the quantization parameter along with the SAD value. A macroblock is likely to result in output with all coefficients equal to zero if it has a low SAD value and a high QUANT parameter. Therefore, we propose skipping macroblock i if

$$SAD_i < THRESHOLD \times QUANT_i. \qquad (5)$$

To avoid adding a multiplier to the circuit, the THRESHOLD value is limited to powers of two, so the baseline unit will require only an additional comparator and clock gating circuitry. Using the prototype system, we examine the effect of different values of THRESHOLD on PSNR and on the number of skipped macroblocks (skipped MBs). Table I shows the results.

Higher thresholds result in more skipped blocks at the expense of PSNR degradation. Fewer macroblocks are skipped for "Foreman," which contains more motion and, hence, higher SAD values relative to QUANT. For all three video clips, we observe no noticeable degradation in video quality up to $THRESHOLD = 128$. Since this threshold also skips a large percentage of blocks, we examine the effects of this technique with $THRESHOLD = 128$.

B. Skipping Input Blocks for the IDCT Block

Because many DCT coefficients are quantized to zero, many IDCT input blocks have all coefficients equal to zero. Since an IDCT results in output with all coefficients equal to zero for

TABLE II
PERCENTAGE OF ALL-ZERO IDCT INPUT BLOCKS

	Claire	Foreman	Miss Am.
All-Zero Input Blocks	93.8%	93.4%	92.9%

TABLE III
FLIP-FLOP PERCENTAGE AND ACTIVITY FOR EACH UNIT

Unit	DCT		IDCT	
	Flip-Flops	Cycles Active	Flip-Flops	Cycles Active
Trans. Memory	69.3%	2.04%	70.0%	2.04%
1-D Unit	17.3%	31.7%	16.4%	32.2%
I/O Registers	13.0%	16.3%	13.1%	16.3%

such an input block, many software implementations skip computation of the IDCT. For our design, we propose disabling the IDCT for input data blocks with all coefficients equal to zero. To disable the IDCT unit, the control unit gates the clock signal, deactivates the enable signal, and activates the reset signal. In the disabled state, the IDCT unit produces output with all coefficients equal to zero.

The H.263 prototype system provides a measurement of the percentage of IDCT input blocks with all coefficients equal to zero. Table II shows that a significant number of input blocks have all coefficients equal to zero. Note that the IDCT skips a larger percentage of blocks than the DCT, since it decides to skip at the block level (instead of the macroblock level for DCT). A nonzero coefficient in a block causes evaluation of the block, not the entire macroblock. Additionally, the DCT produces macroblocks with all coefficients equal to zero that are not predicted from their SAD values.

In the decoder, an input block has all input coefficients equal to zero if both the CBP field and the DC coefficient are zero. An entire macroblock will have all input coefficients equal to zero if the COD bit is "0". The decoder can extract these parameters from the H.263 bit-stream. In the encoder, the VLC can emit a signal for a block with all coefficients equal to zero (that has a run of 64 coefficients equal to zero). The additional circuitry to check input blocks and to disable the IDCT unit should not add significant additional power, gates, or delay. Since a considerable number of IDCT calculations are skipped, we explore the consequences of disabling the IDCT for input data blocks with all coefficients equal to zero. Note that this method does not degrade the picture quality.

C. Gated Registers

Four units (the two I/O units, the 1-D DCT/IDCT unit, and the transposition memory) of the baseline DCT/IDCT block in Fig. 2 contain over 99% of the flip-flops in the circuit. Since registers comprise the majority of power dissipation in the circuit, disabling these registers can save power during periods of inactivity (i.e., clock cycles when there is no possibility of loading a new value). Disabling these registers requires gating the clock signal and deactivating the enable signal. The values in these registers should be preserved, so the reset signal remains deactivated.

Table III shows the percentage of flip-flops each unit contributes to the baseline design and the percentage of cycles that these flip-flops are active in the baseline design. The results are similar for both the DCT and the IDCT.

