RGB2YCrCb & YCrCb2RGB

Color Space Converters

February 1997, ver. 1 — Data Sheet

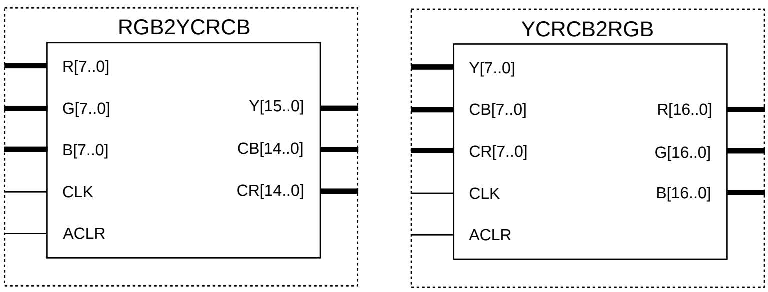
Features

- `RGB2YCrCb` and `YCrCb2RGB` MegaCore functions converting digital video colors to television broadcast signal colors and vice versa
- Optimized for the FLEX® 10K and FLEX 8000 device architectures
- 24-bit RGB2YCrCb and YCrCb2RGB color space conversion
- High-speed operation (100 MHz)
- Supported by MAX+PLUS® II software
- Full precision outputs: rounding and saturation can be performed with the `round` and `saturate` reference designs
- Useful for a variety of applications, including image filtering, machine vision, and digital video

General Description

The `RGB2YCrCb` and `YCrCb2RGB` color space converter functions are useful for a number of image processing and filtering operations. The `RGB2YCrCb` function converts red-green-blue (RGB) color space to the YCrCb (intensity-color red-color blue) color space; the `YCrCb2RGB` function performs the inverse operation. Figure 1 shows the symbols for the `RBG2YCrCb` and `YCrCb2RGB` functions.

Figure 1. RGB2YCrCb & YCrCb2RGB Symbols



Function Prototypes

The Altera Hardware Description Language (AHDL) Function Prototype of the RGB2YCrCb function is shown below:

```
FUNCTION RGB2YCrCb (r[7..0], g[7..0], b[7..0], clk, aclk)
    RETURNS (y[15..0], cb[14..0], cr[14..0]);
```

The AHDL Function Prototype of the YCrCb2RGB function is shown below:

```
FUNCTION YCrCb2RGB (y[7..0], cb[7..0], cr[7..0], clk, aclr)
    RETURNS (r[16..0], g[16..0], b[16..0]);
```

Ports

Table 1 describes the input and output ports for the RGB2YCrCb function.

Table 1. RGB2YCrCb Ports

Name	Type	Numbering Representation Note (1)	Description
r[7..0]	Input	Unsigned 8.0	Red data
g[7..0]	Input	Unsigned 8.0	Green data
b[7..0]	Input	Unsigned 8.0	Blue data
clk	Input	–	Clock
aclr	Input	–	Asynchronous clear
y[15..0]	Output	Unsigned 8.8	Intensity data
cb[14..0]	Output	Unsigned 8.7	Color blue data
cr[14..0]	Output	Unsigned 8.7	Color red data

Note:
(1) The notation *x.y* represents *x* bits to the left of the radix point, and *y* bits to the right.

Table 2 describes the input and output ports for the `YCrCb2RGB` function.

Table 2. YCrCb2RGB Ports			
Name	**Type**	**Numbering Representation** *Note (1)*	**Description**
`y[7..0]`	Input	Unsigned 8.0	Intensity data
`cb[7..0]`	Input	Unsigned 8.0	Color blue data
`cr[7..0]`	Input	Unsigned 8.0	Color red data
`clk`	Input	–	Clock
`aclr`	Input	–	Asynchronous clear
`r[16..0]`	Output	Signed 10.7	Red data
`g[16..0]`	Output	Signed 10.7	Green data
`b[16..0]`	Output	Signed 10.7	Blue data

Note:
(1) Signed refers to two's compliment signed. The notation *x.y* represents *x* bits to the left of the radix point, and *y* bits to the right. The RGB values may be negative if the YCrCb values deviate outside the nominal range. For more control over the values, rounding and saturation functions are available.

