# ADD - ACCELERATOR DESIGN AND DEPLOY DOCUMENTATION

1.0

Wed Aug 23 2017 19:15:14

# Contents

# Chapter 1

# Hierarchical Index

## 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

# Chapter 2

# Class Index

## 2.1 Class List

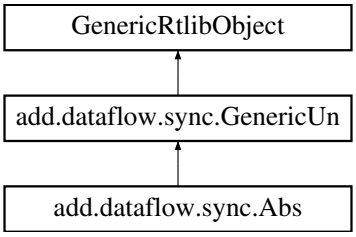
Here are the classes, structs, unions and interfaces with brief descriptions:

# Chapter 3

# Class Documentation

## 3.1 add.dataflow.sync.Abs Class Reference

Inheritance diagram for add.dataflow.sync.Abs:




### Public Member Functions

- Abs ()
- int compute (int data)

### 3.1.1 Detailed Description

Abs component for the ADD Accelerator Design and Deploy.

The component is responsible for delivering the absolute value of the input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.1.2 Constructor & Destructor Documentation

#### 3.1.2.1 add.dataflow.sync.Abs.Abs ( )

Object Constructor.

### 3.1.3 Member Function Documentation

#### 3.1.3.1 int add.dataflow.sync.Abs.compute ( int *data* )

Method responsible for the component computation.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

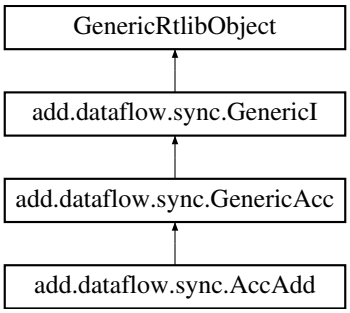
- Returns the result of the computation. In this case, returns the absolute value of the parameter.

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

- add/dataflow/sync/Abs.java

## 3.2 add.dataflow.sync.AccAdd Class Reference

Inheritance diagram for add.dataflow.sync.AccAdd:




### Public Member Functions

- AccAdd ()

### Additional Inherited Members

### 3.2.1 Detailed Description

AccAdd component for the ADD Accelerator Design and Deploy.

The component implements an adder accumulator.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.2.2 Constructor & Destructor Documentation

#### 3.2.2.1 add.dataflow.sync.AccAdd.AccAdd ( )

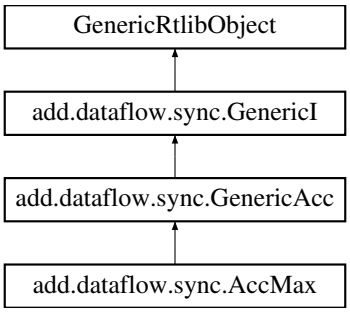
Object Constructor.

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

- add/dataflow/sync/AccAdd.java

## 3.3 add.dataflow.sync.AccMax Class Reference

Inheritance diagram for add.dataflow.sync.AccMax:




### Public Member Functions

- AccMax ()
- void reset ()

### Protected Member Functions

- void accumulate (int data)

### 3.3.1 Detailed Description

AccMax component for the ADD Accelerator Design and Deploy.

The component implements a store for the highest input value.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.3.2 Constructor & Destructor Documentation

#### 3.3.2.1 add.dataflow.sync.AccMax.AccMax ( )

Object Constructor.

### 3.3.3 Member Function Documentation

#### 3.3.3.1 void add.dataflow.sync.AccMax.accumulate ( int *data* ) [protected]

Method that compares the parameter to the stored value. If the parameter is larger, it will override the stored value.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

#### 3.3.3.2 void add.dataflow.sync.AccMax.reset ( )

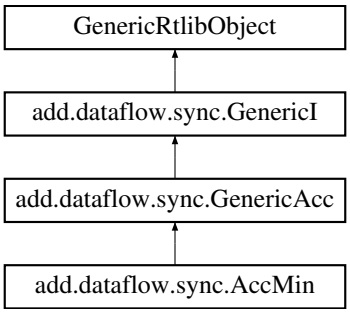
Method responsible for actions required when "Reset" occurs.