Since it contains the most flip-flops and is active for the least amount of time, the transposition memory presents the best opportunity for power reduction. In the transposition memory, the registers need updates only at the end of each of the first eight 1-D DCT/IDCT operations. As a result, the transposition registers require a clock in less than 3% of the cycles. The I/O units and the 1-D DCT/IDCT unit also show potential for significant power reduction; these units are not as promising as the transposition memory, since both contain fewer flip-flops and are active for a greater proportion of time. The registers in the baseline DCT/IDCT block are active for less than one-third of the entire 2-D DCT/IDCT operation. For the remaining time, these registers either store values for future arithmetic operations or stand idle. The input registers in the ser2par unit require a clock for each of the 64 input values, and the output registers in the par2ser unit require a clock for each of the 64 output values. Each I/O register needs a clock in just 16% of the total cycles. All four units contain registers that are inactive for the majority of the 2-D DCT/IDCT cycle, and disabling these registers has no effect PSNR. We examine the efficacy of gating these registers during the DCT/IDCT operation.

D. Constant Shift-Add Multipliers With Reduced Precision

The baseline design employs array multipliers, since array multipliers contribute only 5% of total power dissipation. However, if the above proposed methods achieve a large power reduction, array multipliers may dissipate a more significant proportion of power. Hence, it may be desirable to employ more power efficient multipliers at the cost of higher circuit complexity.

Several low power DCT/IDCT designs report that shift-and-add multipliers dissipate less power than array multipliers [8], [13], [27]. In addition, the shift-and-add multipliers require the least modification to the baseline design. The power dissipation of the shift-and-add multipliers is further reduced with common subexpression sharing. In common subexpression sharing, the various adders share common multiples of the fixed multiplicand.

A major concern for implementing the multipliers is the precision of the constant coefficients in Chen's algorithm. The precision of the coefficients for the baseline DCT/IDCT block shown in Fig. 2 is 12 bits, which is a necessary to conform to the IEEE Standard 1180–1990. This standard prevents an encoder and decoder with different IDCT architectures from producing dissimilar reference frames. Once initiated, the error between reference frames accumulates and propagates to subsequent frames, adversely affecting both PSNR and perceived quality. We observe that a system with identical IDCT architectures in the encoder and the decoder will produce identical IDCT reference frames. A single manufacturer is likely to make such a system for our target applications (cellular videophones, wireless surveillance systems, and mobile patrols). Since this system guarantees avoiding the error propagation apropos IEEE Std. 1180–1990, PSNR and perceived quality become

TABLE IV
PSNR Degradation for Different Coefficient Precisions

	Claire		Foreman		Miss Am.	
	DCT PSNR (dB)	IDCT PSNR (dB)	DCT PSNR (dB)	IDCT PSNR (dB)	DCT PSNR (dB)	IDCT PSNR (dB)
Baseline	40.54	40.58	29.64	29.64	40.52	40.54
10 bits	40.52	40.51	29.61	29.62	40.52	40.49
8 bits	40.12	40.17	28.97	29.14	40.26	40.26
6 bits	29.72	30.57	27.53	27.62	31.72	32.29

the salient factors determining the bit width. Additionally, for low bit rates, the quantization parameter—and hence the quantization noise—is usually large enough to mask small errors introduced by lower precision constants. Table IV shows the effects on PSNR of reduced precision constants in the IDCT and DCT units.

To preserve video quality, the DCT and IDCT use coefficients that are 8 bits wide. Each constant value in Chen's algorithm requires a shift-and-add multiplier, so the total number of multipliers increases from two to seven. However, the shift-and-add multipliers have a reduced number of adder rows and a reduced width. We examine the overall effect of shift-and-add multipliers on gate count, timing, and power.

E. Low Transition Data Path

Each register and arithmetic unit in the 1-D DCT/IDCT unit selects an input from multiple sources through a multiplexer. In the baseline unit, the select inputs (SEL in Fig. 2) are in a "don't care" state while a register or arithmetic unit is inactive. For low power design, it is desirable that each SEL signal remains unchanged from its previous value until the corresponding register or arithmetic unit is active. With fewer transitions on the SEL signals, fewer transitions will occur on the inputs of the registers and arithmetic units. The reduction in data path transitions should reduce power dissipation. Note that this method produces no reduction in PSNR, and it should increase the complexity of the control unit slightly, so we investigate the effectiveness of this method.

