Abstract—This paper proposes a novel two-dimension (2-D) discrete cosine transform (DCT) design by using the low-cost residue/quotient (LCRQ) codes. Based on the LCRQ codes, the proposed method divides the 2-D DCT into residue-based module and quotient-based module for realizing the DCT computations. Performance evaluation shows that the proposed LCRQ-based 2-D DCT realization can speed up the traditional DCT computation to 76.4% with less hardware design requirement.

I. INTRODUCTION

DCT [1] is widely used for image processing and video signal coding due to its decorrection and energy concentration properties [2]. In other words, DCT plays a key role of image and video compression such as JPEG (Joint Photographic Experts Group), H.26x and MPEG (Motion Pictures Expert Group). Therefore, the design of DCT for processing efficiency is a very important issue. A residue number system (RNS) based DCT implementation for computation acceleration was presented recently [3]. Although, the RNS-based DCT design is a powerful solution for maintaining constant performance whereas increasing precision, the implementation cost (area overhead) is approximately double than the traditional architecture. Therefore, this paper proposes a LCRQ-based DCT realization to overcome the design cost problem and increase the computation speed in DCT.

II. LCRQ CODES

A residue code is called a low-cost residue code with check base or modulo $A$ if and only if $A = 2^a - 1$, $a \geq 2$, where $a$ is the number of check bits [4]. Significantly, the residue codes, using modulus of the form $2^a - 1$, are a suitable choice in practical implementation. Additionally, the quotient codes are derived from the residue codes to support the functionality realization. Generally, the input data of a DCT can be expressed as,

$$X = \sum_{i=0}^{n-1} X_i (2^a)^i = X_0 + X_1 \times 2^a$$  \hspace{1cm} (2)

The Thus, the residue code of $X$ modulo $A$ is given by,

$$[X]_A = \sum_{i=0}^{n-1} X_i (2^a)^i \pmod{A} = [X_0 + X_1 \times 2^a]_A = [X_0]_A = X_0$$  \hspace{1cm} (3)

From (3), the quotient code can be written as,

$$[X / A] = ([X_0] / A) + ([X_1 \times 2^a] / A) = X_1$$  \hspace{1cm} (4)

Based on (3) and (4), a LCRQ DCT realization for computation speed acceleration and hardware cost reduction is presented in this paper.

III. LCRQ-BASED 2-D DCT REALIZATION

Fig. 1 shows the architecture design of the proposed LCRQ-based 2-D DCT realization. Four major components such as LCRQ generator, residue-based module, quotient-based module, and RQ combination are included of the proposed method. The LCRQ generator is used to generate the RQ codes from input data. The computations of DCT functionality are performed by using the residue-based and the quotient-based modules. The RQ combination takes charge of combining the separated data, residue and quotient, to restore the decimal data such as follows,

$$X = [X / A] \times A + [X]_A$$  \hspace{1cm} (5)

By using the proposed architecture design, a large dynamic range binary system can be partitioned into two faster parallel operations with small dynamic range in residue-based and quotient-based modules. Significantly, the carry data in the residue-based module has to be delivered to the quotient-based module to operate for avoid functionality error.

Fig. 1 The architecture of LCRQ-based 2-D DCT realization.
A 8×8 2-D DCT is implemented here to verify the feasibility of the proposed method. Generally, the 8×8 2-D DCT architecture can be realized by using a 1D-DCT processor and a transposition register. Fig. 2 illustrates the implementation of an 8-pixel residue-based (quotient-based) 1-D DCT processor using the LCRQ codes. From Fig. 2, the input data r(0)–r(7) from LCRQ generator are sent to some adders, subtractors, multipliers, and delay buffers for DCT functionality computation. There are 29 adders/subtractors and 5 multipliers are needed for the residue-based 1-D DCT operation. Additionally, the coefficients \( m_i \) of the multipliers are list as follows [5].

\[
m_1 = c_4, m_2 = c_6 - c_2, m_3 = c_4, m_4 = c_2 + c_6, m_5 = c_6 \quad (6)
\]

where \( c_i = \cos(i\pi/16) \).

It should be noted that the operation of quotient-based 1-D DCT is the same as the residue-based 1-D DCT. According to the architecture design in Fig. 1 and 1-D DCT implementation in Fig. 2, the proposed LCRQ 2-D DCT can be easily realized.

![Fig. 2 Residue-based and quotient-based 1-D DCT implementation.](image)

**IV. PERFORMANCE EVALUATION**

The proposed LCRQ-based 2-D DCT architecture has been realized by using the structure-level VHDL synthesis to analyze and evaluate the performance. From Fig. 1, the 16-bit dynamic range is required to support the realization of the proposed LCRQ-based 2-D DCT architecture. In other words, the modulo \( A \) is equal to 256 for implementing the residue-based and quotient-based modules. Additionally, according to Fig. 2 and (6), the 7-bit cosine coefficients and 8-bit signal samples are needed for realizing the residue-base and quotient-based 1-D DCT. Fig. 3 shows the comparison results of the proposed architecture and the traditional design [1]. Fig. 3 clearly indicates that the hardware implementation components (logic elements, LEs) of the proposed architecture and the design in [1] are 5476 and 6099, respectively. That is to say, about 10.2% LEs can be effectively reduced by using the proposed architecture design. On the other hand, the performance in speed can be evaluated with the circuit throughput, which is measured in millions of DCT per second (MDPS). In Fig. 3, the throughput of the proposed architecture is up to 76.4% over the traditional DCT design in [1]. Briefly, the proposed LCRQ-based 2-D DCT design has the good performance in easily implementation and high throughput.

**Fig. 3 Comparison results.**

**V. CONCLUSIONS**

This paper develops a novel 2-D DCT realization by using the LCRQ technique. Theoretical analysis and mathematical model of the LCRQ codes are carefully established for implementing the proposed architecture. Performance evaluation shows that the proposed architecture design can achieve higher throughput than the traditional design with less hardware implementation requirement. This is going to be an essential architecture design that makes hardware realization of DCT feasible.

**REFERENCES**