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

- add/dataflow/sync/AccMax.java

## 3.4 add.dataflow.sync.AccMin Class Reference

Inheritance diagram for add.dataflow.sync.AccMin:




### Public Member Functions

- AccMin ()
- void reset ()

### Protected Member Functions

- void accumulate (int data)

### 3.4.1 Detailed Description

AccMin component for the ADD Accelerator Design and Deploy.

The component implements a store for the lowest input value.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.4.2 Constructor & Destructor Documentation

#### 3.4.2.1 add.dataflow.sync.AccMin.AccMin ( )

Object Constructor.

### 3.4.3 Member Function Documentation

#### 3.4.3.1 void add.dataflow.sync.AccMin.accumulate ( int *data* ) [protected]

Method that compares the parameter to the stored value. If the parameter is smaller, it will replace the stored value.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

#### 3.4.3.2 void add.dataflow.sync.AccMin.reset ( )

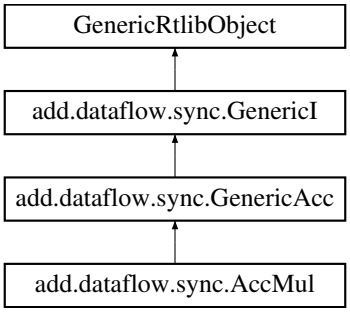
Method responsible for actions required when "Reset" occurs.

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

- add/dataflow/sync/AccMin.java

## 3.5 add.dataflow.sync.AccMul Class Reference

Inheritance diagram for add.dataflow.sync.AccMul:




### Public Member Functions

- AccMul ()
- void reset ()

### Protected Member Functions

- void accumulate (int data)

### 3.5.1 Detailed Description

AccMul component for the ADD Accelerator Design and Deploy.

The component implements a multiplication accumulator.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.5.2 Constructor & Destructor Documentation

#### 3.5.2.1 add.dataflow.sync.AccMul.AccMul ( )

Object Constructor.

### 3.5.3 Member Function Documentation

#### 3.5.3.1 void add.dataflow.sync.AccMul.accumulate ( int *data* ) [protected]

Method that accumulates the input value with the stored. In this case, it multiplies the value stored by the input and stores it.

#### 3.5.3.2 void add.dataflow.sync.AccMul.reset ( )

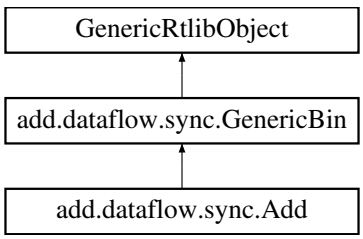
Method responsible for actions required when "Reset" occurs.

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

- add/dataflow/sync/AccMul.java

## 3.6 add.dataflow.sync.Add Class Reference

Inheritance diagram for add.dataflow.sync.Add:




### Public Member Functions

- Add ()
- int compute (int data1, int data2)

### 3.6.1 Detailed Description

Add component for the ADD Accelerator Design and Deploy.

The component is responsible for adding the inputs.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.6.2 Constructor & Destructor Documentation

#### 3.6.2.1 add.dataflow.sync.Add.Add ( )

Object Constructor.

### 3.6.3 Member Function Documentation

#### 3.6.3.1 int add.dataflow.sync.Add.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case performs a addition of the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

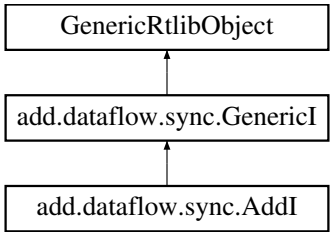
- Returns the result of the computation. In this case the value of the addition of the parameters.

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

- add/dataflow/sync/Add.java

## 3.7 add.dataflow.sync.AddI Class Reference

Inheritance diagram for add.dataflow.sync.AddI:



### Public Member Functions

- AddI ()
- int compute (int data)

### 3.7.1 Detailed Description

AddI component for the ADD Accelerator Design and Deploy.

The component is responsible for adding the input by a (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.7.2 Constructor & Destructor Documentation

#### 3.7.2.1 add.dataflow.sync.AddI.AddI ( )

Object Constructor.