V. Experimental Results

First, we test each low power technique independently on the baseline units. Then, starting with the most effective method—which reduces the most power without significantly affecting gate count and timing—we apply all methods to the baseline units, one at a time. Synopsys Design Compiler estimates the gate-level power dissipation, gate count, and timing parameters for the three test video clips: Claire, Miss America, and Foreman. The QCIF video sequences require 594 DCT/IDCT operations per frame. At 10 frames/s, the circuits (which complete a DCT/IDCT operation in 392 clock cycles) require a minimum clock rate of 2.33 MHz. For power estimation, the synthesized circuits utilize a 0.18 μm TSMC standard cell library with a conservative wire load model and an input clock of 2.5 MHz.

TABLE V
Efficiency of DCT Low Power Methods When Applied Individually

	Claire		Foreman		Miss Am.	
Configuration	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced
Baseline	701		779		729	
Skip MB's	190	72.9%	412	47.1%	241	66.9%
Gated Registers						
Trans. Mem.	310	55.7%	350	55.1%	325	55.4%
1-D DCT	601	14.1%	681	12.6%	637	12.7%
I/O Registers	625	10.8%	705	9.51%	661	9.44%
Multipliers	622	11.3%	704	9.59%	660	9.53%
Data Path	636	9.29%	717	7.92%	672	7.88%

TABLE VI
Efficiency of IDCT Low Power Methods When Applied Individually

	Claire		Foreman		Miss Am.	
Configuration	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced
Baseline	559		540		549	
Skip Blocks	83.9	85.0%	90.8	83.2%	91.7	83.3%
Gated Registers						
Trans. Mem.	234	58.1%	210	61.3%	219	60.1%
1-D DCT	487	13.0%	469	13.1%	477	13.1%
I/O Registers	495	11.5%	480	11.1%	487	11.3%
Multipliers	502	10.2%	486	9.96%	491	10.6%
Data Path	504	9.92%	491	9.10%	496	9.75%

No individual method increases gate count more than 2%, and the combination of methods decreases net gate count due to the reduced multiplier size. Additionally, no method or combination of methods degrades the timing to (anywhere near) the point that the circuit could not operate at 10 frames/s. Because the low power techniques do not significantly affect area and timing, the following results concentrate on power reduction.

Tables V and VI show the effects of each method when independently added to the baseline DCT and IDCT units.

From Table V, the most efficient method for the DCT is the skipping of macroblocks with low SAD parameters, which reduces power dissipation by an average of 62.3%. The method is effective because it skips 57.8% to 79.8% of macroblocks. The "Foreman" sequence shows less power reduction than other sequences because the high amount of motion causes higher SAD values relative to the QUANT parameter. As more motion is introduced into this video sequence, the QUANT parameter reaches its maximum value before the SAD value levels off. Hence, our method skips a smaller percentage of blocks in the high motion sequence.

From Table VI, the most efficient method for the IDCT is the skipping of input blocks with all coefficients equal to zero. The power savings is larger than the DCT because the decision to skip is made at the block level, not at the macroblock level. Additionally, the DCT may produce macroblocks with all coefficients equal to zero that are not predicted by their SAD values. In the IDCT unit, the power savings from skipping input blocks with all coefficients equal to zero is about the same for

TABLE VII
EFFICIENCY OF DCT LOW POWER METHODS WHEN ADDED INCREMENTALLY

	Claire		Foreman		Miss Am.	
Configuration	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced
Baseline	701		779		729	
Skip MB's	190	72.9%	412	47.1%	214	66.9%
Gated Registers						
Trans. Mem.	142	79.7%	248	68.1%	184	74.8%
1-D DCT	108	84.6%	156	80.0%	117	83.9%
I/O Registers	72.2	89.7%	115	85.2%	80.2	89.0%
Multipliers	50.5	92.8%	93.5	88.0%	57.6	92.1%
Data Path	42.7	93.9%	84.6	89.1%	48.7	93.3%
Combined	42.7	93.9%	84.6	89.1%	48.7	93.3%

