

General Description

An entirely new kind of MPEG is being developed by the Joint Video Team (JVT), which consists of MPEG and VCEG (Video Coding Experts Group) members. It is known as H.264 or MPEG4, Part 10. This new standard will have an entirely different type of transform algorithm that will be explained here along with the implementation approach that PixSil Technology is taking for their IP and ASIC development. This IP block implements one portion of this new video compression algorithm, the inverse transform operation. There is a corresponding transform in the encoder called the "Forward Integer Transform". PixSil Technology's approach to hardware implementation of the set of H.264 algorithms is highly modular and scalable.

Applications

- High Quality Video, Low Bit-Rate, Low Power Applications
- Wireless Video
- Video Streaming
- Video Conferencing
- Video Surveillance

Features

- H.264 performance at Level 3.0, 2.0, and 1.1
- Pipelined Operation
- Three types of transforms
- 2, 4, or 8 clocks/ sample: luma DC transform
- 2, 4 or 8 clocks/ sample: residual transform
- 1, 2, 4 clocks/ sample: chroma DC transform
- pre and post-scaling
- Synchronous I/O interface
- 95 % utilization of internal hardware
- 16-bit processor data interface
- Simple hand-shaking interface

Functional Description

The 2-D IIT core implements the algorithm by employing row-column decomposition of the data and applying a 1-D transform on the data row-wise, and then on the row results, column-wise. First, the IIT structure of Figure 1 is applied on the data row-wise as it enters the core. Pre-scaling may be applied. The row results are then transposed in memory and the same structure is applied. The results are emitted sequentially on the output port of the core. Post-scaling may be applied. A dual-port memory is needed since data is simultaneously being written and read.

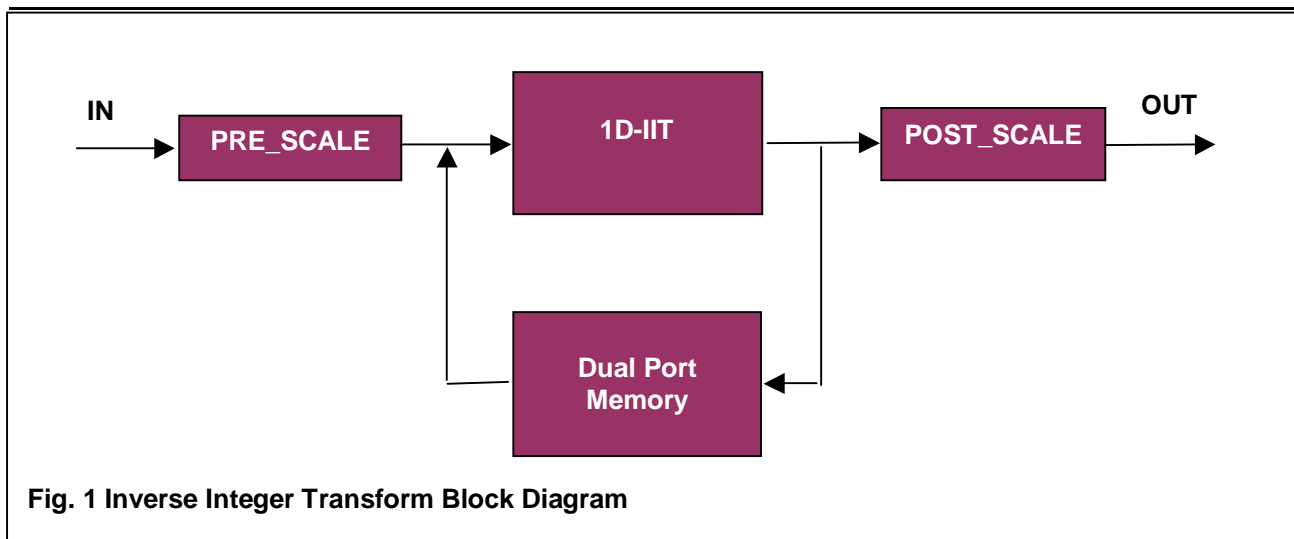


Fig. 1 Inverse Integer Transform Block Diagram

Let us look at the similarities between the current MPEG transform algorithm, 2-D Discrete Cosine Transform (DCT), and the new MPEG transform, Integer Transform (IT). Both of these inverse transforms convert the frequency components to a set of spatial components. Both of these transforms are in two-dimensions, and are separable, i.e. each row can be transformed first using a 1-D transform, and the resulting columns can then be transformed using the same 1-D transform.

