

ripple\_carry\_multiplication

Corso di ASE anno 18/19

Gruppo 14

PREVITERA GABRIELE

PENNONE MIRKO

PENNA SIMONE



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	full_adder Entity Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Member Data Documentation . . . . .	5
3.1.2.1	IEEE . . . . .	6
3.1.2.2	STD_LOGIC_1164 . . . . .	6
3.2	ripple_carry_multiplier Entity Reference . . . . .	6
3.2.1	Detailed Description . . . . .	6
3.3	ripple_carry_multiplier_testbench Entity Reference . . . . .	7
3.4	rippleCarry_adder Entity Reference . . . . .	7
3.4.1	Detailed Description . . . . .	8
3.4.2	Member Data Documentation . . . . .	8
3.4.2.1	c_in . . . . .	8
3.4.2.2	c_out . . . . .	8
3.4.2.3	S . . . . .	8
3.4.2.4	STD_LOGIC_1164 . . . . .	8
3.4.2.5	width . . . . .	8
3.4.2.6	Y . . . . .	8
<b>4</b>	<b>File Documentation</b>	<b>9</b>
4.1	ripple_carry_multiplication.vhd File Reference . . . . .	9
4.1.1	Detailed Description . . . . .	9
	<b>Index</b>	<b>11</b>



# Chapter 1

## Class Index

### 1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

entity <a href="#">full_adder</a> . . . . .	5
entity <a href="#">ripple_carry_multiplier</a> . . . . .	6
entity <a href="#">ripple_carry_multiplier_testbench</a> . . . . .	7
entity <a href="#">rippleCarry_adder</a> . . . . .	7



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

[ripple\\_carry\\_multiplication.vhd](#)

Moltiplicatore che effettua il prodotto come somma di righe della matrice dei prodotti parziali . 9





## Chapter 3

# Class Documentation

### 3.1 full\_adder Entity Reference

#### Libraries

- [IEEE](#)

#### Use Clauses

- [STD\\_LOGIC\\_1164](#)

#### Ports

- **x in STD\_LOGIC**  
*full\_adder input : addendo*
- **y in STD\_LOGIC**  
*full\_adder input : addendo*
- **c\_in in STD\_LOGIC**  
*full\_adder input : carry in ingresso*
- **s out STD\_LOGIC**  
*full\_adder output : somma*
- **c\_out out STD\_LOGIC**  
*full\_adder output : carry*

#### 3.1.1 Detailed Description

Descrizione Somma i 3 bit in ingresso (2 addendi e 1 carry in ingresso).  
In uscita abbiamo il risultato della somma sul bit S e il riporto sul bit C.

#### 3.1.2 Member Data Documentation

### 3.1.2.1 IEEE

[IEEE](#) [Library]

FEDERICO II , CORSO DI ASE 18/19, Gruppo 14 –

### 3.1.2.2 STD\_LOGIC\_1164

[STD\\_LOGIC\\_1164](#) [Package]

last changes: <11/11/2018> <15/10/2018> <log> Aggiunta doc doxygen

The documentation for this class was generated from the following file:

- [full\\_adder.vhd](#)

## 3.2 ripple\_carry\_multiplier Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

- [STD\\_LOGIC\\_1164](#)

### Generics

- **N integer:= 128**  
*parallelismo moltiplicando*
- **M integer:= 128**  
*parallelismo moltiplicatore*

### Ports

- **A** in STD\_LOGIC\_VECTOR(**N** - 1 downto 0 )
- **B** in STD\_LOGIC\_VECTOR(**M** - 1 downto 0 )
- **P** out STD\_LOGIC\_VECTOR(**N** + **M** - 1 downto 0 )

### 3.2.1 Detailed Description

il ripple carry multiplier effettua il prodotto come somma di righe utilizzando più RCA in cascata per sommare ogni riga della matrice dei prodotti parziali

The documentation for this class was generated from the following file:

- [ripple\\_carry\\_multiplication.vhd](#)

### 3.3 ripple\_carry\_multiplier\_testbench Entity Reference

#### Libraries

- [ieee](#)

#### Use Clauses

- [std\\_logic\\_1164](#)
- [numeric\\_std](#)

