round
Data Word Rounder

February 1997, ver. 1 Functional Specification 5

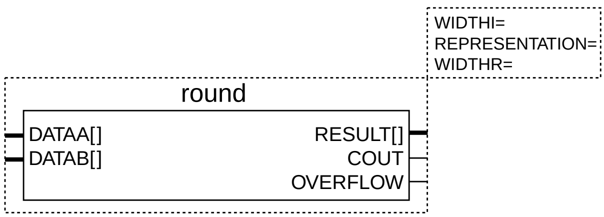
Features

- round reference design implementing a data word rounder
- Parameterized input and output widths
- Optimized for FLEX® 10K and FLEX 8000 device families
- Rounds arbitrary-length words to shorter words
- Signed (two's complement) or unsigned operation
- Built-in post-round adder
- Supported by MAX+PLUS® II
- Useful for a variety of applications, including digital filters, image processing, and video processing

General Description

In most digital signal processing (DSP) systems, word length and word growth effects are important aspects of design. Because the result of a calculation often contains more bits than is desired, some method must be used to reduce the number of significant bits in a particular data path. Although simple truncation of the least significant bits (LSBs) is the easiest method to use, it produces excess noise in the LSB. The `round` function uses a technique of truncation and rounding to decrease the number of significant bits while minimizing noise. Figure 1 shows the symbol for the `round` reference design.

Figure 1. round Symbol



The concept of rounding binary digits is the same as rounding decimal digits. If the digit to the right of the decimal point is greater than 0.5, round the digit up; otherwise, round the digit down. For example, rounding the decimal number 1.25 (01.01_B) yields 1 (01_B), and rounding the decimal number 1.75 (01.11_B) yields 2 (10_B).

A-FS-05-01

The round reference design accepts any bit-width number and properly rounds it to a smaller bit-width by truncating the number and adding a 1 to the number, if appropriate. Also, the architecture of the FLEX logic cells allows the round function to add the rounded number to another number without increasing logic usage.

Function Prototype

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

```
FUNCTION round (dataa[(WIDTHI-1)..0],
datab[(WIDTHR-1)..0]
    WITH (WIDTHI, REPRESENTATION, WIDTHR)
    RETURNS (result[(WIDTHR-1)..0], cout, overflow);
```

Parameters

Table 1 shows the parameters for the round function.

Table 1. round Parameters

Name	Value	Description
WIDTHI	Integers only	Width of input data
REPRESENTATION	"SIGNED" "UNSIGNED"	Sign of the input data. The default value is "UNSIGNED"
WIDTHR	Integers only	Width of output data (< WIDTHI)

Ports

Table 2 shows the ports for the round function.

Table 2. round Ports

Name	Type	Description
dataa[(WIDTHI-1)..0]	Input	Input data
datab[(WIDTHR-1)..0]	Input	Sum data
result[(WIDTH-1)..0]	Output	Rounded output data
cout	Output	Carry out
overflow	Output	Overflow

Functional Description

The `round` function implements a data word rounder that provides parameterized input and output widths.

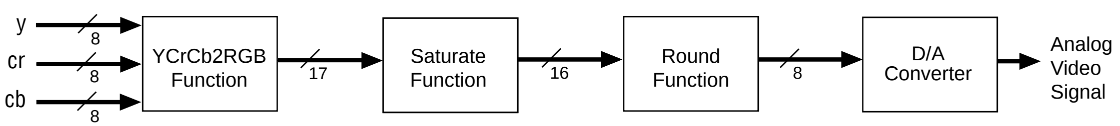
The `round` function rounds a `WIDTHI`-bit number down to a `WIDTHR`-bit number, where `WIDTHI` is the input width and `WIDTHR` is the output width. The `round` function can also add the rounded result to another `WIDTHR`-bit-wide number without increasing logic usage. The `round` function performs the following calculation:

```
result[(WIDTHR–1)..0] = dataa[(WIDTHI–1)..(WIDTHI-WIDTHR)] +
dataa[WIDTHI-WIDTHR–1] + datab[]
```

For example, if the number being rounded is 00101.011_B, the number is rounded down to 00101_B because the fractional part of the number (i.e., the bits to the right of the binary point) is less than 0.5 (0.100_B). When the bit directly to the right of the binary point is 1, a 1 is added to the result because the fractional part of the number is greater than 0.5 (0.100_B).

Figure 2 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 a 17-bit data word that rolls over. To avoid data word roll over, this 17-bit digital video signal is fed to the `saturate` function, where it is saturated to a 16-bit word. The signal is then fed to the `round` function, where it is rounded to an 8-bit word. At this point, the data word is ready for conversion to an analog video signal.

Figure 2. Digital Video Channel



For information on saturation, refer to *Functional Specification 6 (saturate Data Word Saturator)*. For information on color space conversion, refer to *RGB2YCrCb & YCrCb2RGB Color Space Converters Data Sheet*.

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