

restoring

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	Accumulator_Quotient Entity Reference . . . . .	5
3.1.1	Detailed Description . . . . .	6
3.1.2	Member Data Documentation . . . . .	6
3.1.2.1	IEEE . . . . .	6
3.1.2.2	STD_LOGIC_1164 . . . . .	6
3.2	AnodeManager Entity Reference . . . . .	6
3.3	carrySelect_adder Entity Reference . . . . .	7
3.4	carrySelect_addSub Entity Reference . . . . .	7
3.5	carrySelect_cell Entity Reference . . . . .	8
3.6	CathodeCoder Entity Reference . . . . .	9
3.7	CathodeManager Entity Reference . . . . .	9
3.8	Control_Unit Entity Reference . . . . .	9
3.9	counter_mod_2n Entity Reference . . . . .	10
3.10	counter_UpMod2n_Re_Ar Entity Reference . . . . .	11
3.10.1	Detailed Description . . . . .	12
3.10.2	Member Data Documentation . . . . .	12
3.10.2.1	IEEE . . . . .	12

3.10.2.2	STD_LOGIC_1164 . . . . .	12
3.11	counter_UpN_Re_Sr Entity Reference . . . . .	12
3.12	demux1_2 Entity Reference . . . . .	13
3.13	demux1_4 Entity Reference . . . . .	13
3.14	DivisorOnBoard Entity Reference . . . . .	14
3.15	edge_trigger_dn Entity Reference . . . . .	14
3.16	flipflop_d_risingEdge_asyncReset Entity Reference . . . . .	15
3.16.1	Detailed Description . . . . .	16
3.16.2	Member Data Documentation . . . . .	16
3.16.2.1	STD_LOGIC_1164 . . . . .	16
3.17	flipflopmux Entity Reference . . . . .	16
3.17.1	Detailed Description . . . . .	17
3.18	full_adder Entity Reference . . . . .	17
3.19	half_adder Entity Reference . . . . .	18
3.20	mux2_1 Entity Reference . . . . .	18
3.20.1	Detailed Description . . . . .	19
3.20.2	Member Data Documentation . . . . .	19
3.20.2.1	IEEE . . . . .	19
3.20.2.2	STD_LOGIC_1164 . . . . .	19
3.21	mux4_1 Entity Reference . . . . .	19
3.22	muxn_1 Entity Reference . . . . .	20
3.23	NibbleSelector Entity Reference . . . . .	20
3.24	overflow_checker Entity Reference . . . . .	21
3.24.1	Detailed Description . . . . .	21
3.24.2	Member Data Documentation . . . . .	21
3.24.2.1	IEEE . . . . .	21
3.24.2.2	STD_LOGIC_1164 . . . . .	22
3.25	R_Division Entity Reference . . . . .	22
3.25.1	Detailed Description . . . . .	22
3.25.2	Member Data Documentation . . . . .	22

3.25.2.1	IEEE	23
3.25.2.2	STD_LOGIC_1164	23
3.26	reg_clock Entity Reference	23
3.27	register_d_Re_Ar Entity Reference	24
3.27.1	Detailed Description	24
3.27.2	Member Data Documentation	24
3.27.2.1	IEEE	25
3.27.2.2	STD_LOGIC_1164	25
3.28	restoring_divider Entity Reference	25
3.29	restoring_divider_tb Entity Reference	26
3.29.1	Detailed Description	26
3.30	ripple_carry_adder Entity Reference	26
3.31	rippleCarry_adder Entity Reference	27
3.31.1	Detailed Description	27
3.31.2	Member Data Documentation	27
3.31.2.1	c_in	27
3.31.2.2	c_out	27
3.31.2.3	S	28
3.31.2.4	STD_LOGIC_1164	28
3.31.2.5	width	28
3.31.2.6	Y	28
3.32	scan_chain Entity Reference	28
3.32.1	Detailed Description	29
3.33	Scan_chain Entity Reference	29
3.34	Scan_component Entity Reference	30
3.35	Seven_Segments_Display Entity Reference	30
3.36	tb_DivisorOnBoard Entity Reference	31
3.36.1	Detailed Description	31
3.37	TSB_R_div Entity Reference	32
3.37.1	Detailed Description	32

<b>4 File Documentation</b>	<b>33</b>
4.1 AnodeManager.vhd File Reference . . . . .	33
4.1.1 Detailed Description . . . . .	33
4.2 carrySelect_addSub.vhd File Reference . . . . .	33
4.2.1 Detailed Description . . . . .	34
4.3 carrySelect_cell.vhd File Reference . . . . .	34
4.3.1 Detailed Description . . . . .	34
4.4 CathodeCoder.vhd File Reference . . . . .	35
4.4.1 Detailed Description . . . . .	35
4.5 CathodeManager.vhd File Reference . . . . .	35
4.5.1 Detailed Description . . . . .	35
4.6 Control_Unit.vhd File Reference . . . . .	36
4.6.1 Detailed Description . . . . .	36
4.7 Counter_Hit2n.vhd File Reference . . . . .	36
4.7.1 Detailed Description . . . . .	36
4.8 counter_modn.vhd File Reference . . . . .	37
4.8.1 Detailed Description . . . . .	37
4.9 counter_UpN_Re_Sr.vhd File Reference . . . . .	37
4.9.1 Detailed Description . . . . .	37
4.10 demux1_2.vhd File Reference . . . . .	38
4.10.1 Detailed Description . . . . .	38
4.11 demux1_4.vhd File Reference . . . . .	38
4.11.1 Detailed Description . . . . .	38
4.12 DivisorOnBoard.vhd File Reference . . . . .	39
4.12.1 Detailed Description . . . . .	39
4.13 flipflopmux.vhd File Reference . . . . .	39
4.13.1 Detailed Description . . . . .	39
4.14 full_adder.vhd File Reference . . . . .	40
4.14.1 Detailed Description . . . . .	40
4.15 half_adder.vhd File Reference . . . . .	40

4.15.1 Detailed Description . . . . .	40
4.16 latch_d.vhd File Reference . . . . .	41
4.16.1 Detailed Description . . . . .	41
4.17 mux4_1.vhd File Reference . . . . .	41
4.17.1 Detailed Description . . . . .	41
4.18 Nibble_Selector.vhd File Reference . . . . .	42
4.18.1 Detailed Description . . . . .	42
4.19 R_Division.vhd File Reference . . . . .	42
4.19.1 Detailed Description . . . . .	42
4.20 reg_clock.vhd File Reference . . . . .	43
4.20.1 Detailed Description . . . . .	43
4.21 ripple_carry_adder.vhd File Reference . . . . .	43
4.21.1 Detailed Description . . . . .	43
4.22 Scan_chain.vhd File Reference . . . . .	44
4.22.1 Detailed Description . . . . .	44
4.23 scan_chain.vhd File Reference . . . . .	44
4.23.1 Detailed Description . . . . .	44
4.24 Scan_component.vhd File Reference . . . . .	45
4.24.1 Detailed Description . . . . .	45
4.25 Seven_Segment_Display.vhd File Reference . . . . .	45
4.25.1 Detailed Description . . . . .	45
<b>Index</b>	<b>47</b>





