# H.264 Motion Estimation Engine v1.0



## Introduction

The H.264 Motion Estimation Engine Version 1.0 is a fully functional netlist implemented on a Xilinx® FPGA. The Motion Estimation core accepts input parameters and macroblocks and generates output motion vectors and Sum of Absolute (SAD) values in accordance with the ITU-T Video Coding Experts Group (VCEG) together with the ISO/IEC Moving Picture Experts Group (MPEG) as the product of a collective partnership effort known as the Joint Video Team (JVT). The collaborative effort is also known as H.264/AVC/MPEG4 Part 10.

## Features

- H.264/MPEG-4 Part 10 Baseline/Main/High Profiles @ Level 4.1
- Compliant with International Standard ISO/IEC 14496-10:2005 (E) Rec. H.264 (E)
- 1080i@60 fields per second or 1080p@30 frames per second operation at 275 MHz
- 720p@30 frames per second operation at 130 MHz
- Integrated coded block pattern generation
- 8 x 4 block searches for Full PEL locations
- 112 x 128 search range
- 120 candidate positions (10 predictors with 4 x 3 region around each predictor)
- External memory bandwidth 1.65 Gigabits per second (Gbps) (720p@30) to 3.74 Gbps (1080i@60)
- Selectable reference frame at frame update through frame pointer parameter
- User input prediction vectors
- Optional output port for all SAD values and Motion Vectors

| LogiCORE™ Facts<br>Core Specifics | |
|---|---|
| **Supported Device Families** | Virtex®-5, Virtex-4, Spartan®-3A DSP |
| **Resources Used** | |
| Virtex-5 | 2952 LUTs, 3040 FFs, 26 DSP48s, 15 block RAMs |
| Virtex-4 | 2954 LUTs, 3252 FFs, 30 DSP48s, 19 block RAMs |
| Spartan-3A DSP | 3297 LUTs, 3274 FFs, 30 DSP48s, 19 block RAMs |
| **Provided with Core** | |
| Documentation | Data Sheet, User Guide |
| Design File Formats | EDIF |
| Verification | VHDL Test Bench ModelSim® 6.1c SE, MicroSoft Studio v6.0, ActivePerl 5.8.3. |
| Simulation | ModelSim 6.1c SE |
| Instantiation Template | VHDL Wrapper |
| **Design Tool Requirements** | |
| Synthesis | Synplicity Synplify_Pro 8.8.04 |
| Xilinx Implementation Tools | Xilinx ISE™ 9.2.01i (from ngd_build) |
| **Support** | |
| Xilinx provides technical support for this LogiCORE product when used as described in the product documentation. Xilinx cannot guarantee timing functionality, or support of product if implemented in devices not listed in the documentation, or if customized beyond that allowed in the product documentation, or if any changes are made in sections of the design marked DO NOT MODIFY. | |

## Applications

The H.264 Motion Estimation core can be utilized in H.264 video encoding applications where hardware acceleration is needed to achieve real time operation. Typical video applications are video surveillance, video conferencing, and video broadcast.

## Description

The H.264 Motion Estimation core computes the sum of absolute difference (SAD) for a set of 120 search locations within a 112 x 128 search window for 8 x 4 blocks. The search locations are determined by a set of 10 seeds that are provided by the user and the 4 x 3 region to the right and down from each seed. The core provides as output the 120 SAD calculated values and the motion vectors. In addition, the coded block pattern is computed for a macroblock and the best motion vector for each sub-block is provided.

The functional inputs and outputs to the H.264 Motion Estimation Core are:

- Inputs:
  - New Macroblock
  - Parameters for the New Macroblock
    - Macroblock location (h, v)
    - Motion Vector predictors

      Reference Frame pointer

      Reference Frame
- Outputs
  - Output Parameters
    - Best Motion Vector
    - Motion Vectors
    - SAD values
    - Coded block pattern

Figure 1 is a diagram of the H.264 Motion Estimation architecture. Motion estimation requires the associated portion of the reconstructed frame to be available before any processing can start. The Reference frame is stored in the external memory and is accessible through an external memory controller. One port is reserved for writing the reconstructed frame, and the user is responsible for loading it prior to starting the motion estimation process. The second port is reserved for motion estimation reading the necessary data from the external memory.