TABLE VIII
EFFICIENCY OF IDCT LOW POWER METHODS WHEN ADDED INCREMENTALLY

	Claire		Foreman		Miss Am.	
Configuration	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced	Power (μW)	Percent Reduced
Baseline	559		540		549	
Skip Blocks	83.9	85.0%	90.8	83.2%	91.7	83.3%
Gated Registers						
Trans. Mem.	55.4	90.1%	60.5	88.8%	60.9	88.9%
1-D DCT	29.7	94.7%	35.2	93.5%	34.0	93.8%
I/O Registers	23.0	95.9%	28.1	94.8%	27.4	95.0%
Multipliers	16.8	97.0%	22.7	95.8%	22.5	95.9%
Data Path	15.7	97.2%	19.1	96.3%	19.1	96.5%
Combined	15.7	97.2%	19.1	96.3%	19.1	96.5%

the three video clips. Although the high motion video sequence "Foreman" produces larger magnitude coefficients, the quantization unit forces just as many coefficients to zero as in the low motion video sequences.

The next most efficient method, which has a similar impact on both the DCT and the IDCT, is the gating of registers. Registers account for a large proportion of the power dissipation in the DCT and IDCT units. Since registers need only be enabled when they have meaningful input, gating registers saves a significant amount of power. The greatest power savings from clock gating comes in the transposition memory. The transposition memory accounts for about 70% of the total flip-flops in the DCT or IDCT circuit, and it can be disabled for a large proportion of the 2-D cycle. The registers in the 1-D unit and in the I/O unit are active for a larger proportion of time than the transposition memory, and they each account for approximately the same amount of the remaining flip flops.

For both the DCT and the IDCT, smaller, yet significant improvements in power dissipation result from the low power multipliers and the low transition data path.

The five methods are then added to the baseline unit in order from the most efficient to the least efficient. Skipping blocks and macroblocks are considered first, followed by the clock gating techniques (starting on the unit with the most power savings). Finally, the low power multipliers and the low transition data path are added. For the DCT and IDCT blocks, Tables VII and VIII show the experimental results under the employment of the methods.

We observe from Tables VII and VIII that each power savings technique has an impact, even after others are added. For both the DCT and the IDCT, the combination of power savings methods impacts the overall power far more than any single method. The average power reduction of the proposed methods for the DCT and IDCT is 92.1% and 96.6%.

Skipping DCT macroblocks and lowering the multiplier precision both degrade picture quality. Table IX shows the effects of these methods on PSNR. The "Foreman" sequence exhibits less degradation than other sequences due to the larger proportion of quantization noise caused by its large amount of motion. Note that the combined methods degrade PSNR more than individual methods. However, the degradation is unnoticeable to

human eyes as shown in a stimulus comparison test between baseline and low power sequence pairs [32], [33].

It is difficult to compare the power efficiency of different DCT/IDCT designs due to differences in supply voltages, operating frequencies, throughput rates and processes. Reference [13] proposes scaling formulas for the purpose of comparing power dissipation between different DCT designs. The scaling formulas are as follows:

$$S = \frac{W}{W'}$$
$$U = \frac{V}{V'}$$
$$P' = P \times \frac{S}{U^3}$$
$$f' = f \times \frac{S^2}{U}$$
$$PP' = P' \times \frac{f_{ref}}{f'} \qquad (6)$$

where
W' reference process;
W process;
V' reference supply voltage;
V supply voltage;
P power;
P' power after scaling;
f' frequency after scaling;
f_{ref} reference frequency;
f frequency;
S process scaling factor;
U voltage scaling factor;
P' power for fixed voltage and process;
PP' power for fixed throughput.