$$f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} \\ c_{10} & c_{11} & c_{12} & c_{13} \\ c_{20} & c_{21} & c_{22} & c_{23} \\ c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

The differences are considerable. The Inverse DCT (IDCT) is defined for an 8 x 8 block of pixels, while the Inverse IT (IIT) defined as a 4 x 4 transform. The Inverse Integer Transform (IIT) approximates an IDCT using only integer values for the coefficients. This can simplify the hardware considerably. Also, there are no multiplications! Only additions and some simple shifts are required within the integer transform algorithm, which can be implemented easily by the digital logic. Also, the input values are scaled so that none of the computations within the transform algorithm will exceed 16-bit results.

Finally, the “residual transform” is identical to the 4 x 4 transform above with 16 additional right shift operations on intermediate results within the transform.

Inverse Transform Algorithm

The image data is actually composed of Luma (intensity) pixels and Chroma (color) pixels. These are further separated into DC and AC components. If we look in more detail about the IIT, there are actually three types of transforms to support the different types of pixels. Each of these transforms can be calculated using the same IP supplied by PixSil Technology.

PixSil Technology uses a pipelined architecture for the implementation of the transform. The data is sent into the block in row order. Internally, there is a pre-scaler, two ALUs, transpose memory, and a post-scaler. Data will emerge from the block in column order.

PixSil’s architecture enables each of the three different transforms to be calculated using the same hardware. If data is available to be transformed at the input, 100% of the block’s hardware can be utilized at each clock period, resulting in a very small and power-efficient implementation.

The first transform is “chroma DC transform”. This is the simplest transform since it is a 2 x 2 transform of the 4 chroma DC coefficients, shown as c_{ij} below. Calculation of the chroma DC transform requires 8 additions/subtractions.

Pre-Scaling

During a residual transform, the input data may be scaled by one of three values loaded into register CTL_REG3. A left shift is then performed on the resulting scaled data according to one of four values loaded in CTL_REG1. Please see the section on “Control Registers” for more information.

$$f = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Post-Scaling

During a residual transform, the output data may be scaled by one of three values loaded into register CTL_REG3. A left shift is then performed on the resulting scaled data according to one of four values loaded in CTL_REG1.

The next transform is called “luma DC transform”. This transform is a 4 x 4 transform of the 16 luma DC coefficients shown as c_{ij} below. Calculation of the luma DC transform requires 96 additions/subtractions, which can be reduced to 64 additions/subtractions in a two-stage process.

During a chroma DC transform, the output data may be scaled by one of three values loaded into register CTL_REG3. If QPY is greater than or equal to 6, a left shift is then performed on the resulting scaled data according to one of four values loaded in CTL_REG1. Otherwise, the result and then shifted to the right by 1.

During a luma_DC transform, the output data may be scaled by one of three values loaded into register CTL_REG3. If QPY is greater than or equal to 12, a left shift is then performed on the resulting scaled data according to one of four values loaded in CTL_REG1. Otherwise, 2 may be added to the result and then shifted to the right by 2MPY. Please see the section on "Control Registers" for more information on pre and post-scaling.

IIT used in H.264 Decoder

PixSil has designed its configurable hardware to be embedded within a System-on-chip (SOC). The processor performs most of the high-level frame management functions as well as sets the parameters for control of the data stream. DMA engines direct the data stream to the hardware block. Each block is selected to optimize the particular requirements of that level with the minimum amount of hardware. Clock speed may be varied for power minimization. PixSil Technology has designed the Integer Transform Block modules to meet the various throughput and I/O requirements for several of the different levels of H.264. The H.264 Inverse Integer Transform block may be integrated into the design of the full H.264 decoder as shown in Figure 3 or connected to the processor system bus for use as an accelerator. An SDRAM is the only external component required for PixSil Technology's FPGA/ASIC implementation of the decoder system.

