

CAST



LJPEG-D

Lossless JPEG Decoder Core

The LJPEG-D core implements a Lossless JPEG (LJPEG) decoder in a compact, high-performance, stand-alone package ideal for applications where bit-by-bit accurate reproduction of an image is essential.

The LJPEG-D decodes images that conform to the spatial (sequential) lossless encoding mode (SOF3) of the ISO/IEC 10918-1 standard (CCITT T.81 recommendation). Rather than the Discrete Cosine Transform (DCT) functions used for lossy JPEG compression - which can introduce round-off errors - the lossless part of the standard employs a reversible predictor function. The LJPEG-D core can thus decode images with no information loss, and requires a smaller physical implementation than what necessary for lossy JPEG image decoding.

Evaluation designs show that the LJPEG-D core fits in a variety of Xilinx devices, requiring, for example, approximately 3,300 slices for a Spartan-3 implementation. Its optimized architecture also enables high performance, reaching up to 95 MSamples/sec with Virtex-5.

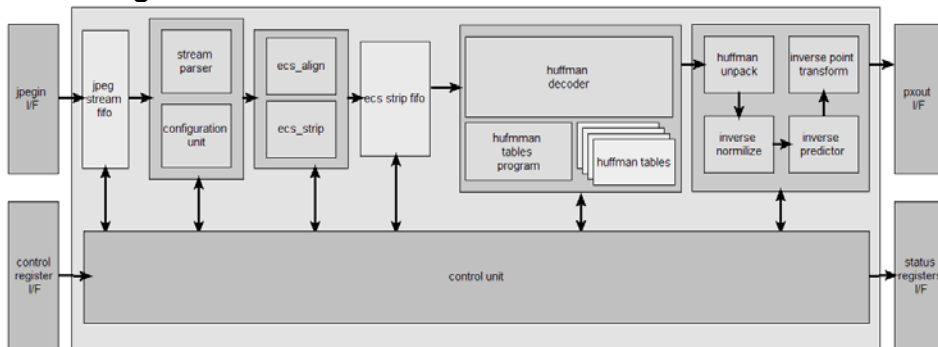
The LJPEG-D is a fully synchronous, strictly positive-edge design with no internal three-state buffers. Comprehensive documentation and a complete verification environment - including a bit-accurate model - help designers integrate and verify the core.

Applications

The LJPEG-D provides a fast, economical solution whenever lossless image compression is essential, including applications such as:

- Medical, military, and space imaging.
- Professional, studio-quality cameras and editing suites.
- High-end film and photo scanners.
- Industrial machine vision systems.

Block Diagram



Features

- Conforms to the spatial (sequential) lossless encoding mode (SOF3) of the ISO/IEC 10918-1 standard (CCITT T.81 recommendation).
- Standalone operation.
 - ISO/IEC 10918-1 JPEG stream input.
 - Decoded pixel samples output.
- Self-programmable through the standard JPEG markers.
 - Programmable image dimensions.
 - Full range sample precision support (2 to 16 bits per sample)
 - Up to four stream programmable Huffman tables.
 - Programmable Restart Interval.
 - Programmable Point Transform function.
 - Headers errors catch-up features.
- Compact, high-performance architecture.
 - Easily fits most Xilinx device families (see implementation results table).
 - Achieves up to 95 MSamples/sec (Virtex-5 implementation).
- Robust and simple to use.
 - General purpose, fully stallable, streaming I/O interfaces.

Limitations with respect to the ISO/IEC 10918-1 standard:

- Up to three image-components are supported (Nf field of the SOF3 marker segment = 1 or 2 or 3).
- Single scan encoding (only one SOS marker segment, with Ns field = Nf).
- No DNL marker support (Y field of the SOF3 marker segment > 0).
- Fixed parameters
 - No sub-sampling (Hi and Vi fields of the SOF3 marker segment = 1).
 - Prediction function is fixed to the left-hand predictor, predictor 1. (Ss field of SOS marker segment = 1).

Functional Description

Lossless JPEG was added to the ITU-T JPEG recommendations in 1995. The JPEG lossless mode of operation does not use the 2D-DCT that is used in the lossy mode, since round-off errors prevent a 2D-DCT calculation from being reversible. For the same reason, one would not normally use color space conversion or down-sampling, although these are permitted by the standard.

The lossless mode of the standard codes the difference between each pixel and the "predicted" value for the pixel. The predicted value is a function of the already-transmitted pixels just above and to the left of the current one (8 different predictor functions are defined in the standard). The sequence of the calculated differences (prediction errors) is encoded using the same back end (Huffman or arithmetic) used in the lossy mode of the standard.

The LJPEG-D core receives an ISO/IEC 10918-1 compatible lossless JPEG stream via the JPEG-In Interface. The core is capable of decoding compressed images with up to three components, having 2-16 bits precision per component sample, previously encoded using the predictor 1 function and the Huffman coding back end.

While the LJPEG-D core reads the JPEG stream, it parses the marker segments and programs itself accordingly. The programmable parameters the core can extract from the stream include the image dimensions, the Huffman coding tables, the restart interval if any, and the point transform function. During the parsing phase of the JPEG markers, the core enables a header error catch-up function so that corrupted streams can be detected. After parsing the marker segments, the core decompresses the entropy coded segment of the lossless JPEG stream, and outputs image samples via the pixel-out interface.

Implementation Results

LJPEG-D core has been evaluated in a variety of technologies. The following are sample Xilinx results using balanced area/speed constraints during synthesis and place and route, while assuming that all core I/Os are routed off-chip.

Family Device	Slices	IOBs	GCLK	Fmax (MHz)	Special Features	ISE Version
Spartan-3 3s1000-5	3,315	84	1	38	2 RAMB16	9.2i
Spartan-3E 3s1200e-5	3,703	84	1	40	2 RAMB16	9.2i
Virtex-II 2v1000-6	3,423	84	1	50	2 RAMB16	9.2i
Virtex-4 4vlx15-12	3,214	84	1	66	2 RAMB16	9.2i
Virtex-5 5vlx30-3	1,508	84	1	95	1 RAMB18X2 1 RAMB18X2SDP	9.2i
Virtex-6 6vlx75t-1	1,300	84	1	80	2 RAMB18E1	11.5i

Support

The LJPEG-D 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 LJPEG-D core has been verified through extensive simulation using a large set of test vectors and reference results.

Deliverables

The LJPEG-D is available as a soft core (synthesizable HDL) for ASIC technologies and as a firm core (netlist) for FPGA technologies, and includes everything required for successful implementation. The Xilinx version includes:

- Post-synthesis NGC or EDIF netlist.
- Place and route scripts.
- Simulation scripts.
- Sophisticated self-checking Testbench (Verilog versions use Verilog 2001) including test vectors, expected results, and verification engine.
- A bit-accurate model (BAM) including custom vector generation support, and a software library of the bit accurate model functions.
- Comprehensive user documentation, including detailed specifications and a system integration guide.