Using the above scaling formulas, we compare the power efficiency of our designs to previous designs that target applications with similar throughput and accuracy requirements (JPEG, H.261, H.263, MPEG, and MPEG-2). The most demanding throughput requirement is MPEG-2 MP@ML, which requires a throughput of 14 million samples/s. Our maximum throughputs are 29.4 million samples/s for DCT and 25.1 million samples/s for IDCT. Although the comparison may not be fair due to the imprecision of scaling, it is a good indication

TABLE IX
PSNR DEGRADATION OF PROPOSED LOW POWER METHODS

Configuration	Claire PSNR (dB)	Foreman PSNR (dB)	Miss Am. PSNR (dB)
Baseline	40.53	29.64	40.52
Low Precision Alone	39.19	29.33	39.46
Skip MB's Alone	39.41	29.41	40.08
Combined	38.51	29.12	39.10

TABLE X
SCALED COMPARISON OF DCT AND IDCT ARCHITECTURES

Architecture	Scaled, Normalized, Power	Supply Voltage (V)	Process Technology (μm)
[11]	0.540	1.56	0.6
[13]	0.801	1.50	1.2
[26]	1.00	1.30	0.5
Proposed DCT	0.0997	1.80	0.18
Proposed IDCT	0.0321	1.80	0.18

TABLE XI
CHARACTERISTICS OF LOW POWER DCT AND IDCT ARCHITECTURES

	Gate Count (NAND2)	Max Frequency (MHz)	Max Throughput (Msamples/s)	Process Technology (μm)	Supply Voltage (V)
DCT	14 706	180	29.4	0.18	1.8
IDCT	14 805	154	25.1	0.18	1.8

on the standing of the proposed methods. Table X shows that our methods perform better than other methods, mostly by an order of magnitude.

VI. CONCLUSION

This paper presents four effective low power techniques for 2-D DCT and IDCT blocks, which are intended for low bit rate wireless video applications. The most efficient scheme for the DCT is skipping macroblocks; low motion video sequences save more power than the high motion video sequences. The remaining three techniques result in similar power savings for high and low motion video sequences. For the DCT unit, the average power saving for the four methods combined is 92%. The most efficient scheme for the IDCT block is skipping input data blocks with all coefficients equal to zero, which saves a similar amount of power for all three sequences. The average power savings for the four methods combined is 97% in the IDCT unit. For both the DCT and IDCT, the combined methods reduce much more power than any individual method

Finally, it is important to note that our methods can be integrated with other methods such as adaptive precision [11] and zero-valued coefficients [11], [16], [27] to further reduce power. Some more general methods such as using low power libraries or lowering the supply voltage can also reduce additional power. For more efficient power savings in high motion sequences, further experiments may justify using a higher THRESHOLD for higher SAD values.

We conclude the paper by summarizing the characteristics of our DCT and IDCT designs in Table XI.

