



An Industry Standard Benchmark Consortium

DENBench™ Version 1.0

Benchmark Name: Huffman Decoding

Highlights

- **Benchmarks potential performance of a processor in a digital camera and is modeled on picture data (YUV data)**
- **Huffman Decoding is a key algorithm in JPEG, MPEG, and compression schemes**
- **Integer implementation**
- **Stresses table lookup, bit manipulation, shifting**
- **Implements CRC to Check Output Quality**

Application Huffman decoding is a key algorithm in JPEG, MPEG, and other compression schemes used in digital cameras. Details on Huffman coding and decoding are available here: http://en.wikipedia.org/wiki/Huffman_coding

Benchmark Description The Huffman Decoding benchmark initializes the AC and DC chrominance and luminance tables (4 tables), fills the various buffers, and then performs the Huffman decoding function using a fairly standard Huffman implementation.

Analysis of Computing Resources This benchmark concentrates mostly on lookup, bit manipulation, and shifting rather than file I/O. Memory-to-memory operations are important for performance, as intermediate values are constantly being stored.

Optimizations Allowed **Out of the Box / Standard C Full Fury / Optimized**

- The C code must not be changed for Out of the Box unless it must be modified to get it to compile. All changes must be documented and must not have a performance impact.
- For Out of the Box, additional hardware can be used if it does not require code changes.
- All optimized libraries must be part of the standard compiler package, and/or available to all customers
- The EEMBC Test Harness Regular or Test Harness Lite may be used. Test harness changes may be made for portability reasons if they do not impact performance
- For Optimized, the basic algorithm may be changed and/or the code can be rewritten in assembler. We report PSNR scores to help you judge quality of computational processing.
- For Optimized, optimized libraries can be used if they are publicly available.
- For Optimized, hardware-assist can be used if it is on the same processor as that being benchmarked.
- For Optimized, in-lining is allowed.
- Additional data files may be used by ECL during certification to ensure the correctness of the optimized benchmark. You should not assume data patterns during optimization.