# Chapter 1

## Class Index

### 1.1 Class List

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

entity <a href="#">Accumulator_Quotient</a>	
Uncomment the following library declaration if instantiating any Xilinx primitives in this code	5
entity <a href="#">AnodeManager</a>	6
entity <a href="#">carrySelect_adder</a>	7
entity <a href="#">carrySelect_addSub</a>	7
entity <a href="#">carrySelect_cell</a>	8
entity <a href="#">CathodeCoder</a>	9
entity <a href="#">CathodeManager</a>	9
entity <a href="#">Control_Unit</a>	9
entity <a href="#">counter_mod_2n</a>	10
entity <a href="#">counter_UpMod2n_Re_Ar</a>	11
entity <a href="#">counter_UpN_Re_Sr</a>	12
entity <a href="#">demux1_2</a>	13
entity <a href="#">demux1_4</a>	13
entity <a href="#">DivisorOnBoard</a>	14
entity <a href="#">edge_trigger_dn</a>	14
entity <a href="#">flipflop_d_risingEdge_asyncReset</a>	
Flipflop_d_risingEdge_asyncReset implementa un flipflop di tipo d che commuta sul fronte di salita, con segnale di enable e reset asincrono	15
entity <a href="#">flipflopmux</a>	16
entity <a href="#">full_adder</a>	17
entity <a href="#">half_adder</a>	18
entity <a href="#">mux2_1</a>	
Definisco il componente e la sua interfaccia	18
entity <a href="#">mux4_1</a>	19
entity <a href="#">muxn_1</a>	20
entity <a href="#">NibbleSelector</a>	20
entity <a href="#">overflow_checker</a>	21
entity <a href="#">R_Division</a>	
Uncomment the following library declaration if instantiating any Xilinx primitives in this code	22
entity <a href="#">reg_clock</a>	23
entity <a href="#">register_d_Re_Ar</a>	
Registro di dimensione "width" che prende in ingresso un dato D e lo memorizza	24
entity <a href="#">restoring_divider</a>	25
entity <a href="#">restoring_divider_tb</a>	26

entity <a href="#">ripple_carry_adder</a> . . . . .	26
entity <a href="#">rippleCarry_adder</a> . . . . .	27
entity <a href="#">scan_chain</a> . . . . .	28
entity <a href="#">Scan_chain</a> . . . . .	29
entity <a href="#">Scan_component</a> . . . . .	30
entity <a href="#">Seven_Segments_Display</a> . . . . .	30
entity <a href="#">tb_DivisorOnBoard</a> . . . . .	31
entity <a href="#">TSB_R_div</a> . . . . .	32

## Chapter 2

# File Index

### 2.1 File List

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

<a href="#">AnodeManager.vhd</a>	Componente per gestire gli anodi di un display a 7 segmenti . . . . .	33
<a href="#">carrySelect_addSub.vhd</a>	Sommatore Carry Select in grado di effettuare anche l'operazione di sottrazione . . . . .	33
<a href="#">carrySelect_cell.vhd</a>	Singolo blocco di un sommatore carry Select . . . . .	34
<a href="#">CathodeCoder.vhd</a>	Componente che codifica i catodi di un display a 7 segmenti . . . . .	35
<a href="#">CathodeManager.vhd</a>	Componente che permette di gestire i catodi di un display a 7 segmenti . . . . .	35
<a href="#">Control_Unit.vhd</a>	Unità di controllo della divisione restoring . . . . .	36
<a href="#">Counter_Hit2n.vhd</a>	Contatore modulo 2 alla N . . . . .	36
<a href="#">counter_modn.vhd</a>	Contatore modulo 2 alla N . . . . .	37
<a href="#">counter_UpN_Re_Sr.vhd</a>	Contatore modulo N . . . . .	37
<a href="#">demux1_2.vhd</a>	Demultiplexer 1 ingresso 2 uscite . . . . .	38
<a href="#">demux1_4.vhd</a>	Demultiplexer 1 ingresso 4 uscite . . . . .	38
<a href="#">DivisorOnBoard.vhd</a>	Componente di alto livello per implementare il divisore restoring su board . . . . .	39
<a href="#">flipflopmux.vhd</a>	Flip flop D con multiplexer . . . . .	39
<a href="#">full_adder.vhd</a>	Sommatore Full-Adder 3 ingressi 2 uscite . . . . .	40
<a href="#">half_adder.vhd</a>	Sommatore <a href="#">half_adder</a> 2 ingressi 2 uscite . . . . .	40
<a href="#">latch_d.vhd</a>	Latch D . . . . .	41
<a href="#">mux4_1.vhd</a>	Multiplexer con 4 ingressi e 1 uscita . . . . .	41
<a href="#">Nibble_Selector.vhd</a>	Componente che seleziona il nibble corretto . . . . .	42

<a href="#">R_Division.vhd</a>	
Componente di alto livello che effettua la divisione Restoring . . . . .	42
<a href="#">reg_clock.vhd</a>	
Registro clockato . . . . .	43
<a href="#">ripple_carry_adder.vhd</a>	
Sommatore ripple carry adder . . . . .	43
<a href="#">Scan_chain.vhd</a>	
Registro di n flip flop D multiplexati . . . . .	44
<a href="#">scan_chain.vhd</a>	
Registro di n flip flop D multiplexati . . . . .	44
<a href="#">Scan_component.vhd</a>	
Implementazione di una scan chain . . . . .	45
<a href="#">Seven_Segment_Display.vhd</a>	
Display a 7 segmenti . . . . .	45

## Chapter 3

# Class Documentation

### 3.1 Accumulator\_Quotient Entity Reference

Uncomment the following library declaration if instantiating any Xilinx primitives in this code.