REFERENCES

[1] "Video Coding for Low-Bitrate Communication, Draft International Standard,", ITU-T Rec. H.263, 1998.
[2] B. Erol, F. Kossentini, and H. Alnuweiri, "Implementation of a fast H.263+ encoder/decoder," in *Conf. Rec. Thirty-Second Asilomar Conference on Signals, Systems & Computers*, vol. 1, Sept. 1998, pp. 462–466.
[3] W. H. Chen, C. H. Smith, and S. C. Fralick, "A fast computational algorithm for the discrete cosine transform," *IEEE Trans. Commun.*, vol. COM-25, pp. 1004–1009, Sep. 1977.
[4] B. G. Lee, "A new algorithm to compute the discrete cosine transform," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 1243–1245, Dec. 1984.
[5] N. I. Cho and S. U. Lee, "Fast algorithm and implementation of 2-D discrete cosine transform," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 297–305, Mar. 1991.
[6] S.-F. Hsiao, W.-R. Shiue, and J.-M. Tseng, "A cost-efficient and fully-pipelinable architecture for DCT/IDCT," *IEEE Trans. Consumer Electron.*, vol. 45, pp. 515–525, Aug. 1999.
[7] K. Kim and J.-S. Koh, "An area efficient DCT architecture for MPEG-2 video encoder," *IEEE Trans. Consumer Electron.*, vol. 45, pp. 62–67, Feb. 1999.
[8] A. Madisetti and A. N. Willson, Jr., "A 100 MHz 2-D 8 × 8 DCT/IDCT processor for HDTV applications," *IEEE Trans. Circuits Systems Video Technol.*, vol. 5, pp. 158–165, Apr. 1995.
[9] I.-K. Kim, J.-J. Cha, and H.-J. Cho, "A design of 2-D DCT/IDCT for real-time video applications," in *Proc. 6th Int. Conf. VLSI and CAD*, Oct. 1999, pp. 557–559.
[10] C. V. Schimpfle, P. Rieder, and J. A. Nossek, "A power efficient implementation of the discrete cosine transformation," in *Conf. Rec. Thirty-First Asilomar Conference on Signals, Systems & Computers*, vol. 1, Nov. 1997, pp. 729–733.
[11] T. Xanthopoulos and A. P. Chandrakasan, "A low power DCT core using adaptive bitwidth and arithmetic activity exploiting signal correlations and quantization," *IEEE J. Solid State Cricuits*, vol. 35, pp. 740–750, May 2000.
[12] Y.-F. Jang, J.-N. Kao, J.-S. Yang, and P.-C. Huang, "A 0.8μ 100 MHz 2-D DCT core processor," *IEEE Trans. Consumer Electron.*, vol. 40, pp. 703–710, Aug. 1994.
[13] J. Li and S.-L. Lu, "Low power design of two-dimensional DCT," in *Proc. Ninth Annu. IEEE Int. ASIC Conf. Exh.*, Sept. 1996, pp. 309–312.
[14] H. B. Kim, "High-Level Synthesis and Implementation of Built-in Self-Testable Data Path Intensive Circuit," Ph.D. dissertation, EE Dept., Virginia Polytech. Inst. State Univ., Blacksburg, 1999.
[15] E. Feig and S. Winograd, "Fast algorithms for the discrete cosine transform," *IEEE Trans. Signal Processing*, vol. 40, pp. 2174–2193, Sept. 1992.
[16] E. Scopa, A. Leone, R. Guerrieri, and G. Baccarani, "A 2-D DCT low-power architecture for H.261 coders," in *Proc. 1995 ICASSP*, vol. 5, May 1995, pp. 3271–3274.
[17] Y.-P. Lee, T.-H. Chen, L.-G. Chen, M.-J. Chen, and C.-W. Ku, "A cost-effective architecture for 8 × 8 two-dimensional DCT/IDCT using direct method," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 459–467, June 1997.
[18] L.-G. Chen, J.-Y. Jiu, H.-C. Chang, Y.-P. Lee, and C.-W. Ku, "A low power 2-D DCT chip design using direct 2-D algorithm," in *Proc. ASP-DAC '98. Asia and South Pacific Design Automation Conf.*, Feb. 1998, pp. 145–150.
[19] K.-W. Shin, H.-W. Jeon, and Y.-S. Kang, "An efficient VLSI implementation of vector-radix 2-D DCT using mesh-connected 2-D array," in *Proc. 1994 IEEE Int. Symp. Circuits and Systems, 1994*, vol. 4, June 1994, pp. 47–50.
[20] Y.-M. Huang, J.-L. Lu, and C.-T. Hsu, "A refined fast 2-D discrete cosine transform algorithm with regular butterfly structure," *IEEE Trans. Consumer Electron.*, vol. 44, pp. 376–383, May 1998.
[21] T.-S. Chang, C.-S. Kung, and C.-W. Jen, "A simple processor core design for DCT/IDCT," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 439–447, Apr. 2000.
[22] J. Chen and K. J. R. Liu, "Cost-Effective low-power architectures of video coding systems," in *Proc. 1999 IEEE Int. Symp. Circuits and Systems*, vol. 1, June 1999, pp. 153–156.
[23] ——, "Low-Power architectures for compressed domain video coding co-processor," *IEEE Trans. Multimedia*, vol. 2, pp. 111–128, June 2000.
[24] V. Srinivasan and K. J. R. Liu, "VLSI design of high-speed time-recursive 2-D DCT/IDCT processor for video applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 87–96, Feb. 1996.