IIT Control, Data, and Processor Signal Overview

There are three types of signals interfacing with the core: Data, Control and Processor. The Data input bus is 8 bits wide; Data output bus is 16 bits wide. The Control signals provide the handshaking for data flow control at input/output of the core; also indicating the start and type of transform that will be performed. The processor bus can load and read the internal registers, containing parameters or control information.

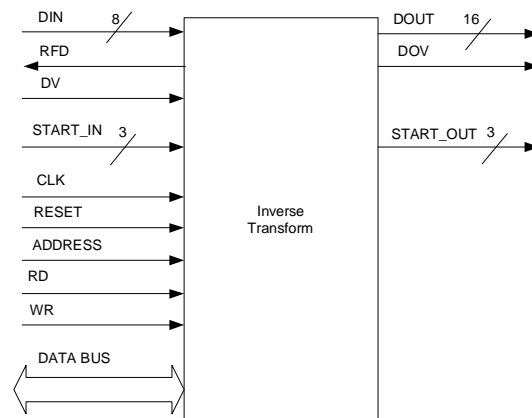


Fig 2. Inverse Integer Transform Core Diagram and Port Definitions

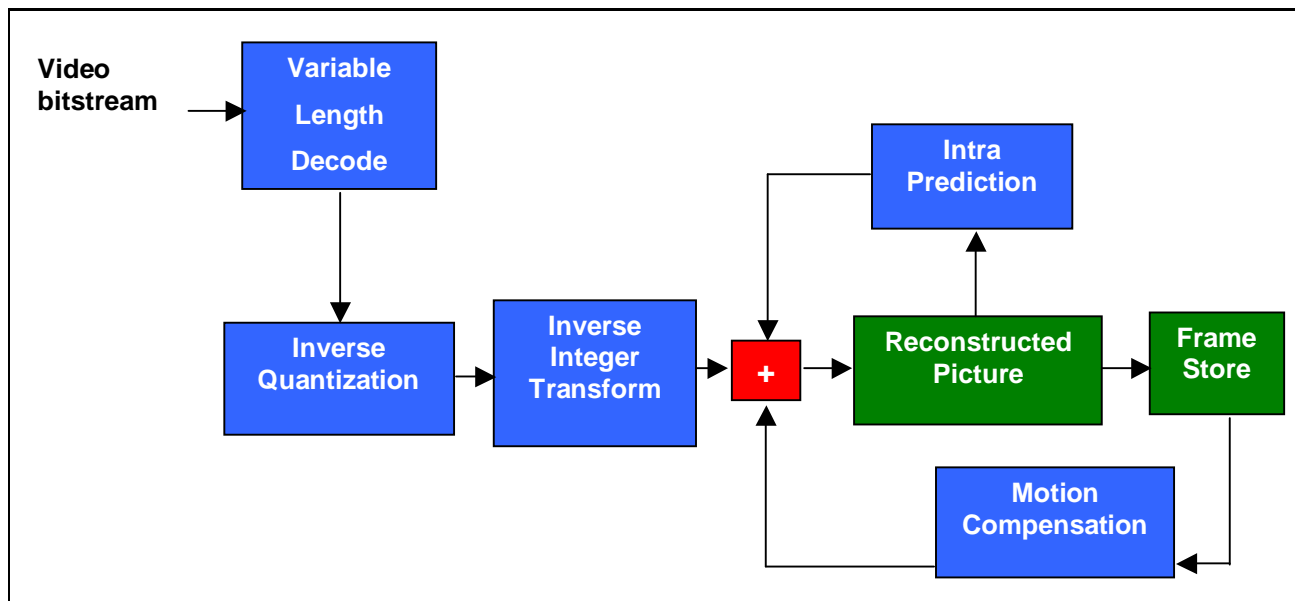


Fig. 3 H.264 Decoder Block Diagram

Control Registers

There are three 16-bit control registers in the IIT block, all of which are loadable from the data bus. The fields of each register are shown below:

CTL_REG1

QPY_6[3:0]	the luma quantisation parameter divided by 6
QSY_6[3:0]	the luma quantisation parameter for SP/SI slices divided by 6
QPC_6[3:0]	the chroma quantisation parameter divided by 6
QSC_6[3:0]	the chroma quantisation parameter for SP/SI slices divided by 6

Depending on the control input signals sMbFlag and Intra16, the internal signal QP_6 is set equal to QPY_6 or QSY_6 or QPC_6 or QSC_6.

