

## H.265 HD Video Decoder IP Core

### Product Description

SOC provides the H.265 decoder in three formats: IP cores for FPGAs, ASIC Chipsets, and all-in-one hardware modules.

IP cores are available for both Xilinx and Altera FPGAs. SOC configures the cores according to user specifications, including I/O formats.

The SOC codec chipsets (SOC-MCodec™) are AISC chips based on the SOC IP cores. Standard H.265 decoder chipsets for different specifications are available.

The SOC codec modules are System-on-Module (SoM) cards, based on SOC codec IP cores, that can be connected to user devices/PCBs using a standard DDR3 memory connector.

Users have the options of using the codec IP cores, chipsets, or modules. If IP cores are preferred, users have the option of Xilinx or Altera FPGAs.

SOC also offers product development boards, which allow users to develop products using the SOC codec IP core, chipsets, and modules.

### Key Features

- All-hardware Design (without embedded processors)
- High Speed (Low latency)
- Small Silicon Footprint
- Low Power
- High Reliability (due to hardware architecture)
- High-Precision – 8/10/12bits
- High-Video Quality
- Video Scalar IP core available
- High-Output Bandwidth Version Available
- User Controllable API
- Option of IP Core or Module
- Video Transmission Cores available
- Development Board available

### Specifications

- Standard: H.265/HEVC (ISO/IEC 23008-2:2015)
- Video Encoder Profiles: Main 4:2:2, 12
- Input Bit Rates: 1-100Mbps & above
- Video Resolutions: HD 1080p upto 120fps
- Chroma Formats: 4:2:2 or 4:2:0
- Precision: 8 - 12 bits
- Input Format: H.265 Elementary, or Transport Stream
- Video Output Format: RGB or YUV
- Audio Support: AAC or MPEG-2 Layer-II
- Latency: 0.25ms
- Power Consumption: 1-3w (Core only)
- Target FPGAs: Xilinx or Altera

### FPGA Resources

	Xilinx FPGAs	Altera FPGAs
Logic Resources:	80,000 LUTs	50,000 ALMs
Block RAMs:	5,000kb	5,000kbits
DSPs:	40 DSPs	40 DSPs

### H.265 Video/Audio Decoder Chipset



### H.265 Video/Audio Decoder Module