The motion estimation is delivered as a netlist. The Motion Estimation core requires the user to provide a list of initial motion vectors where the searches will be performed. The Motion Estimation module is responsible for performing the searches around the provided location, as well as for the search area data management. The motion estimation algorithm uses the input information such us macroblock location for processing the current macroblock.

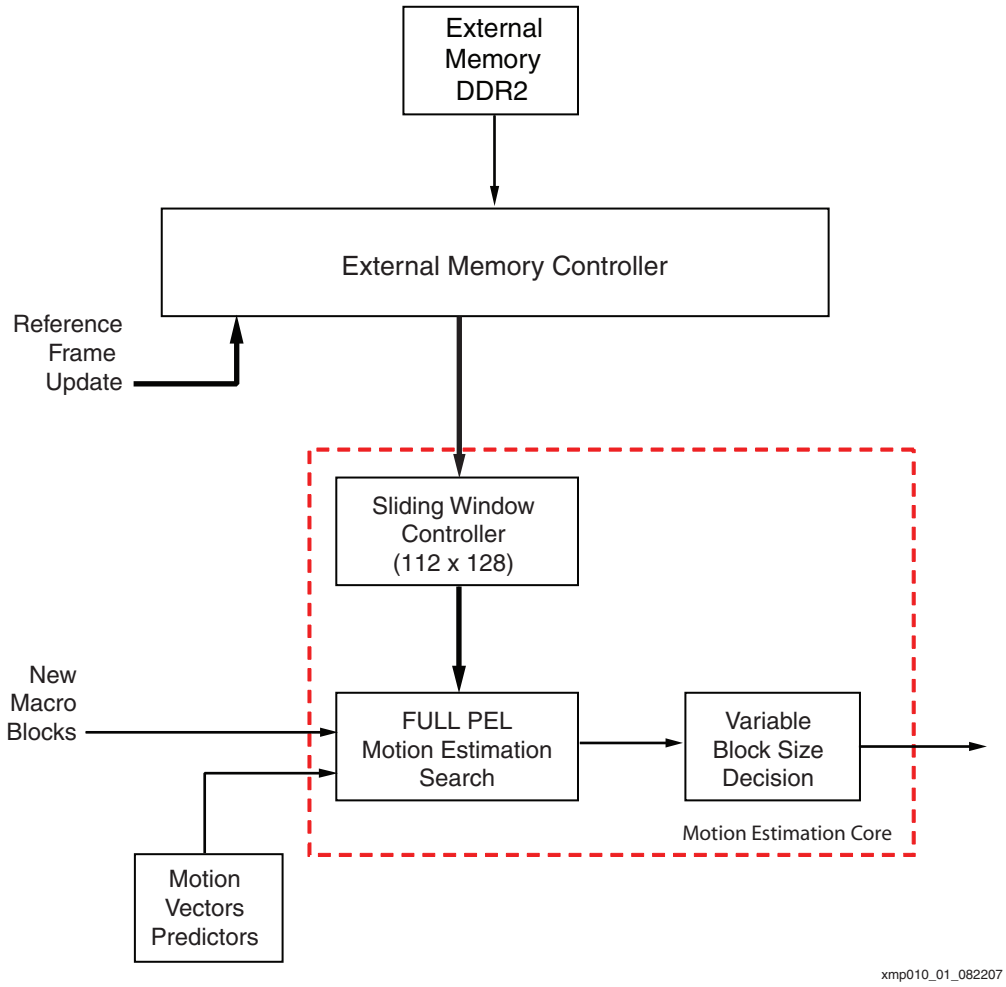


*Figure 1:* **H-264 Motion Estimation Architecture**

The motion estimation core uses a sliding window module to load the information required by the current macroblock. The core allows a total search window of 128 x 128. However, to enable parallelism between search module and fetching data from the memory controller, one column is always used to load data. This makes the search window available for search 112 x 128 ((128-16) x 128). The external memory bandwidth required can be calculated based on following information: the sliding window controller requires refreshing of one column of macroblocks in the search area. The bandwidth formula is:

$$BW_{Mem} = N_{Total} \times F_{rate} \times 8 \times (16 \times 16)$$

Where $BW_{Mem}$ is the required memory bandwidth, $N_{Total}$ is the total number of macroblocks from a frame and $F_{rate}$ if the video frame rate in frames per second.