### 3.7.3 Member Function Documentation

#### 3.7.3.1 int add.dataflow.sync.AddI.compute ( int *data* )

Method responsible for the component computation: in this case performs a addition of the parameter by an (immediate) id.

#### Parameters

| | |
|---|---|
| *data* | - Value to be used for computing. |

#### Returns

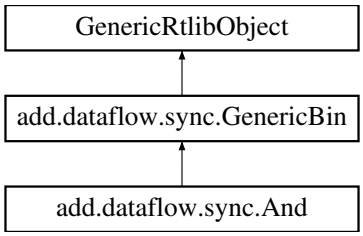
- Returns the result of the computation. In this case the value of the addition of the parameter by the id.

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

- add/dataflow/sync/AddI.java

## 3.8 add.dataflow.sync.And Class Reference

Inheritance diagram for add.dataflow.sync.And:




## Public Member Functions

- And ()
- int compute (int data1, int data2)

### 3.8.1 Detailed Description

And component for the ADD Accelerator Design and Deploy.

The component is responsible for the logical operation "And" between the input

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.8.2 Constructor & Destructor Documentation

#### 3.8.2.1 add.dataflow.sync.And.And ( )

Object Constructor.

### 3.8.3 Member Function Documentation

#### 3.8.3.1 int add.dataflow.sync.And.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case it performs the logical operation "And" between the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

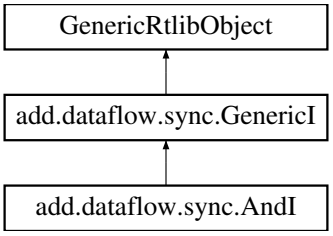
- Returns the result of the computation. In this case the result of the logical operation "And" between the parameters.

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

- add/dataflow/sync/And.java

## 3.9 add.dataflow.sync.AndI Class Reference

Inheritance diagram for add.dataflow.sync.AndI:




### Public Member Functions

- AndI ()
- int compute (int data)

### 3.9.1 Detailed Description

AndI component for the ADD Accelerator Design and Deploy.

The component is responsible for the logical operation "AND" between the input and a id (immediate)

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.9.2 Constructor & Destructor Documentation

#### 3.9.2.1 add.dataflow.sync.AndI.AndI ( )

Object Constructor.

### 3.9.3 Member Function Documentation

#### 3.9.3.1 int add.dataflow.sync.AndI.compute ( int *data* )

Method responsible for the component computation: in this case it performs the logical operation "AND" between the parameter and the (immediate) id.

#### Parameters

| | |
|---|---|
| *data* | - Value to be used for computing. |

#### Returns

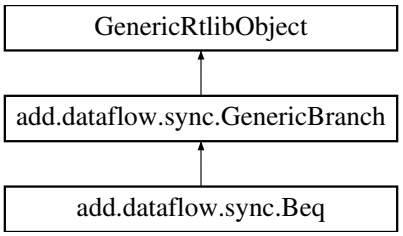
- Returns the result of the computation. In this case the result of the logical operation "AND" between the parameter and the id.

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

- add/dataflow/sync/AndI.java

## 3.10 add.dataflow.sync.Beq Class Reference

Inheritance diagram for add.dataflow.sync.Beq:




### Public Member Functions

- Beq ()
- int compute (int data1, int data2)

### 3.10.1 Detailed Description

Beq component for the ADD Accelerator Design and Deploy.

The component is responsible for comparing equality between the input. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.10.2 Constructor & Destructor Documentation

#### 3.10.2.1 add.dataflow.sync.Beq.Beq ( )

Object Constructor.

### 3.10.3 Member Function Documentation

#### 3.10.3.1 int add.dataflow.sync.Beq.compute ( int *data1*, int *data2* )

Method responsible for component computing: in this case performs a comparison of equality between the input. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

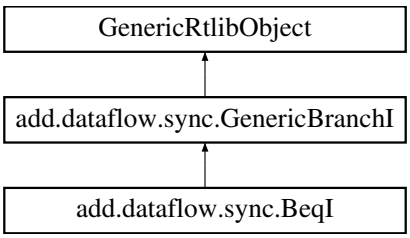
- Returns the result of the computation. In this case "1" if the parameters are equal or "0" if they are different.