Functional Description

The `RGB2YCrCb` function uses the following equations when converting gamma-corrected RGB data to YCrCb data:

$$Y' = 0.257R' + 0.504G' + 0.098B' + 16$$
$$Cr = 0.439R' - 0.368G' - 0.071B' + 128$$
$$Cb = -0.148R' - 0.291G' + 0.439B' + 128$$

The `YCrCb2RGB` function uses the following equations when converting YCrCb data to RGB data:

$$R' = 1.164(Y' - 16) + 1.596(Cr - 128)$$
$$G' = 1.164(Y' - 16) - 0.813(Cr - 128) - 0.392(Cb - 128)$$
$$B' = 1.164(Y' - 16) + 2.017(Cb - 128)$$

☞ Refer to ***Video Demystified*** for more information on these equations (see "References" for details).

Because the inputs are multiplied by constant values, the look-up table (LUT) architecture of the FLEX 10K and FLEX 8000 devices is ideal for efficiently performing the conversion equations.

Some of the coefficients are smaller in magnitude, and therefore the multiplier only has to be as large as the equivalent number of binary digits in the coefficient. For example in the Y' equation, when multiplying the 8-bit input (`b[7..0]`) by 0.098_{10} (or 0.00011001_2), only a 5×8-bit multiplier is required instead of an 8×8 bit-multiplier.

The `y` output of the `RGB2YCrCb` function is an unsigned 16-bit binary number with 8 bits to the left of the radix point and 8 bits to the right. To convert the `y` output to an 8-bit binary number, the value must be rounded.

For more information on data word rounding, refer to *Functional Specification 5 (round Data Word Rounder)*.

Figure 2 shows the block diagram of the Y' component of the `RGB2YCrCb` function. The Cr and Cb components of the conversion are calculated in a similar manner.

Figure 2. RGB2YCrCb Converter Block Diagram

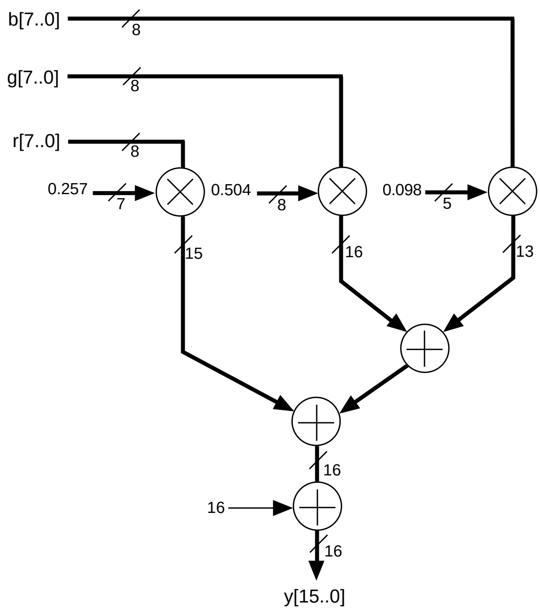
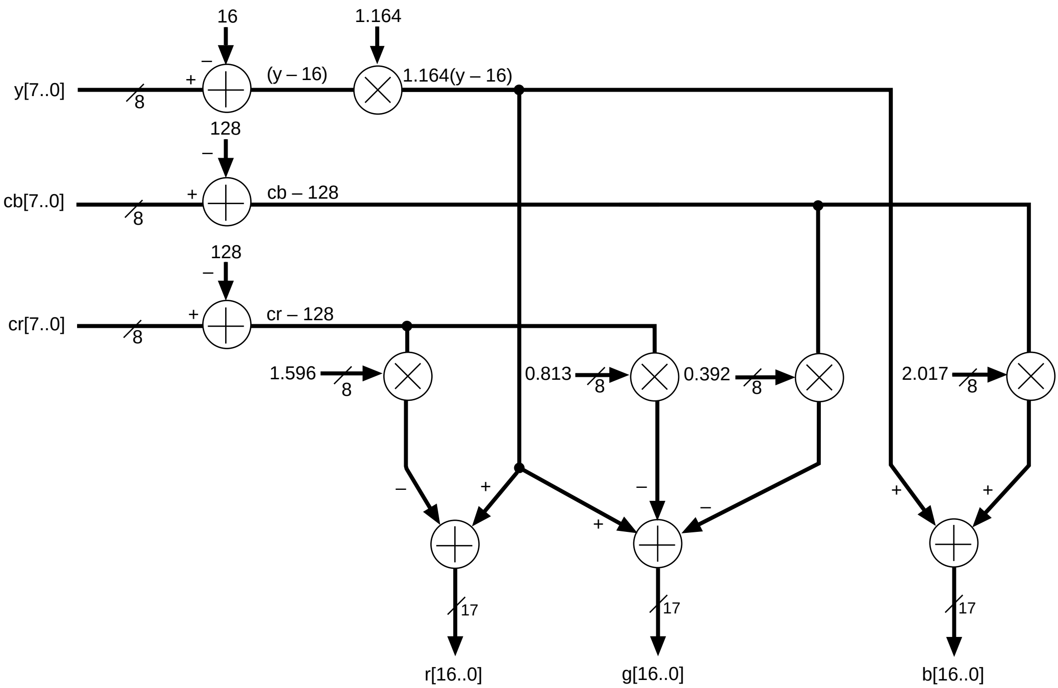