#### Libraries

- [IEEE](#)

#### Use Clauses

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

#### Generics

- [N](#) integer:= 8
- [M](#) integer:= 8

#### Ports

- [High\\_Reg](#) in [STD\\_LOGIC\\_VECTOR\(N- 1 downto 0 \)](#)  
*ha dimensione n+1 perch l'algoritmo di divisione nel primo passo richiede uno shift per fare spazio al quoziente in posizione Q(0)*
- [shift](#) in [std\\_logic](#)
- [Low\\_Reg](#) in [STD\\_LOGIC\\_VECTOR\(m- 1 downto 0 \)](#)
- [H\\_read](#) out [STD\\_LOGIC\\_VECTOR\(N- 1 downto 0 \)](#)
- [L\\_read](#) out [STD\\_LOGIC\\_VECTOR\(m- 1 downto 0 \)](#)
- [clk](#) in [std\\_logic](#)
- [en\\_a](#) in [std\\_logic](#)
- [en\\_q](#) in [std\\_logic](#)
- [reset\\_a\\_n](#) in [std\\_logic](#)
- [reset\\_q\\_n](#) in [std\\_logic](#)
- [Sign\\_A](#) out [std\\_logic](#)
- [Q\\_0](#) in [STD\\_LOGIC](#)

### 3.1.1 Detailed Description

Uncomment the following library declaration if instantiating any Xilinx primitives in this code.

Uncomment the following library declaration if using arithmetic functions with Signed or Unsigned values

### 3.1.2 Member Data Documentation

#### 3.1.2.1 IEEE

[IEEE](#) [Library]

Company: Engineer:

Create Date: 10:34:23 01/23/2018 Design Name: Module Name: [Accumulator\\_Quotient](#) - Behavioral Project  
Name: Target Devices: Tool versions: Description:

#### 3.1.2.2 STD\_LOGIC\_1164

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

Revision: Revision 0.01 - File Created Additional Comments:

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

- [Accumulator\\_Quotient.vhd](#)

## 3.2 AnodeManager Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [counter](#) in [STD\\_LOGIC\\_VECTOR](#)( [1](#) downto [0](#) )
- [enable](#) in [STD\\_LOGIC\\_VECTOR](#)( [3](#) downto [0](#) )
- [anodes](#) out [STD\\_LOGIC\\_VECTOR](#)( [3](#) downto [0](#) )

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

- [AnodeManager.vhd](#)

### 3.3 carrySelect\_adder Entity Reference

#### Libraries

- [IEEE](#)

#### Use Clauses

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

#### Generics

- **M NATURAL:= 4**  
*M parallelismo dei ripplecarry adder.*
- **P NATURAL:= 2**  
*P parallelismo delle celle dell carry select Come metto M e P, marco e co fanno la stima dei tempi e mettono solo (M\*P) da cui ricavano poi M e P io direi di fare una versione con M e P espliciti e una versione come l'hanno fatta loro, ma su quella.*

#### Ports

- **A in STD\_LOGIC\_VECTOR(((M \*P)- 1 )downto 0 )**  
*input addendo*
- **B in STD\_LOGIC\_VECTOR(((M \*P)- 1 )downto 0 )**  
*input addendo*
- **c\_in in STD\_LOGIC**  
*input carry in ingresso*
- **S out STD\_LOGIC\_VECTOR(((M \*P)- 1 )downto 0 )**  
*output somma*
- **c\_out out STD\_LOGIC**  
*output carry in uscita*

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

- carrySelect\_adder.vhd

### 3.4 carrySelect\_addSub Entity Reference

#### Libraries

- [IEEE](#)

#### Use Clauses

- [STD\\_LOGIC\\_1164](#)
- [math\\_real](#)
- [numeric\\_std](#)

## Generics

- **M NATURAL:= 4**
- **P NATURAL:= 2**  
*P parallelismo delle celle dell carry select.*

## Ports

- **A in STD\_LOGIC\_VECTOR(((M\*P)-1)downto 0 )**  
*input addendo*
- **B in STD\_LOGIC\_VECTOR(((M\*P)-1)downto 0 )**  
*input addendo*
- **subtract in STD\_LOGIC**
- **S out STD\_LOGIC\_VECTOR(((M\*P)-1)downto 0 )**  
*output somma*
- **overflow out STD\_LOGIC**
- **c\_out out STD\_LOGIC**  
*output carry in uscita*

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

- [carrySelect\\_addSub.vhd](#)

## 3.5 carrySelect\_cell Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- **width NATURAL:= 4**

### Ports

- **A in STD\_LOGIC\_VECTOR((width-1)downto 0 )**
- **B 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**

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

- [carrySelect\\_cell.vhd](#)



## 3.6 CathodeCoder Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [nibble](#) in `STD_LOGIC_VECTOR( 3 downto 0 )`
- [cathodes](#) out `STD_LOGIC_VECTOR( 6 downto 0 )`

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

- [CathodeCoder.vhd](#)

## 3.7 CathodeManager Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [counter](#) in `STD_LOGIC_VECTOR( 1 downto 0 )`
- [nibbles](#) in `STD_LOGIC_VECTOR( 15 downto 0 )`
- [dots](#) in `STD_LOGIC_VECTOR( 3 downto 0 )`
- [cathodes](#) out `STD_LOGIC_VECTOR( 7 downto 0 )`

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

- [CathodeManager.vhd](#)

## 3.8 Control\_Unit Entity Reference

### Libraries

- [IEEE](#)

## Use Clauses

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

## Ports

- [counter\\_hit](#) in **STD\_LOGIC**  
*valore del contatore*
- [sel\\_q0](#) out **std\_logic**  
*segnale di selezione del multiplexer: se 0,  $Q[0] = X(0)$ , se 1,  $Q[0] = \text{not } A(N)$*
- [count\\_in](#) out **STD\_LOGIC**
- [S](#) in **STD\_LOGIC**  
*Risultato confronto/sottrazione (  $0 \Rightarrow 2R > V$ ,  $1 \Rightarrow 2R < V$  ) "segno di A".*
- [subtract](#) out **STD\_LOGIC**  
*decide se l'addizionatore deve effettuare somma o sottrazione*
- [reset\\_n](#) in **STD\_LOGIC**
- [en\\_a](#) out **STD\_LOGIC**
- [en\\_q](#) out **STD\_LOGIC**
- [en\\_q0](#) out **std\_logic**
- [reset\\_a\\_n](#) out **std\_logic**
- [shift](#) out **std\_logic**
- [start\\_division](#) in **STD\_LOGIC**  
*segnale di start della divisione*
- [stop\\_division](#) out **STD\_LOGIC**  
*segnale di stop della divisione*
- [clock](#) in **STD\_LOGIC**
- [reset\\_counter\\_n](#) out **STD\_LOGIC**

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

- [Control\\_Unit.vhd](#)