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

- add/dataflow/sync/Beq.java

## 3.11 add.dataflow.sync.BeqI Class Reference

Inheritance diagram for add.dataflow.sync.BeqI:




### Public Member Functions

- BeqI ()
- int compute (int data)

### 3.11.1 Detailed Description

BeqI component for the ADD Accelerator Design and Deploy.

The component is responsible for comparing equality between the input and a constant (immediate). Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.11.2 Constructor & Destructor Documentation

#### 3.11.2.1 add.dataflow.sync.Beql.Beql ( )

Object Constructor.

### 3.11.3 Member Function Documentation

#### 3.11.3.1 int add.dataflow.sync.Beql.compute ( int *data* )

Method responsible for component computing: in this case performs a comparison of equality between the input and a constant. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

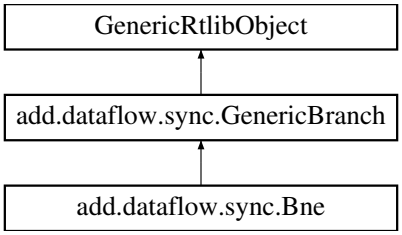
- Returns the result of the computation. In this case "1" if the parameter is equal to the constraint or "0" if they are different.

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

- add/dataflow/sync/Beql.java

## 3.12 add.dataflow.sync.Bne Class Reference

Inheritance diagram for add.dataflow.sync.Bne:




### Public Member Functions

- Bne ()
- int compute (int data1, int data2)

### 3.12.1 Detailed Description

Bne component for the ADD Accelerator Design and Deploy.

The component is responsible for comparing inequality between the input. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.12.2 Constructor & Destructor Documentation

#### 3.12.2.1 add.dataflow.sync.Bne.Bne ( )

Object Constructor.

### 3.12.3 Member Function Documentation

#### 3.12.3.1 int add.dataflow.sync.Bne.compute ( int *data1*, int *data2* )

Method responsible for component computing: in this case performs a comparison of inequality between the input. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

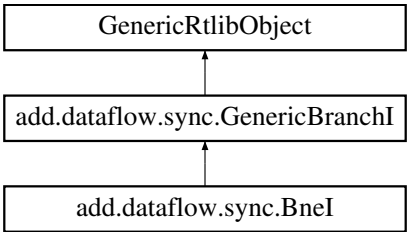
- Returns the result of the computation. In this case "1" if the parameters are different or "0" if they are equal.

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

- add/dataflow/sync/Bne.java

## 3.13 add.dataflow.sync.BneI Class Reference

Inheritance diagram for add.dataflow.sync.BneI:




### Public Member Functions

- BneI ()
- int compute (int data)

### 3.13.1 Detailed Description

BEQI component for the ADD Accelerator Design and Deploy.

The component is responsible for comparing inequality between the input and a constant (immediate). Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.13.2 Constructor & Destructor Documentation

#### 3.13.2.1 add.dataflow.sync.BneI.BneI ( )

Object Constructor.

### 3.13.3 Member Function Documentation

#### 3.13.3.1 int add.dataflow.sync.BneI.compute ( int *data* )

Method responsible for component computing: in this case performs a comparison of inequality between the input and a constant. Depending on the result of the comparison, the "IF" output or the "ELSE" output will receive the value "1" while the other will receive the value "0".

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

- Returns the result of the computation. In this case "0" if the parameter is equal to the constraint or "1" if they are different.

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

- add/dataflow/sync/BneI.java

## 3.14 add.util.ConfReader Class Reference

### Public Member Functions

- int[] ReadConfig (File file)

### 3.14.1 Detailed Description

Class responsible for providing useful routines for the project.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.14.2 Member Function Documentation

#### 3.14.2.1 int [ ] add.util.ConfReader.ReadConfig ( File *file* )

Method responsible for reading a configuration file and returning a vector with the values read.

**Parameters**

| | |
|---|---|
| *file* | - File to read |

**Returns**

- Returns a vector containing the values read in the file.

