A Bitplane Decomposition Matrix Based VLSI Integer Transform Architecture for HEVC

Abstract:

VLSI integer transform architecture is proposed for high efficient video coding (HEVC) encoder. The architecture is designed based on the signed bitplane transform (SBT) matrices which are derived from the bitplane decompositions of the integer transform matrices in HEVC. Mathematically, an integer transform matrix can be equally expressed by the binary weighted sum of several SBT matrices which are only composed of binary 0 or _1. The SBT matrices are very simple and have lower bitwidth than the original integer transform in the form. The SBT matrices are also sparse in which there are many zero elements. The sparse characteristic of SBT matrices is very helpful for saving the addition operators of SBT. In the proposed architecture, instead of the original integer transform in high bitwidth, the video data can be respectively transformed with the SBT matrices in lower bitwidth. As the result, the delay of transform unit circuit can be significantly reduced with the proposed SBT. Moreover, exploiting the redundant element characteristic of SBT matrices, in which the elements are 0 or _1, the adder reuse strategy is proposed for our transform architecture, which can save the circuit area efficiently. The simulation results show that, employing the proposed strategies, the VLSI transform architecture can be synthesized in a proper area with a high working frequency and low latency. The architecture can support all HEVC encoders coding Ultra HD video sequences in real time.

Existing system:

HEVC is the latest video coding with higher coding performance than other existing ones. Many novel coding algorithms are introduced in HEVC. Especially in the aspect of transform, up to 32x32 integer transform is applied for improving coding performance. The existing transform architectures consider how to reduce the number of arithmetic operators, such as addition and multiplication, more than the data bitwidth in transform. In fact, the data bitwidth is also an important factor impacting on the circuit speed and area of VLSI architecture. Circuit with large bitwidth needs larger number of fan-in or fanout of logic gate and more MOS devices are...
required in the logic gate circuit. Thus, the capacitive load and resistance of logic gate all increase with widening bitwidth.

**Proposed system:**

The transform architecture, instead of the integer transform matrix circuits, the SBT matrices circuits are implemented and the input data are transformed with each SBT matrix circuits respectively. Due to the simple elements of SBT matrices, the bitwidths of intermediate transformed data and output data are significantly reduced. The bitwidth of output data should be $n + d\log N_2 e$ maximally. Taking 32x32 1D integer transform as an example, the increasing bitwidth of output data is only 5 bits with SBT algorithm, comparing with the 11 bits increasing of straight forward integer transform. The bitwidths of SBT increase slow as the intermediate data processed stage by stage, which shortens circuit delay and constrains the clock cycle smaller. Although, the delay of integer transform circuit is reduced based on the proposed bit transform algorithm, more adders are required due to more SBTs. However, the bitwidths of adders used in SBT are also so low that the addition operation is also very fast.

**Applications:**

1) Digital signal processing

**Advantages**

1) High Speed,
2) Area and Delay reduced.

**System Configuration:**

In the hardware part a normal computer where Xilinx ISE 14.3 software can be easily operated is required, i.e., with a minimum system configuration

**HARDWARE REQUIREMENT**
Processor  - Pentium –III

Speed  - 1.1 GHz

RAM  - 1 GB (min)

Hard Disk  - 40 GB

Floppy Drive  - 1.44 MB

Key Board  - Standard Windows Keyboard

Mouse  - Two or Three Button Mouse

Monitor  - SVGA

SOFTWARE REQUIREMENTS


❖ Front End : Modelsim 6.3 for Debugging and Xilinx 14.3 for Synthesis and Hardware Implementation

❖ This software’s where Verilog source code can be used for design implementation.