CTL_REG2

Depending on the value of (qP%6), there are three values for the Level shift multiply prescaling operation. These values are loaded into the three level shift multiplier registers.

LS_M0[4:0]	This register is used to multiply c_{ij} when (i, j) is (0,0), (0,2), (2,0), (2,2)
LS_M1[4:0]	This register is used to multiply c_{ij} when (i, j) is (1,1), (1,3), (3,1), (3,3)
LS_M2[4:0]	This register is used to multiply c_{ij} in all other cases.

CTL_REG3

The fields in this register are:

QPY_6M2[3:0]	the luma quantisation parameter divided by 6, then minus 2.
2MQPY_6[1:0]	2 minus the luma quantisation parameter divided by 6.
QPC_6M1[2:0]	the chroma quantisation parameter divided by 6, then minus 1.
LS_OUT[4:0]	LevelScale Multiplier for Output

Timing

The timing diagram shown in Figure 4 explains different aspects of the control signal interface. Fig. 4 is a typical timing diagram for a 2-D IIT. Data will enter in row-order and exit in column-order. The numbers on the boxes on DIN port represents the sequence in which the core interprets the data. Data_Valid qualifies the data on DIN when RFD is also high. So, if the Data_valid pins remains high even when RFD is low, the core does not read the data on DIN port, and similarly, when the RFD pin is high, the core waits for the Data_valid to be pulled high to read data on DIN. After some processing latency, the core emits the results sequentially column-wise on the DOUT port, qualified by a high Output_Data_Valid signal.