## 3.9 counter\_mod\_2n Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- [N natural](#):= 2

## Ports

- `reset_n` in STD\_LOGIC
- `en` in STD\_LOGIC
- `clk` in STD\_LOGIC
- `count` out STD\_LOGIC\_VECTOR(N-1 downto 0)
- `count_hit` out STD\_LOGIC

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

- [counter\\_modn.vhd](#)

## 3.10 counter\_UpMod2n\_Re\_Ar Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

- [STD\\_LOGIC\\_1164](#)
- [numeric\\_std](#)
- [math\\_real](#)

### Generics

- `n` NATURAL:= 1  
*usato per definire il valore massimo (2\*n)-1 di fine conteggio.*
- `enable_level` std\_logic:= '1'  
*definisce il livello enable*

## Ports

- `enable` in STD\_LOGIC  
*counter\_UpMod2n\_Re\_Ar input: segnale enable*
- `reset_n` in STD\_LOGIC  
*counter\_UpMod2n\_Re\_Ar input: segnale reset*
- `clock` in STD\_LOGIC  
*counter\_UpMod2n\_Re\_Ar input: segnale di clock per sincronizzare*
- `count_hit` out STD\_LOGIC  
*counter\_UpMod2n\_Re\_Ar output: segnale di fine conteggio*
- `COUNTS` out STD\_LOGIC\_VECTOR((n-1)downto 0)  
*counter\_UpMod2n\_Re\_Ar output: conteggio in uscita*

### 3.10.1 Detailed Description

Conta il numero di impulsi che sono stati applicati in ingresso (sul fronte di salita del clock).  
Una volta raggiunto il valore massimo  $(2^{**n})-1$ , il conteggio riparte da 0.

### 3.10.2 Member Data Documentation

#### 3.10.2.1 IEEE

[IEEE](#) [Library]

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

#### 3.10.2.2 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:

- counter\_UpMod2n\_Re\_Ar.vhd

## 3.11 counter\_UpN\_Re\_Sr Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

- [STD\\_LOGIC\\_1164](#)
- [NUMERIC\\_STD](#)
- [STD\\_LOGIC\\_UNSIGNED](#)
- [numeric\\_std](#)
- [math\\_real](#)

### Generics

- [n](#) integer:= 4
- [enable\\_level](#) STD\_LOGIC:= '1'

### Ports

- [enable](#) in STD\_LOGIC  
*enable input*
- [reset\\_n](#) in STD\_LOGIC  
*reset input*
- [clock](#) in STD\_LOGIC  
*clock input*
- [count\\_hit](#) out STD\_LOGIC  
*count\_hit output*
- [COUNTS](#) out STD\_LOGIC\_VECTOR([n- 1](#) downto [0](#) )  
*COUNT output.*

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

- [Counter\\_Hit2n.vhd](#)
- [counter\\_UpN\\_Re\\_Sr.vhd](#)

## 3.12 demux1\_2 Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [SEL](#) in STD\_LOGIC
- [A](#) in STD\_LOGIC
- [X](#) out STD\_LOGIC\_VECTOR([1](#) downto [0](#) )

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

- [demux1\\_2.vhd](#)

## 3.13 demux1\_4 Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [SEL](#) in [STD\\_LOGIC\\_VECTOR](#)( [1](#) downto [0](#) )
- [A](#) in [STD\\_LOGIC](#)
- [X](#) out [STD\\_LOGIC\\_VECTOR](#)( [3](#) downto [0](#) )

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

- [demux1\\_4.vhd](#)

## 3.14 DivisorOnBoard Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [clk](#) in [STD\\_LOGIC](#)
- [reset](#) in [STD\\_LOGIC](#)
- [load\\_divisor](#) in [STD\\_LOGIC](#)
- [load\\_dividend](#) in [STD\\_LOGIC](#)
- [start\\_division](#) in [STD\\_LOGIC](#)
- [in\\_byte](#) in [STD\\_LOGIC\\_VECTOR](#)( [7](#) downto [0](#) )
- [anodes](#) out [STD\\_LOGIC\\_VECTOR](#)( [3](#) downto [0](#) )
- [cathodes](#) out [STD\\_LOGIC\\_VECTOR](#)( [7](#) downto [0](#) )

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

- [DivisorOnBoard.vhd](#)

## 3.15 edge\_trigger\_dn Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- [width](#) integer

### Ports

- [d](#) in [STD\\_LOGIC\\_vector](#)(width- 1 downto 0 )
- [en](#) in [STD\\_LOGIC](#)
- [reset\\_n](#) in [STD\\_LOGIC](#)
- [clk](#) in [STD\\_LOGIC](#)
- [q](#) out [STD\\_LOGIC\\_vector](#)(width- 1 downto 0 )

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

- [latch\\_d.vhd](#)

## 3.16 flipflop\_d\_risingEdge\_asyncReset Entity Reference

[flipflop\\_d\\_risingEdge\\_asyncReset](#) implementa un flipflop di tipo d che commuta sul fronte di salita, con segnale di enable e reset asincrono.

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- [init\\_value](#) [STD\\_LOGIC](#) := ' 0 '
- definisce il livello iniziale del flipflop*
- [reset\\_level](#) [STD\\_LOGIC](#) := ' 0 '
- definisce il livello reset*
- [enable\\_level](#) [STD\\_LOGIC](#) := ' 1 '
- definisce il livello enable*

## Ports

- **clock in STD\_LOGIC**  
*flipflop\_d\_risingEdge\_asyncReset* input : segnale di clock per sincronizzare
- **enable in STD\_LOGIC**  
*flipflop\_d\_risingEdge\_asyncReset* input : segnale enable
- **reset in STD\_LOGIC**  
*flipflop\_d\_risingEdge\_asyncReset* input : segnale reset
- **d in STD\_LOGIC**  
*flipflop\_d\_risingEdge\_asyncReset* input : input data
- **q out STD\_LOGIC**  
*flipflop\_d\_risingEdge\_asyncReset* output : output data

### 3.16.1 Detailed Description

*flipflop\_d\_risingEdge\_asyncReset* implementa un flipflop di tipo d che commuta sul fronte di salita, con segnale di enable e reset asincrono.

