

# CAST

## CSC

### Color Space Conversion Core

The CSC core is a compact, high-performance, highly flexible color space conversion core, which can be used to convert from any color space with 3 color channels, to any color space with 3 color channels (e.g. RGB to YCrCb, YCrCb to RGB).

Different color space models are used for different purposes in video/image processing systems. For example computer monitors typically receive frames in the RGB color space, while in order to be compressed frames are typically converted to a luminance – chrominance color space (e.g. YCrCb). So, color space conversion is often necessary when transferring data between devices that use different color space models.

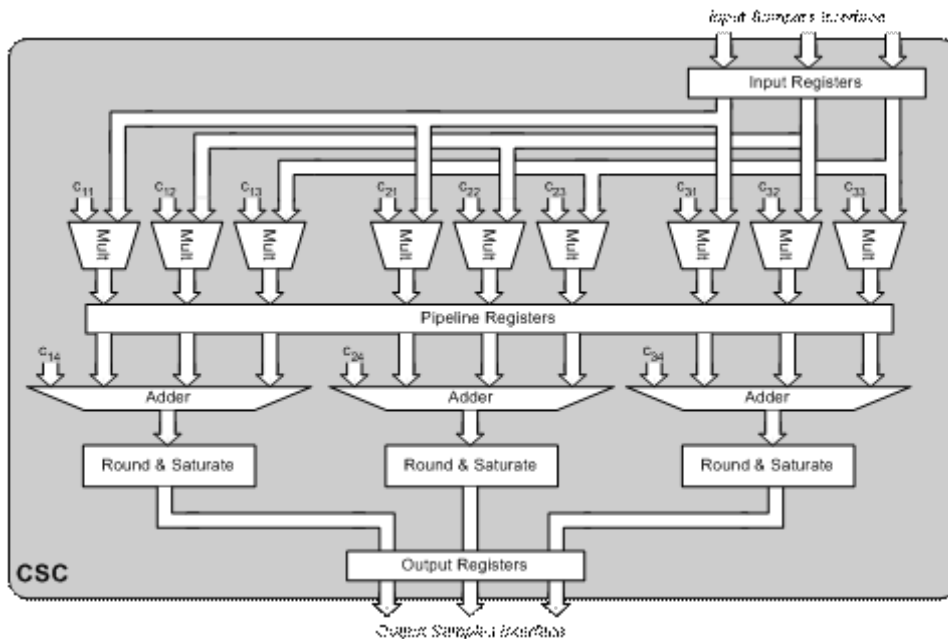
The CSC is a testable, microcode-free design developed for reuse in ASICs and FPGAs.

#### Applications

The CSC can be utilized for a variety of multimedia applications including:

- Image Grabbers
- Display Systems
- Image/Video processing applications
- Image/Video compression / decompression systems

#### Block Diagram



#### Features

##### Synthesis-time configurable conversion function

- Computer R'G'B' to Y'CrCb
- Y'CrCb to Computer R'G'B'
- Studio R'G'B' to Y'CrCb
- Y'CrCb to Studio R'G'B'
- Computer R'G'B' to Y'UV
- Y'UV to Computer R'G'B'
- User specified

##### Synthesis-time "tunable" architecture

- Configurable number of bits per input and output sample
- Configurable number of bits per transformation-table coefficient
- Configurable data-path accuracy
- Pipelined or non-pipelined multipliers
- Selectable synchronous and/or asynchronous reset
- Configurable sensitivity level for all control signals

##### Optional Supplementary Functionality

- Gamma correction removal
- Quantization/Dithering
- Up or Down -Sampling

##### Design Quality

- Continuous one symbol per clock cycle processing
- High clock rate
- Low gate count (<3k eq. gates)
- Low Latency (5 cycles)
- Fully scan insertable design
- Registered input and outputs
- Solid verification plan
- ANSI-C Bit-accurate model

## Functional Description

The CSC core performs the ABC to XYZ color space conversion. The CSC core uses the following equation in order to convert data from one color space to another:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{bmatrix} \times \begin{bmatrix} A \\ B \\ C \\ 1 \end{bmatrix}$$

After the conversion of a triplet of input samples, the core performs rounding and saturation control before outputting the converted triplet of samples.

## Implementation Results

The CSC core can be implemented either in FPGA devices or ASIC technologies. The following table provides indicative implementation data in terms of area and clock frequency for an ASIC technologies. The provided indicative implementation data are acquired for the pipelined version of the core implementing Y'CrCb to computer R'G'B' conversion with input and output bit-width of 8 bits, conversion coefficients bit-width of 10 bits, and the data-path accuracy of 12 bits. Other conversions perform slightly faster, and occupy approximately the same area.

It is noted that area requirements are reduced for the non-pipelined version of the core, and if the enable and clear ports are not used (i.e. they are hardwired before synthesis). For example, for the same configuration described above but with hardwiring the enable and clear ports, the core occupies 22% less area (4157 eq. gates).

ASIC	Logic Gates <sup>1,3</sup>	I/O <sup>3</sup>	f <sub>MAX</sub> (MHz) <sup>1,2</sup> HCLK/CCLK/PCLK	Memory
ATMEL 0.18 um	5422	54	313	-

<sup>1</sup> Optimized for speed

<sup>2</sup> Pre-layout results as reported from synthesis tool+silicon vendor design kit

<sup>3</sup> Assuming all core I/Os are routed off-chip

## 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.

## Deliverables

The core includes everything required for successful implementation:

- HDL RTL source code (soft core) or a post-synthesis EDIF netlist (firm core)
- An example chip implementation, which uses the CSC in a sample system
- Sophisticated self-checking HDL Testbench including external FIFOs, buffers, models of interfaces, and the core (Verilog versions use Verilog 2001)
- Simulation script, vectors, expected results, and comparison utility
- Synthesis (soft) or place and route (firm) script
- Comprehensive user documentation, including detailed specifications and a system integration guide