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

- add/util/ConfReader.java

## 3.15 add.dataflow.DataflowSyncSimulBase Class Reference

### Public Member Functions

- int[] startSimulation (int[] conf, String designPath, int outSize)
- int[] startSimulation (String confPath, String designPath, String desiredReturn, int outSize)
- int[] startFpgaJtag (int[] conf, String quartusStpPath, int outSize)
- int[] startFpgaJtag (String confPath, String quartusStpPath, String desiredReturn, int outSize)
- int[] execHades (int[] rawData, String designPath, int outSize)
- int[] execFpga (int[] rawData, String quartusStpPath, int outSize)

### 3.15.1 Detailed Description

Base class for executing algorithms in the simulator or in the bundle with FPGA.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.15.2 Member Function Documentation

#### 3.15.2.1 int [ ] add.dataflow.DataflowSyncSimulBase.execFpga ( int[ ] *rawData,* String *quartusStpPath,* int *outSize* )

Execution in FPGA Boards

**Parameters**

| | |
|---|---|
| *rawData* | - Vector of data to be processed |
| *quartusStpPath* | - Path to the quartus_stp application. |
| *outSize* | - Output vector size. |

**Returns**

- Returns a vector with the processing results.

#### 3.15.2.2 int [ ] add.dataflow.DataflowSyncSimulBase.execHades ( int[ ] *rawData,* String *designPath,* int *outSize* )

Execution in HADES Simulator

**Parameters**

| | |
|---|---|
| *rawData* | - Vector of data to be processed |
| *designPath* | - Design to be used to run the simulator. |
| *outSize* | - Output vector size. |

**Returns**

- Returns a vector with the processing results.

#### 3.15.2.3 int [ ] add.dataflow.DataflowSyncSimulBase.startFpgaJtag ( int[ ] *conf,* String *quartusStpPath,* int *outSize* )

Method responsible for running the algorithm on the FPGA board.

**Parameters**

| | |
|---|---|
| *conf* | - Configuration vector and data to be executed. |
| *quartusStpPath* | - Path to the quartus_stp application. |
| *outSize* | - Output vector size. |

**Returns**

- Returns a vector with the processing results.

#### 3.15.2.4 int [ ] add.dataflow.DataflowSyncSimulBase.startFpgaJtag ( String *confPath,* String *quartusStpPath,* String *desiredReturn,* int *outSize* )

Method responsible for running the algorithm on the FPGA boardand display the output in the system default output.

**Parameters**

| | |
|---|---|
| *confPath* | - File containing the configuration and data to be processed. |
| *quartusStpPath* | - Path to the quartus_stp application. |
| *desiredReturn* | - Expected outcome. |
| *outSize* | - Output vector size. |

**Returns**

- Returns a vector with the processing results.

#### 3.15.2.5 int [ ] add.dataflow.DataflowSyncSimulBase.startSimulation ( int[ ] *conf,* String *designPath,* int *outSize* )

Method responsible for executing the algorithm in the simulator.

**Parameters**

| | |
|---|---|
| *conf* | - Configuration vector and data to be executed. |
| *designPath* | - Design to be used to run the simulator. |
| *outSize* | - Output vector size. |

**Returns**

- Returns a vector with the processing results.

#### 3.15.2.6 int [ ] add.dataflow.DataflowSyncSimulBase.startSimulation ( String *confPath,* String *designPath,* String *desiredReturn,* int *outSize* )

Method responsible for executing the algorithm in the simulator and display the output in the system default output.

**Parameters**

| | |
|---|---|
| *confPath* | - File containing the configuration and data to be processed. |
| *designPath* | - Design to be used to run the simulator. |
| *desiredReturn* | - Expected outcome. |
| *outSize* | - Output vector size. |

**Returns**

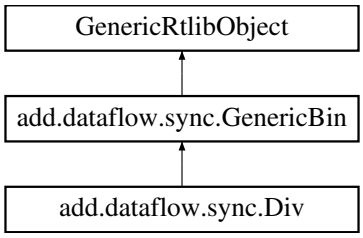
- Returns a vector with the processing results.