### 3.16.2 Member Data Documentation

#### 3.16.2.1 STD\_LOGIC\_1164

*STD\_LOGIC\_1164* [Package]

last changes: <14/11/2018> <13/11/2018> <log> create

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

- *flipflop\_d\_risingEdge\_asyncReset.vhd*

## 3.17 flipflopmux Entity Reference

### Libraries

- *IEEE*  
*architecture behavioural end*

### Use Clauses

- *STD\_LOGIC\_1164*



## Ports

- [clock](#) in STD\_LOGIC  
*clock*
- [en](#) in STD\_LOGIC  
*enable*
- [reset\\_n](#) in STD\_LOGIC  
*reset*
- [scan\\_en](#) in STD\_LOGIC  
*segnale di selezione del multiplexer per modalità (0 = normale, 1 = controllo)*
- [d](#) in STD\_LOGIC  
*ingresso del flipflop in modalità normale*
- [scan\\_in](#) in STD\_LOGIC  
*ingresso del flipflop in modalità controllo*
- [q](#) out STD\_LOGIC  
*uscita del flipflop*

### 3.17.1 Detailed Description

flipflopmux è un flip flop D con multiplexer: scan\_en è il segnale di controllo del multiplexer, se scan\_en = 0 l'ingresso è d, se scan\_en = 1 l'ingresso è scan\_in.

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

- [flipflopmux.vhd](#)

## 3.18 full\_adder Entity Reference

### Libraries

- [ieee](#)

### Use Clauses

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

## Ports

- [x](#) in std\_logic
- [y](#) in std\_logic
- [cin](#) in std\_logic
- [cout](#) out std\_logic
- [sum](#) out std\_logic

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

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

### 3.19 half\_adder Entity Reference

#### Libraries

- [ieee](#)

#### Use Clauses

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

#### Ports

- [x](#) in **std\_logic**
- [y](#) in **std\_logic**
- [s](#) out **std\_logic**
- [c](#) out **std\_logic**

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

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

### 3.20 mux2\_1 Entity Reference

definisco il componente e la sua interfaccia

#### Libraries

- [IEEE](#)

#### Use Clauses

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

#### Ports

- [SEL](#) in **STD\_LOGIC**  
*mux2\_1 input: selezione*
- [A](#) in **STD\_LOGIC**  
*mux2\_1 input: A*
- [B](#) in **STD\_LOGIC**  
*mux2\_1 input: B*
- [X](#) out **STD\_LOGIC**  
*mux2\_1 output: X*

### 3.20.1 Detailed Description

definisco il componente e la sua interfaccia

Descrizione Quando l'ingresso SEL è basso, l'uscita assume il valore del segnale A, altrimenti quando il segnale SEL è alto l'uscita assume il valore del segnale B.

### 3.20.2 Member Data Documentation

#### 3.20.2.1 IEEE

[IEEE](#) [Library]

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

#### 3.20.2.2 STD\_LOGIC\_1164

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

last changes: <14/11/2018> <13/11/2018> <log> create

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

- mux2\_1.vhd

## 3.21 mux4\_1 Entity Reference

### Libraries

- [IEEE](#)  
*architecture dataflow of [mux2\\_1](#) end*

### Use Clauses

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

### Ports

- [SEL](#) in [STD\\_LOGIC\\_VECTOR](#)( [1](#) downto [0](#) )
- [A](#) in [STD\\_LOGIC\\_VECTOR](#)( [3](#) downto [0](#) )
- [X](#) out [STD\\_LOGIC](#)

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

- [mux4\\_1.vhd](#)

## 3.22 muxn\_1 Entity Reference

### Libraries

- [ieee](#)

### Use Clauses

- [std\\_logic\\_1164](#)
- [STD\\_LOGIC\\_UNSIGNED](#)

*ieee=synopsys (lanciarla con questa opzione: ghdl -a ieee=synopsys mux1\_n.vhd // ghdl -r -ieee=synopsys mux1↔  
\_n\_tsb -vcd=mux1\_n.vcd*

### Generics

- [width](#) **natural**

### Ports

- [inputs](#) in **std\_logic\_vector( 2\*\*width- 1 downto 0 )**
- [selectors](#) in **std\_logic\_vector(width- 1 downto 0 )**
- [output](#) out **std\_logic**

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

- mux1\_n.vhd

## 3.23 NibbleSelector Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [nibbles](#) in **STD\_LOGIC\_VECTOR( 15 downto 0 )**
- [counter](#) in **STD\_LOGIC\_VECTOR( 1 downto 0 )**
- [nibbleSelected](#) out **STD\_LOGIC\_VECTOR( 3 downto 0 )**

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

- [Nibble\\_Selector.vhd](#)

## 3.24 overflow\_checker Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Ports

- [a](#) in **STD\_LOGIC**  
*bit più significativo (segno) di A*
- [b](#) in **STD\_LOGIC**  
*bit più significativo (segno) di B*
- [subtract](#) in **STD\_LOGIC**  
*bit di operazione: 1 se sottrazione, 0 se addizione*
- [s](#) in **STD\_LOGIC**  
*bit più significativo (segno) di S*
- [overflow](#) out **STD\_LOGIC**  
*bit alto se ho una condizione di overflow*

### 3.24.1 Detailed Description

Descrizione La macchina controlla se vi è overflow nel risultato confrontando le cifre più significative (segno) dei due operandi e del risultato con subtract. Ho overflow in caso di:

- somma di due positivi con risultato negativo
- somma di due negativi con risultato positivo
- differenza di positivo e negativo con risultato negativo
- differenza di negativo e positivo con risultato positivo

### 3.24.2 Member Data Documentation

#### 3.24.2.1 IEEE

[IEEE](#) [Library]

### 3.24.2.2 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:

- `overflow_checker.vhd`

## 3.25 R\_Division Entity Reference

Uncomment the following library declaration if instantiating any Xilinx primitives in this code.

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- `n integer:= 4`

### Ports

- `Dividend` in `STD_LOGIC_VECTOR( 2 *n- 1 downto 0 )`
- `Divisor` in `STD_LOGIC_VECTOR( 2 *n- 1 downto 0 )`
- `Quotient` out `STD_LOGIC_VECTOR( 2 *n- 1 downto 0 )`
- `Remainder` out `STD_LOGIC_VECTOR( 2 *n- 1 downto 0 )`
- `Reset_n` in `STD_LOGIC`
- `clock` in `std_logic`
- `start` in `STD_LOGIC`

### 3.25.1 Detailed Description

Uncomment the following library declaration if instantiating any Xilinx primitives in this code.