[25] S.-F. Hsiao and W.-R. Shiue, "New hardware-efficient algorithm and architecture for the computation of 2-D DCT on a linear systolic array," in *Proc. 1999 IEEE ICASSP*, vol. 6, Mar. 1999, pp. 3517–3520.

[26] T. Xanthopolous and A. P. Chandrakasan, "A low-power IDCT macrocell for MPEG-2 MP@ML exploiting data distribution properties for minimal activity," *IEEE J. Solid State Circuits*, vol. 34, pp. 693–703, May 1999.

[27] K. Kim and P. A. Beerel, "A high-performance low-power asynchronous matrix vector multiplier for discrete cosine transform," in *Proc. First IEEE Asia Pacific Conf. ASIC's, 1999*, Aug. 1999, pp. 135–138.

[28] M. Kuhlmann and K. K. Parhi, "Power comparison of flow-graph and distributed arithmetic based DCT architectures," in *Conf. Rec. Thirty-Second Asilomar Conference on Signals, Systems & Computers*, vol. 2, Nov. 1998, pp. 1214–1219.

[29] J.-S. Chiang and H.-C. Huang, "New architecture for high throughput-rate real-time 2-D DCT and the VLSI design," in *Proc. Ninth Annu. IEEE Int. ASIC Conf. Exh.*, Sep. 1996, pp. 219–222.

[30] I.-M. Pao and M.-T. Sun, "Modeling DCT coefficients for fast video encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 608–616, June 1999.

[31] A. Yu, R. Lee, and M. Flynn, "Performance enhancement of H.263 encoder based on zero coefficient prediction," in *Proc. ACM Conf. Multimedia '97*, Seattle, WA, Nov. 9–13, 1997, pp. 21–29.

[32] "Methodology for the Subjective Assessment of the Quality of Television Pictures,", ITU-R Rec. ITU-R BT.500–10, 1998.

[33] "Subjective Video Quality Assessment Methods for Multimedia Applications,", ITU-T Rec. P.910, 1999.

Nathaniel J. August was born in Maryland in 1975. He received the B.S. degree in computer engineering in 1998 and the M.S. degree in electrical engineering in 2001 from Virginia Tech, Blacksburg, where his research interests included low power VLSI design and signal processing. He is currently pursuing the Ph.D. degree at Virginia Tech as a Bradley Fellow and a Cunningham Fellow.

In between degrees, he was a Validation Eengineer for Intel Corporation, Portland, OR and Folsom, CA, on various projects including pre-silicon validation of gigabit Ethernet adapters and post-silicon validation of PCI chipsets. His current research interests include low power VLSI design for ultra wideband (UWB) systems in applications such as wireless personal area networks (WPANs), radio frequency identification (RFID), and wireless ad hoc and sensor networks.

Dong Sam Ha (M'86–SM'97) received the B.S. degree in electrical engineering from Seoul National University (SNU), Seoul, Korea, in 1974, and the M.S. and Ph.D. degrees in electrical engineering from the University of Iowa, Iowa City, in 1984 and 1986, respectively.

Since fall 1986, he has been a faculty member of the Bradley Department of Electrical Engineering, Virginia Polytechnic Institute and State University (Virginia Tech), Blacksburg. Currently, he is Professor with the department. Prior to his graduate study, he was a Research Engineer for the Agency for Defense Development in Korea from 1975 to 1979. While on leave from May to December of 1996, he was with the Semiconductor Research Center, SNU, where he investigated built-in self-test synthesis. Along with his students, he developed four computer-aided design tools for digital circuit testing. The source code for these four tools has been distributed to over 160 universities and research institutions worldwide, and the tools have been used for various research and teaching purposes at those universities. His current research interests include ultra wideband system design, low-power VLSI design for 3G/4G wireless communications, low-power/high speed analog and mixed-signal VLSI design, and low-power VLSI system design for wireless video.

Dr. Ha was Technical Program Chair of ASIC/SOC Conference, 2003.