Figure 3 shows the block diagram for the `YCrCb2RGB` function. The *r*, *g*, and *b* outputs of the `YCrCb2RGB` function are 17-bit two's compliment signed numbers with 10 bits to the left of the radix point and 7 bits to the right. To convert the *r*, *g*, and *b* outputs to 8-bit signed numbers, the data words should be saturated and rounded.

For more information on rounding, refer to *Functional Specification 5 (round Data Word Rounder)*; for more information on saturation, refer to *Functional Specification 6 (saturate Data Word Saturator)*.

Figure 3. YCrCb2RGB Converter Block Diagram

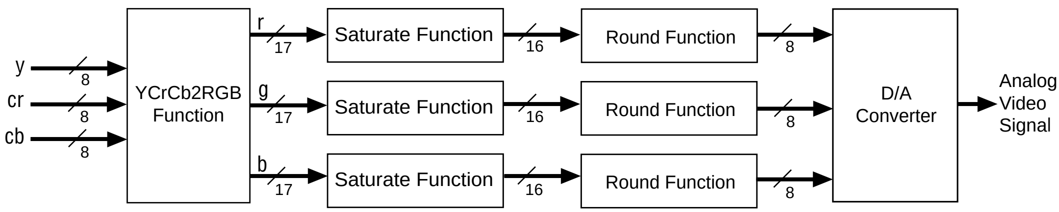


Digital Video System Application

Figure 4 is a simplified illustration of an 8-bit color channel in a digital video system. In this example, color space conversion is performed using the `YCrCb2RGB` function. Depending on the input values to the `YCrCb2RGB` function, the multiplication used in the color conversion may result in 17-bit data words that roll over. To avoid data word roll over, the 17-bit digital video signals are fed to the `saturate` function, where they are saturated to 16-bit words. The signals are then fed to the `round` function, where they are rounded to 8-bit words. At this point, the data words are ready for conversion to an analog video signal.

For more information on data word roll over and saturating, refer to *Functional Specification 6 (saturate Data Word Saturator)*; for more information on rounding, refer to *Functional Specification 5 (round Data Word Rounder)*.

Figure 4. Digital Video Channel



References

Jack, Keith. ***Video Demystified, A Handbook for the Digital Engineer***. Second Edition. Solana Beach: Hightext Publications, 1996.

This book discusses fundamental computer-video applications, including color space conversion considerations such as equations, data word saturation, and data word rounding.

ISBN: 1-878707-23-X (paperback edition), Library of Congress catalog card number: 96-076446.

2610 Orchard Parkway
San Jose, CA 95134-2020
(408) 894-7000
http://www.altera.com
Applications Hotline:
(800) 800-EPLD
Customer Marketing:
(408) 894-7104
Literature Services:
(888) 3-ALTERA
lit_req@altera.com

Printed on Recycled Paper.