Uncomment the following library declaration if using arithmetic functions with Signed or Unsigned values

### 3.25.2 Member Data Documentation

## 3.25.2.1 IEEE

[IEEE](#) [Library]

Company: Engineer:

Create Date: 13:13:13 01/31/2018 Design Name: Module Name: NR\_Division - Strucutal Project Name: Target  
Devices: Tool versions: Description:

## 3.25.2.2 STD\_LOGIC\_1164

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

Revision: Revision 0.01 - File Created Additional Comments:

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

- [NR\\_Division.vhd](#)

## 3.26 reg\_clock Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- [width](#) integer:= 8

### Ports

- [value](#) in [STD\\_LOGIC\\_VECTOR](#)(width- 1 downto 0 )
- [clock](#) in [STD\\_LOGIC](#)
- [enable](#) in [STD\\_LOGIC](#)
- [reset\\_n](#) in [STD\\_LOGIC](#)
- [output](#) out [STD\\_LOGIC\\_VECTOR](#)(width- 1 downto 0 )

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

- [reg\\_clock.vhd](#)

## 3.27 register\_d\_Re\_Ar Entity Reference

Registro di diensione "width" che prende in ingresso un dato D e lo memorizza.

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- **width** **NATURAL** := **8**  
*definisce il parallelismo del registro*
- **reset\_level** **STD\_LOGIC** := '**0**'  
*definisce il livello reset*
- **enable\_level** **STD\_LOGIC** := '**1**'  
*definisce il livello enable*

### Ports

- **clock** in **STD\_LOGIC**  
*register\_d\_Re\_Ar input : segnale di clock per sincronizzare*
- **enable** in **STD\_LOGIC**  
*register\_d\_Re\_Ar input : segnale enable*
- **reset** in **STD\_LOGIC**  
*register\_d\_Re\_Ar input : segnale reset*
- **d** in **STD\_LOGIC\_VECTOR**(**width** - **1** downto **0**)  
*register\_d\_Re\_Ar input : inpput data*
- **q** out **STD\_LOGIC\_VECTOR**(**width** - **1** downto **0**)  
*register\_d\_Re\_Ar input : output data*

### 3.27.1 Detailed Description

Registro di diensione "width" che prende in ingresso un dato D e lo memorizza.

### 3.27.2 Member Data Documentation



## 3.27.2.1 IEEE

[IEEE](#) [Library]

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

## 3.27.2.2 STD\_LOGIC\_1164

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

last changes: <16/11/2018> <16/11/2018> <log> create

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

- [register\\_d\\_Re\\_Ar.vhd](#)

## 3.28 restoring\_divider Entity Reference

### Libraries

- [IEEE](#)  
*architecture behavioural of [overflow\\_checker](#) end*

### Use Clauses

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

### Generics

- [n](#) integer:= **8**

### Ports

- [X](#) in **STD\_LOGIC\_VECTOR**(n- **1** downto **0** )  
*dividendo della divisione*
- [Y](#) in **STD\_LOGIC\_VECTOR**(n- **1** downto **0** )  
*divisore della divisione*
- [Q](#) out **STD\_LOGIC\_VECTOR**(ndownto **0** )  
*quoziente della divisione*
- [R](#) out **STD\_LOGIC\_VECTOR**(n- **1** downto **0** )  
*resto della divisione*
- [Reset\\_n](#) in **STD\_LOGIC**
- [clock](#) in **std\_logic**
- [start](#) in **STD\_LOGIC**  
*alto quando inizia la moltiplicazione*
- [stop](#) out **STD\_LOGIC**  
*alto quando il risultato è pronto*

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

- [R\\_Division.vhd](#)

## 3.29 restoring\_divider\_tb Entity Reference

### Libraries

- [ieee](#)  
*architecture behavioral of [register\\_d\\_Re\\_Ar](#) end*

### Use Clauses

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

### Generics

- [n](#) integer := 6

### 3.29.1 Detailed Description

Uncomment the following library declaration if using arithmetic functions with Signed or Unsigned values

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

- [restoring\\_divider\\_tb.vhd](#)

## 3.30 ripple\_carry\_adder Entity Reference

### Libraries

- [ieee](#)

### Use Clauses

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

### Generics

- [width](#) natural

### Ports

- [X](#) in [std\\_logic\\_vector](#)(width- 1 downto 0 )
- [Y](#) in [std\\_logic\\_vector](#)(width- 1 downto 0 )
- [cin](#) in [std\\_logic](#)
- [cout](#) out [std\\_logic](#)
- [sum](#) out [std\\_logic\\_vector](#)(width- 1 downto 0 )

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

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

## 3.31 rippleCarry\_adder Entity Reference

### Libraries

- IEEE

### 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

#### 3.31.1 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.

#### 3.31.2 Member Data Documentation

##### 3.31.2.1 c\_in

*c\_in* in STD\_LOGIC [Port]

*rippleCarry\_adder* input: addendo

##### 3.31.2.2 c\_out

*c\_out* out STD\_LOGIC [Port]

*rippleCarry\_adder* output: carry

*rippleCarry\_adder* output: somma

### 3.31.2.3 S

`S out STD_LOGIC_VECTOR(width - 1 downto 0 )` [Port]

`rippleCarry_adder` input : carry in ingresso

### 3.31.2.4 STD\_LOGIC\_1164

`STD_LOGIC_1164` [Package]

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

### 3.31.2.5 width

`width NATURAL:= 8` [Generic]

usato per definire il parallelismo del `rippleCarry_adder`

### 3.31.2.6 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`

## 3.32 scan\_chain Entity Reference

### Libraries

- `IEEE`

### Use Clauses

- `STD_LOGIC_1164`

### Generics

- `width integer:= 8`  
*dimensione del registro*
- `shift_direction std_logic:= ' 1 '`  
*shift a sinistra*

## Ports

- **clock** in STD\_LOGIC  
*segnale clock di tempificazione*
- **en** in STD\_LOGIC  
*segnale di abilitazione 1-attivo*
- **reset\_n** in STD\_LOGIC  
*segnale di reset 0-attivo*
- **scan\_en** in STD\_LOGIC  
*segnale di selezione modalità (0 = normale, 1 = controllo)*
- **scan\_in** in STD\_LOGIC  
*primo valore scan-in*
- **d\_reg** in STD\_LOGIC\_VECTOR(**width - 1** downto **0**)  
*valore in ingresso nel registro*
- **scan\_out** out STD\_LOGIC  
*ultimo valore scan-out*
- **q\_reg** out STD\_LOGIC\_VECTOR(**width - 1** downto **0**)  
*valore in uscita del registro*

### 3.32.1 Detailed Description

Scan chain è un registro di width flipflop D multiplexati. Quando scan\_en = 0, il componente si comporta come un normale registro. Quando scan\_en = 1, diventa uno shift register che shifta ad ogni colpo di clock. La direzione dello shift è regolata dal generic shift\_direction (0 = right, 1 = left)

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

- [scan\\_chain.vhd](#)

## 3.33 Scan\_chain Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- **scan\_lenght** natural := **4**
- **left\_shift\_n** std\_logic := '1'