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

- add/dataflow/DataflowSyncSimulBase.java

## 3.16 add.dataflow.sync.Div Class Reference

Inheritance diagram for add.dataflow.sync.Div:




### Public Member Functions

- Div ()
- int compute (int data1, int data2)

### 3.16.1 Detailed Description

Div component for the ADD Accelerator Design and Deploy.

The component is responsible for dividing the inputs.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.16.2 Constructor & Destructor Documentation

#### 3.16.2.1 add.dataflow.sync.Div.Div ( )

Object Constructor.

### 3.16.3 Member Function Documentation

#### 3.16.3.1 int add.dataflow.sync.Div.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case performs a division of the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

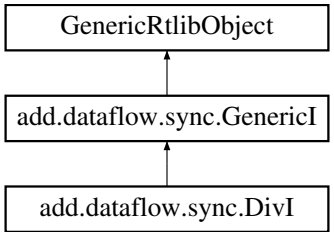
- Returns the result of the computation. In this case the value of the division of the parameters.

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

- add/dataflow/sync/Div.java

## 3.17 add.dataflow.sync.DivI Class Reference

Inheritance diagram for add.dataflow.sync.DivI:




### Public Member Functions

- DivI ()
- int compute (int data)

### 3.17.1 Detailed Description

DivI component for the ADD Accelerator Design and Deploy.

The component is responsible for dividing the input by a (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.17.2 Constructor & Destructor Documentation

#### 3.17.2.1 add.dataflow.sync.DivI.DivI ( )

Object Constructor.

### 3.17.3 Member Function Documentation

#### 3.17.3.1 int add.dataflow.sync.DivI.compute ( int *data* )

Method responsible for the component computation: in this case performs a division of the parameter by an (immediate) id.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

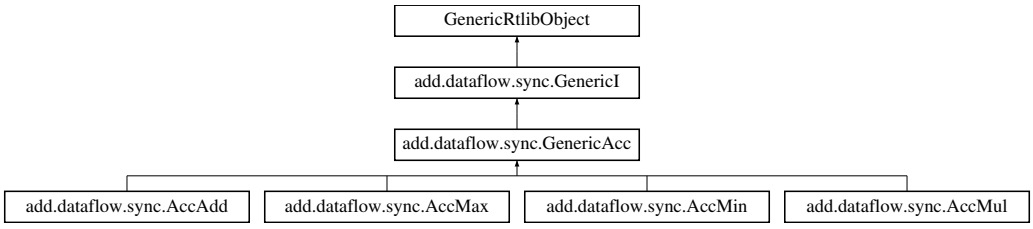
- Returns the result of the computation. In this case the value of the division of the parameter by the id.

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

- add/dataflow/sync/DivI.java

## 3.18 add.dataflow.sync.GenericAcc Class Reference

Inheritance diagram for add.dataflow.sync.GenericAcc:



### Public Member Functions

- GenericAcc ()
- void reset ()
- void evaluate (Object arg)
- int getAcc ()
- void setAcc (int acc)
- int getCounter ()
- void setCounter (int counter)

### Protected Member Functions

- void accumulate (int data)

## 3.18.1 Detailed Description

GenericAcc component for the ADD Accelerator Design and Deploy.

The component implements a generic accumulator.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

## 3.18.2 Constructor & Destructor Documentation

### 3.18.2.1 add.dataflow.sync.GenericAcc.GenericAcc ( )

Object Constructor.

## 3.18.3 Member Function Documentation

### 3.18.3.1 void add.dataflow.sync.GenericAcc.accumulate ( int *data* ) [protected]

Method responsible for performing the accumulation or not.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for the computation. |

### 3.18.3.2 void add.dataflow.sync.GenericAcc.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN input is high level. It will execute the reset(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the ACC value when the computation finishes.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.18.3.3 int add.dataflow.sync.GenericAcc.getAcc ( )

**Returns**

the acc

#### 3.18.3.4 int add.dataflow.sync.GenericAcc.getCounter ( )

**Returns**

the counter

#### 3.18.3.5 void add.dataflow.sync.GenericAcc.reset ( )

Method executed when the signal from the reset input goes to high logic level.In this case it clears the text displayed by the component and de accumulator.

