

CAST

DCT-FI

2D Forward and Inverse Discrete Cosine Transform Core

Features

Ease of Integration & Performance

- High clock speed (>250 MHz in 0.18um ASIC technologies)
- Low gate count
- Single clock cycle per sample operation on both directions
- Low latency (89 cycles)

Design Quality

- Fully compliant with the JPEG standard
- Registered input and outputs
- Strictly positive edge triggered fully synchronous design
- Robust verification environment
- No internal latches or tri-states, scan-ready design

Optional add-on Features

- Operation over 16x16 blocks of samples/DCT coefficients
- Down-scaling in the frequency domain by run-time programmable integer factor (1 up to 8)
- Programmable mode of operation (8-8 or 2-4-8)

The DCT-FI core implements the combined 2D Forward/Inverse Cosine Transforms. Most of the image/video compression standards (JPEG, MPEGx, H.261, H.263, DV etc) are based on the Discrete Cosine Transform (DCT). Able to operate over 8x8 and 16x16 blocks of samples/DCT coefficients, the DCT-FI covers the needs of hardware image/video compression and decompression systems in the most efficient manner. Possibly the fastest core in the market, it is able to provide processing rates up to 200 MSamples/sec in FPGA technologies and over 250 MSamples/sec in ASIC technologies. Furthermore, the core allow the designers to perform area/quality trade-offs by adjusting the cosine coefficients and data-path precision. Down-scaling in the frequency domain, as an optionally supported feature of the core, allows reconstruction at various resolutions from the same input stream of coefficients. Finally the 2-4-8 DCT/IDCT transform, as this is specified in the DVC (DV) standard, can as well be optionally supported by the DCT-FI core.

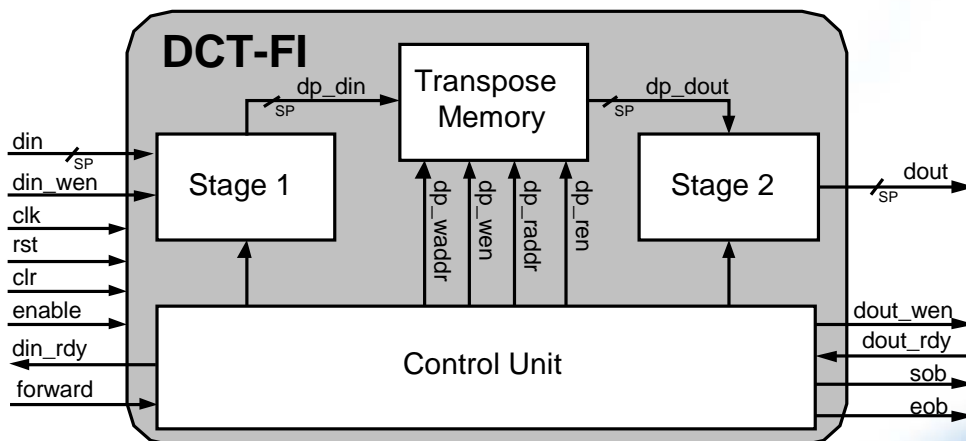
Comprehensive documentation and a complete verification environment - including a bit-accurate model - help designers integrate and verify the core. The DCT-FI is designed for reuse in ASIC and FPGA implementations. The design is fully synchronous with positive edge clocking and no internal tri-state buffers, and scan insertion is straightforward.

Applications

The DCT-FI can be used in a variety of multimedia and image processing applications, including:

- Office automation equipment (Multifunction printers, digital copiers etc)
- Digital cameras & camcorders
- Video production, video conference
- Display-projection systems
- Surveillance systems

Block Diagram



Functional Description

The forward DCT (DCT) is a transform that converts a signal into its constituent frequency components as represented by a set of coefficients. The inverse DCT (IDCT) reconstructs the original signal from its constituent DCT coefficients. A 2-dimensional array of coefficients results by applying the DCT to 2-dimensional signals, such as images. The core receives image samples or DCT coefficients and outputs DCT coefficients or image samples on a block by block basis, where each block has a size of either 8x8 or 16x16. The core implements the DCT or IDCT over the input blocks by performing two 1-dimensional transforms, using row-column decomposition, as defined by the following formulas:

DCT:

$$Y_{uv} = \frac{2}{N} C_u C_v \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} X_{ij} \cos \frac{(2i+1)u\pi}{2N} \cos \frac{(2j+1)v\pi}{2N} = \sqrt{\frac{2}{N}} C_u \sum_{i=0}^{N-1} \left[\sqrt{\frac{2}{N}} C_v \sum_{j=0}^{N-1} X_{ij} \cos \frac{(2i+1)v\pi}{2N} \right] \cos \frac{(2j+1)u\pi}{2N}$$

IDCT:

$$X_{ij} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \frac{2}{N} C_u C_v Y_{uv} \cos \frac{(2i+1)u\pi}{2N} \cos \frac{(2j+1)v\pi}{2N} = \sum_{u=0}^{N-1} \sqrt{\frac{2}{N}} C_u \left[\sum_{v=0}^{N-1} \sqrt{\frac{2}{N}} C_v Y_{uv} \cos \frac{(2j+1)v\pi}{2N} \right] \cos \frac{(2i+1)u\pi}{2N}$$

where $C_u = C_v = \frac{1}{\sqrt{2}}$ for $u, v = 0$ and $C_u = C_v = 1$ otherwise, X_{ij} are the image samples, Y_{uv} are the DCT coefficients.

The intermediate results being produced from the first 1-dimensional transform are stored in the “Transpose Memory”. The Transpose Memory is a dual ported RAM capable of storing an entire 8x8 or 16x16 block resulting from applying the first stage of row decomposition. While the Transpose Memory is written in row-major order, the second stage of processing reads data from the Transpose Memory in a column-major order, effectively performing a transposition of the intermediate results.

The number of bits used for each intermediate result stored in the Transpose Memory, as well as the number of bits used to represent each of the cosine coefficients, is configurable at synthesis time. This allows the designers to perform their own accuracy versus core area tradeoffs. Furthermore, the bit-width of both core inputs and outputs is also configurable at synthesis time. It is noted that the default settings for these synthesis parameters, result to a DCT/IDCT implementation that satisfy the accuracy criteria of the JPEG standard.

The first DCT coefficient/image sample of a block will appear at the output 89 clock cycles after the first image sam-

ple/DCT coefficient of an input block has been fed to the core.

Implementation Results

DCT-FI reference designs have been evaluated in a variety of technologies. The following are sample ASIC results all core I/Os are assumed to be routed on-chip and with pre-layout results after area optimization during synthesis, as reported from the synthesis tool and silicon vendor design kit under typical conditions.

ASIC Technology	Logic Eq. Gates	Frequency	Memory
UMC 0.18μ process	31,593	250 MHz	960 bits
TSMC 0.09μ process	30,383	450 MHz	960 bits

Support

The core as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

Verification

The core has been verified through extensive simulation and rigorous code coverage measurements. Being embedded in numerous of products, the core is silicon proven in both FPGA and ASIC technologies.

Deliverables

The core is available in ASIC (synthesizable HDL) and FPGA (netlist) forms, and includes everything required for successful implementation. The ASIC version includes:

- HDL RTL source code
- A bit-accurate model (BAM) of the core including support of custom test vector generation
- Sophisticated self-checking Testbench (Verilog versions use Verilog 2001) supporting test vectors, expected results, and verification
- RTL and gate level (FPGAs) simulation scripts
- Synthesis scripts
- Comprehensive user documentation, including detailed specifications and a system integration guide