restoring

Corso di ASE anno 18/19

Gruppo 14
PREVITERA GABRIELE
PENNONE MIRKO
PENNA SIMONE

Contents

Chapter 1

Class Index

1.1 Class List

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

Chapter 2

File Index

2.1 File List

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

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] = not A(N)
- count_in **out STD_LOGIC**
- S **in STD_LOGIC**

  Risultato confronto/sottrazione (0 => 2R > V , 1=> 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]
```

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

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