#### Generics

- **N NATURAL:= 8**  
*parallelismo primo operando*
- **M NATURAL:= 8**  
*parallelismo secondo operando*

The documentation for this class was generated from the following file:

- TSB\_mul\_sum\_row.vhd

### 3.4 rippleCarry\_adder Entity Reference

#### Libraries

- [IEEE](#)  
*i restanti M bit del prodotto sono ricavati dalle somme parziali sull'ultima riga*

#### Use Clauses

- [STD\\_LOGIC\\_1164](#)

#### Generics

- **width NATURAL:= 8**

#### Ports

- **X** in [STD\\_LOGIC\\_VECTOR](#)(**width** - 1 downto 0 )
- **Y** in [STD\\_LOGIC\\_VECTOR](#)(**width** - 1 downto 0 )
- **c\_in** in [STD\\_LOGIC](#)
- **S** out [STD\\_LOGIC\\_VECTOR](#)(**width** - 1 downto 0 )
- **c\_out** out [STD\\_LOGIC](#)  
*rippleCarry\_adder output: carry*

### 3.4.1 Detailed Description

Descrizione Somma le 2 stringe di bit in ingresso (2 addendi ) e 1 bit (carry in ingresso). Caratterizzato da una serie di [full\\_adder](#) in cascata che propagano il riporto.

In uscita abbiamo il risultato della somma sul bit S e il riporto sul bit C.

### 3.4.2 Member Data Documentation

#### 3.4.2.1 c\_in

[c\\_in](#) in STD\_LOGIC [Port]

[rippleCarry\\_adder](#) input: addendo

#### 3.4.2.2 c\_out

[c\\_out](#) out STD\_LOGIC [Port]

[rippleCarry\\_adder](#) output: carry

[rippleCarry\\_adder](#) output: somma

#### 3.4.2.3 S

[S](#) out STD\_LOGIC\_VECTOR([width](#) - 1 downto 0 ) [Port]

[rippleCarry\\_adder](#) input : carry in ingresso

#### 3.4.2.4 STD\_LOGIC\_1164

[STD\\_LOGIC\\_1164](#) [Package]

last changes: <11/11/2018> <15/10/2018> <log> Aggiunta doc doxygen

#### 3.4.2.5 width

[width](#) NATURAL:= 8 [Generic]

usato per definire il parallelismo del [rippleCarry\\_adder](#)

#### 3.4.2.6 Y

[Y](#) in STD\_LOGIC\_VECTOR([width](#) - 1 downto 0 ) [Port]

[rippleCarry\\_adder](#) input: addendo

The documentation for this class was generated from the following file:

- [rippleCarry\\_adder.vhd](#)

## Chapter 4

# File Documentation

### 4.1 ripple\_carry\_multiplication.vhd File Reference

Moltiplicatore che effettua il prodotto come somma di righe della matrice dei prodotti parziali.

#### Entities

- [ripple\\_carry\\_multiplier](#) entity

#### 4.1.1 Detailed Description

Moltiplicatore che effettua il prodotto come somma di righe della matrice dei prodotti parziali.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings



# Index

- c\_in
  - rippleCarry\_adder, [8](#)
- c\_out
  - rippleCarry\_adder, [8](#)
- full\_adder, [5](#)
  - IEEE, [5](#)
  - STD\_LOGIC\_1164, [6](#)
- IEEE
  - full\_adder, [5](#)
- ripple\_carry\_multiplication.vhd, [9](#)
- ripple\_carry\_multiplier, [6](#)
- ripple\_carry\_multiplier\_testbench, [7](#)
- rippleCarry\_adder, [7](#)
  - c\_in, [8](#)
  - c\_out, [8](#)
  - S, [8](#)
  - STD\_LOGIC\_1164, [8](#)
  - width, [8](#)
  - Y, [8](#)
- S
  - rippleCarry\_adder, [8](#)
- STD\_LOGIC\_1164
  - full\_adder, [6](#)
  - rippleCarry\_adder, [8](#)
- width
  - rippleCarry\_adder, [8](#)
- Y
  - rippleCarry\_adder, [8](#)