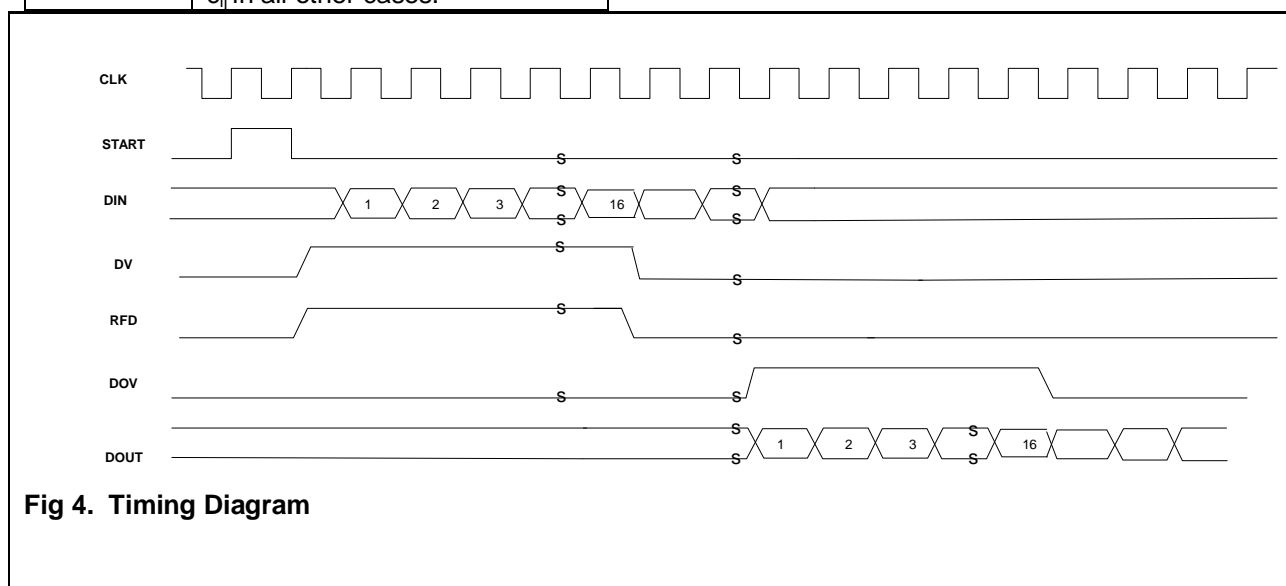


Fig 4. Timing Diagram

Interface Signals

<p>DIN (Data input): The input pins DIN provide the data stream into the IIT core. The user can set this width to 2, 4, 8. The data is read from this port when the RFD is also set high.</p>	<p>ODV (Output Data Valid): This signal indicates that there is valid output data on the DOUT port.</p>
<p>RFD (Ready for data) : The active high output control signal RFD indicates that the IIT core is capable of accepting new input data. When RFD is asserted, the ND signal is enabled; otherwise, it is ignored. The RFD signal can be used to flow control the input to the core. Note that RFD signal is asserted one clock before the rising edge of the CLK signal. This provides for a complete clock cycle for data set-up on the DIN and control signals.</p>	<p>ADR – Address of the control register for read or write operation.</p>
<p>DV (Data valid): This signal indicates that there is valid input data on the DIN port. When the active high DV signal is asserted along with the active high output control signal RFD, data on the input port is sampled.</p>	<p>CS (Chip select) – This chip is selected for a bus transaction</p>
<p>START_IN: These three input signals indicate which type of transform is starting (Residual, DC Chroma, DC Luma). The first data value will be on next cycle when the Data_valid signal is active.</p>	<p>DATA_BUS – Bidirectional data lines from the processor bus.</p>
<p>START_OUT : These three signals output indicate which type of transform is starting (Residual, DC Chroma, DC Luma). The first data value will be on next cycle when the Output_Data_valid signal is active.</p>	<p>RD (Read) - When this line is high, data is read from control register, specified by ADR, and is valid on the rising edge of the clock.</p>
	<p>WR (Write) – When this line is high, data is written to control register, specified by ADR, on the rising edge of the clock.</p>
	<p>RST (Reset): The core offers a synchronous reset facility through this port. A low level on this port resets the internal state of the core.</p>
	<p>CLK (Clock): The input signal CLK is the processing clock for the fully synchronous core. All clocking is done on the rising edge of the CLK signal.</p>
	<p>DOUT (Data output): The output port DOUT provides the result of the IIT core. The output control signal RDY when asserted indicates the value on the DOUT port is valid. The results of the 2-D Inverse Integer Transform are provided sequentially on this port, column-wise.</p>

When the user pulls the RESET pin low, the core resets all the internal state machines to initial values irrespective of whether a set of data has already been received or not.

Processor Data Bus Interface

The signals are:

Data – bidirectional data lines. This bus is 16 bits wide.

Control – read/write signals

Address – specifies address of control register. Up to 4 addresses may be specified. The address map is shown in Table 1. There is also one chip select line.

The timing of the signals will vary according to the bus interface selected. Currently, PixSil Technology supports the ARM, ARC and MIPs processors.

Performance and Scalability

Both the I/O bandwidth and the processing resources can be scaled as shown in Table 2. Currently, PixSil Technology currently supports three versions. The Input data bus can scale from 8,4,2 bits. Accordingly, the Output data bus will scale to 16,8,4 bits.

Table 2

Throughput, Speed, and Silicon Resources

H.264 Level	Cycles per sample	Gates	Clk
1.1 (3K MB/s)	8	3 K	10 MHz
2.0 (12K MB/s)	4	5 K	40 MHz
3.0 (40K MB/s)	2	8 K	30 MHz

MB/s represents Macroblocks per second.

Table 1

Address of control registers

Control Register	Address	Access
CTL_REG1	b00	Read/write
CTL_REG2	b01	Read/write
CTL_REG3	b10	Read/write

Ordering Information

The IIT core and verification IP may be ordered directly from PixSil Technology at the address shown below.

Reference

Joint Final Committee Draft (JFCD) of Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC)

For more information, please contact us at:

PixSil Technology Corporation
 4533 MacArthur Blvd., Newport Beach, CA 92660
 Email: info@pixsiltech.com Internet: www.pixsiltech.com