## Ports

- [Din](#) in `STD_LOGIC_VECTOR(scan_lenght- 1 downto 0 )`
- [Scan\\_in](#) in `STD_LOGIC`
- [en](#) in `std_logic`
- [clock](#) in `std_logic`
- [Scan\\_en](#) in `STD_LOGIC`
- [Scan\\_out](#) out `std_logic`
- [reset\\_n](#) in `std_logic`
- [Dout](#) out `STD_LOGIC_VECTOR(scan_lenght- 1 downto 0 )`

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

- [Scan\\_chain.vhd](#)

## 3.34 Scan\_component Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

## Ports

- [Din](#) in `STD_LOGIC`
- [Scan\\_in](#) in `STD_LOGIC`
- [en](#) in `std_logic`
- [clock](#) in `std_logic`
- [Scan\\_en](#) in `STD_LOGIC`
- [reset\\_n](#) in `std_logic`
- [Dout](#) out `STD_LOGIC`

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

- [Scan\\_component.vhd](#)

## 3.35 Seven\_Segments\_Display Entity Reference

### Libraries

- [IEEE](#)

### Use Clauses

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

### Generics

- [clock\\_frequency\\_in](#) **natural**:= 50000000
- [clock\\_frequency\\_out](#) **natural**:= 250

### Ports

- [clk](#) in **STD\_LOGIC**
- [reset\\_n](#) in **STD\_LOGIC**
- [values](#) in **STD\_LOGIC\_VECTOR**( 15 downto 0 )
- [dots](#) in **STD\_LOGIC\_VECTOR**( 3 downto 0 )
- [enable](#) in **STD\_LOGIC\_VECTOR**( 3 downto 0 )
- [anodes](#) out **STD\_LOGIC\_VECTOR**( 3 downto 0 )
- [cathodes](#) out **STD\_LOGIC\_VECTOR**( 7 downto 0 )

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

- [Seven\\_Segment\\_Display.vhd](#)

## 3.36 tb\_DivisorOnBoard Entity Reference

### Libraries

- [ieee](#)

### Use Clauses

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

#### 3.36.1 Detailed Description

Uncomment the following library declaration if using arithmetic functions with Signed or Unsigned values

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

- [tb\\_DivisorOnBoard.vhd](#)

### 3.37 TSB\_R\_div Entity Reference

#### Libraries

- [ieee](#)

#### Use Clauses

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

#### 3.37.1 Detailed Description

Uncomment the following library declaration if using arithmetic functions with Signed or Unsigned values

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

- TSB\_R\_div.vhd



## Chapter 4

# File Documentation

### 4.1 AnodeManager.vhd File Reference

Componente per gestire gli anodi di un display a 7 segmenti.

#### Entities

- [AnodeManager](#) entity

#### 4.1.1 Detailed Description

Componente per gestire gli anodi di un display a 7 segmenti.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

### 4.2 carrySelect\_addSub.vhd File Reference

Sommatore Carry Select in grado di effettuare anche l'operazione di sottrazione.

## Entities

- [carrySelect\\_addSub](#) entity

### 4.2.1 Detailed Description

Sommatore Carry Select in grado di effettuare anche l'operazione di sottrazione.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.3 carrySelect\_cell.vhd File Reference

Singolo blocco di un sommatore carry Select.

## Entities

- [carrySelect\\_cell](#) entity

### 4.3.1 Detailed Description

Singolo blocco di un sommatore carry Select.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.4 CathodeCoder.vhd File Reference

Componente che codifica i catodi di un display a 7 segmenti.

### Entities

- [CathodeCoder](#) entity

### 4.4.1 Detailed Description

Componente che codifica i catodi di un display a 7 segmenti.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.5 CathodeManager.vhd File Reference

Componente che permette di gestire i catodi di un display a 7 segmenti.

### Entities

- [CathodeManager](#) entity

### 4.5.1 Detailed Description

Componente che permette di gestire i catodi di un display a 7 segmenti.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.6 Control\_Unit.vhd File Reference

Unità di controllo della divisione restoring.

### Entities

- [Control\\_Unit](#) entity

### 4.6.1 Detailed Description

Unità di controllo della divisione restoring.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.7 Counter\_Hit2n.vhd File Reference

Contatore modulo 2 alla N.

### Entities

- [counter\\_UpN\\_Re\\_Sr](#) entity

### 4.7.1 Detailed Description

Contatore modulo 2 alla N.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.8 counter\_modn.vhd File Reference

Contatore modulo 2 alla N.

### Entities

- [counter\\_mod\\_2n](#) entity

### 4.8.1 Detailed Description

Contatore modulo 2 alla N.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.9 counter\_UpN\_Re\_Sr.vhd File Reference

Contatore modulo N.

### Entities

- [counter\\_UpN\\_Re\\_Sr](#) entity

### 4.9.1 Detailed Description

Contatore modulo N.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.10 demux1\_2.vhd File Reference

Demultiplexer 1 ingresso 2 uscite.

### Entities

- [demux1\\_2](#) entity

### 4.10.1 Detailed Description

Demultiplexer 1 ingresso 2 uscite.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.11 demux1\_4.vhd File Reference

Demultiplexer 1 ingresso 4 uscite.

### Entities

- [demux1\\_4](#) entity

### 4.11.1 Detailed Description

Demultiplexer 1 ingresso 4 uscite.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.12 DivisorOnBoard.vhd File Reference

Componente di alto livello per implementare il divisore restoring su board.

### Entities

- [DivisorOnBoard](#) entity

### 4.12.1 Detailed Description

Componente di alto livello per implementare il divisore restoring su board.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.13 flipflopmux.vhd File Reference

flip flop D con multiplexer

### Entities

- [flipflopmux](#) entity

### 4.13.1 Detailed Description

flip flop D con multiplexer

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.14 full\_adder.vhd File Reference

Sommatore Full-Adder 3 ingressi 2 uscite.

### Entities

- [full\\_adder](#) entity

### 4.14.1 Detailed Description

Sommatore Full-Adder 3 ingressi 2 uscite.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.15 half\_adder.vhd File Reference

Sommatore [half\\_adder](#) 2 ingressi 2 uscite.

### Entities

- [half\\_adder](#) entity

### 4.15.1 Detailed Description

Sommatore [half\\_adder](#) 2 ingressi 2 uscite.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings



## 4.16 latch\_d.vhd File Reference

latch D

### Entities

- [edge\\_trigger\\_dn](#) entity

### 4.16.1 Detailed Description

latch D

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.17 mux4\_1.vhd File Reference

Multiplexer con 4 ingressi e 1 uscita.