The factor 8 is attributed to the fact that the sliding window height is 8 macroblocks high and the algorithm requires one column of MB refresh for each motion estimated macroblock. For example, a 720p video sequence processing requires the following bandwidth: 720p => 3600 macroblocks, frame rate 30 fps, and it results in 210.9375 Mega Pixels per second => 1.7 Gbps as throughput requirements for the external memory.

The Full PEL search module is an efficient and highly parallel module which performs the evaluation of the sum of absolute differences. The basic search performed is on 8 x 4 blocks. The user provides ten search seeds and the search module uses the respective seeds to perform a search of 4 x 3 candidate locations around this seed. The best cost is chosen as the winner and passed further in the processing chain to the variable block calculator. The rough processing performance of the search block is approximately 10 seeds for every 8 x 4 block for a 720p, 30 fps video sequence, and hardware running at 130 MHz. This corresponds, under the actual time constraints, to an evaluation of 120 candidate positions evaluated for every block.

The processing cycle budget is calculated based on the following formula:

$$No_{clk} = \frac{F_{MHz} \times 10^6}{N_{Total} \times F_{rate}} \quad Cycles / Macroblock$$

Where $No_{clk}$ represents the number of clock cycles allocated for processing one basic unit (Macroblock). The number of clock required depends on the $F_{MHz}$, the clock frequency that hardware runs, and also on $N_{Total}$, the total number of macroblocks from a frame and on $F_{rate}$ the video frame rate in frames per second. For example, for a 720p sequence at 30 fps and the hardware running at the 130 MHz, the number of clocks allocated for each macroblock is (130 x 106)/(80 x 45 x 30) = 1204 clocks per macroblock. For 8 blocks of 8 x 4 blocks, this corresponds to a clock budget of 150 clocks per 8 x 4 block. Each Macroblock contains 8 blocks of 8 x 4, which shows that the implementation can evaluate 10 motion vectors seeds for each macroblock. For this motion estimation core, the number of seeds is fixed to 10.

The results of the full PEL are passed to the Variable Block Size (VBS) decision blocks. VBS uses the basic information of the FULL PEL search to determine if two or more 8 x 4 blocks have the same motion vectors, and based on this information, it decides whether to merge the blocks into bigger blocks, (i.e., two 8 x 4 to 8 x 8 blocks). The supported block sizes are 8 x 4, 8 x 8, 16 x 8, 8 x 16, and 16 x 16.

## Performance

Target clock FMax and subsequent macroblock throughput are summarized in Table 1.

*Table 1:* **Performance Summary**

| FPGA Family | Clock FMax | Macroblock Throughput (Macroblocks/s) | External Memory Bandwidth (Gbps) | Notes |
|---|---|---|---|---|
| Spartan-3 DSP speed grade -4 | 130 MHz | 108,000 | 1.65 | Supports up to 720P@30 |
| Virtex-4 speed grade -12 | 225 MHz | 200,000 | 3.1 | Supports up to 720P@50 |
| Virtex-5 speed grade -3 | 275 MHz | 244,800 | 3.74 | Supports up to 1080P@30, 1080i@60 |

## Ordering Information

The H.264 Motion Estimation Engine core is provided in netlist format. The core is licensed under the terms of the Xilinx LogiCORE Site License Agreement, which conforms to the terms of the SignOnce IP Site License standard defined by the Common License Consortium. A free evaluation version of the core is available at the Xilinx IP Evaluation Lounge.

The core may be licensed through your local Xilinx sales representative. For part number and ordering information, refer to the H.264 Motion Estimation Engine product page.

## Revision History

The following table shows the revision history for this document.

| Date | Version | Revision |
|---|---|---|
| 10/24/07 | 1.0 | Initial Xilinx release. |
| 04/23/08 | 1.1 | Edits to Table 1. |

## Notice of Disclaimer