### Entities

- [mux4\\_1](#) entity

### 4.17.1 Detailed Description

Multiplexer con 4 ingressi e 1 uscita.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.18 Nibble\_Selector.vhd File Reference

Componente che seleziona il nibble corretto.

### Entities

- [NibbleSelector](#) entity

### 4.18.1 Detailed Description

Componente che seleziona il nibble corretto.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.19 R\_Division.vhd File Reference

Componente di alto livello che effettua la divisione Restoring.

### Entities

- [restoring\\_divider](#) entity

### 4.19.1 Detailed Description

Componente di alto livello che effettua la divisione Restoring.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.20 reg\_clock.vhd File Reference

Registro clockato.

### Entities

- [reg\\_clock](#) entity

### 4.20.1 Detailed Description

Registro clockato.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.21 ripple\_carry\_adder.vhd File Reference

Sommatore ripple carry adder.

### Entities

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

### 4.21.1 Detailed Description

Sommatore ripple carry adder.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.22 Scan\_chain.vhd File Reference

Registro di n flip flop D multiplexati.

### Entities

- [Scan\\_chain](#) entity

### 4.22.1 Detailed Description

Registro di n flip flop D multiplexati.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.23 scan\_chain.vhd File Reference

Registro di n flip flop D multiplexati.

### Entities

- [scan\\_chain](#) entity

### 4.23.1 Detailed Description

Registro di n flip flop D multiplexati.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.24 Scan\_component.vhd File Reference

Implementazione di una scan chain.

### Entities

- [Scan\\_component](#) entity

### 4.24.1 Detailed Description

Implementazione di una scan chain.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings

## 4.25 Seven\_Segment\_Display.vhd File Reference

Display a 7 segmenti.

### Entities

- [Seven\\_Segments\\_Display](#) entity

### 4.25.1 Detailed Description

Display a 7 segmenti.

#### Author

Gabriele Previtera, Mirko Pennone, Simone Penna

#### Date

04/03/2019

#### Version

0.2

#### Dependencies:

Nothings



# Index

Accumulator\_Quotient, [5](#)  
    IEEE, [6](#)  
    STD\_LOGIC\_1164, [6](#)  
AnodeManager, [6](#)  
AnodeManager.vhd, [33](#)  
  
c\_in  
    rippleCarry\_adder, [27](#)  
c\_out  
    rippleCarry\_adder, [27](#)  
carrySelect\_addSub, [7](#)  
carrySelect\_addSub.vhd, [33](#)  
carrySelect\_adder, [7](#)  
carrySelect\_cell, [8](#)  
carrySelect\_cell.vhd, [34](#)  
CathodeCoder, [9](#)  
CathodeCoder.vhd, [35](#)  
CathodeManager, [9](#)  
CathodeManager.vhd, [35](#)  
Control\_Unit, [9](#)  
Control\_Unit.vhd, [36](#)  
Counter\_Hit2n.vhd, [36](#)  
counter\_UpMod2n\_Re\_Ar, [11](#)  
    IEEE, [12](#)  
    STD\_LOGIC\_1164, [12](#)  
counter\_UpN\_Re\_Sr, [12](#)  
counter\_UpN\_Re\_Sr.vhd, [37](#)  
counter\_mod\_2n, [10](#)  
counter\_modn.vhd, [37](#)  
  
demux1\_2, [13](#)  
demux1\_2.vhd, [38](#)  
demux1\_4, [13](#)  
demux1\_4.vhd, [38](#)  
DivisorOnBoard, [14](#)  
DivisorOnBoard.vhd, [39](#)  
  
edge\_trigger\_dn, [14](#)  
  
flipflop\_d\_risingEdge\_asyncReset, [15](#)  
    STD\_LOGIC\_1164, [16](#)  
flipflopmux, [16](#)  
flipflopmux.vhd, [39](#)  
full\_adder, [17](#)  
full\_adder.vhd, [40](#)  
  
half\_adder, [18](#)  
half\_adder.vhd, [40](#)  
  
IEEE  
    Accumulator\_Quotient, [6](#)

counter\_UpMod2n\_Re\_Ar, [12](#)  
mux2\_1, [19](#)  
overflow\_checker, [21](#)  
R\_Division, [22](#)  
register\_d\_Re\_Ar, [24](#)  
  
latch\_d.vhd, [41](#)  
  
mux2\_1, [18](#)  
    IEEE, [19](#)  
    STD\_LOGIC\_1164, [19](#)  
mux4\_1, [19](#)  
mux4\_1.vhd, [41](#)  
muxn\_1, [20](#)  
  
Nibble\_Selector.vhd, [42](#)  
NibbleSelector, [20](#)  
  
overflow\_checker, [21](#)  
    IEEE, [21](#)  
    STD\_LOGIC\_1164, [21](#)  
  
R\_Division, [22](#)  
    IEEE, [22](#)  
    STD\_LOGIC\_1164, [23](#)  
R\_Division.vhd, [42](#)  
reg\_clock, [23](#)  
reg\_clock.vhd, [43](#)  
register\_d\_Re\_Ar, [24](#)  
    IEEE, [24](#)  
    STD\_LOGIC\_1164, [25](#)  
restoring\_divider, [25](#)  
restoring\_divider\_tb, [26](#)  
ripple\_carry\_adder, [26](#)  
ripple\_carry\_adder.vhd, [43](#)  
rippleCarry\_adder, [27](#)  
    c\_in, [27](#)  
    c\_out, [27](#)  
    S, [27](#)  
    STD\_LOGIC\_1164, [28](#)  
    width, [28](#)  
    Y, [28](#)  
  
S  
    rippleCarry\_adder, [27](#)  
STD\_LOGIC\_1164  
    Accumulator\_Quotient, [6](#)  
    counter\_UpMod2n\_Re\_Ar, [12](#)  
    flipflop\_d\_risingEdge\_asyncReset, [16](#)  
    mux2\_1, [19](#)  
    overflow\_checker, [21](#)

- R\_Division, [23](#)
  - register\_d\_Re\_Ar, [25](#)
  - rippleCarry\_adder, [28](#)
- Scan\_chain, [29](#)
- scan\_chain, [28](#)
- Scan\_chain.vhd, [44](#)
- scan\_chain.vhd, [44](#)
- Scan\_component, [30](#)
- Scan\_component.vhd, [45](#)
- Seven\_Segment\_Display.vhd, [45](#)
- Seven\_Segments\_Display, [30](#)
- TSB\_R\_div, [32](#)
- tb\_DivisorOnBoard, [31](#)
- width
  - rippleCarry\_adder, [28](#)
- Y
  - rippleCarry\_adder, [28](#)