#### 3.18.3.6 void add.dataflow.sync.GenericAcc.setAcc ( int *acc* )

**Parameters**

| | |
|---|---|
| *acc* | the acc to set |

#### 3.18.3.7 void add.dataflow.sync.GenericAcc.setCounter ( int *counter* )

**Parameters**

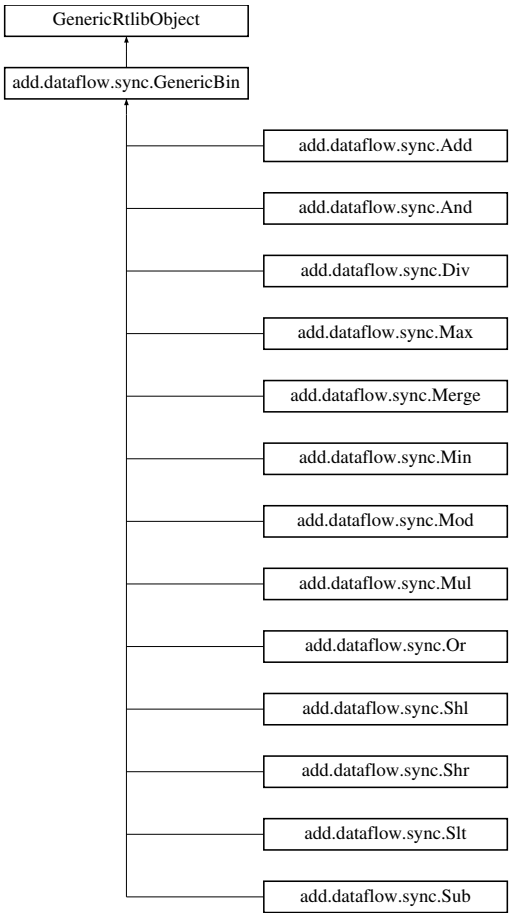
| | |
|---|---|
| *counter* | the counter to set |

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

- add/dataflow/sync/GenericAcc.java

## 3.19 add.dataflow.sync.GenericBin Class Reference

Inheritance diagram for add.dataflow.sync.GenericBin:




## Public Member Functions

- GenericBin ()
- void constructPorts ()
- void setString (String s)
- void setSymbol (Symbol s)
- int compute (int data1, int data2)
- void notCompute ()
- void reseted ()
- void tickUp ()
- void tickDown ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- Label getStringLabel ()
- void setStringLabel (Label stringLabel)
- Label getLabelNome ()
- void setLabelNome (Label labelNome)
- String getComponentType ()
- void setComponentType (String componentType)
- String getS ()
- void setS (String s)

- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRin1 ()
- void setPortRin1 (PortStdLogic1164 portRin1)
- PortStdLogic1164 getPortRin2 ()
- void setPortRin2 (PortStdLogic1164 portRin2)
- PortStdLogic1164 getPortRout ()
- void setPortRout (PortStdLogic1164 portRout)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector getPortDin1 ()
- void setPortDin1 (PortStdLogicVector portDin1)
- PortStdLogicVector getPortDin2 ()
- void setPortDin2 (PortStdLogicVector portDin2)
- PortStdLogicVector getPortDout ()
- void setPortDout (PortStdLogicVector portDout)

### 3.19.1 Detailed Description

GenericBin component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components with two inputs.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.19.2 Constructor & Destructor Documentation

#### 3.19.2.1 add.dataflow.sync.GenericBin.GenericBin ( )

Object Constructor.

### 3.19.3 Member Function Documentation

#### 3.19.3.1 int add.dataflow.sync.GenericBin.compute ( int *data1*, int *data2* )

Method responsible for the computation of the output.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 1. |

**Returns**

- Return of computation

#### 3.19.3.2 void add.dataflow.sync.GenericBin.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.19.3.3 void add.dataflow.sync.GenericBin.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.19.3.4 void add.dataflow.sync.GenericBin.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN (1 and 2) inputs are high level. It will execute the reseted(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.19.3.5 String add.dataflow.sync.GenericBin.getComponentType ( )

**Returns**

the componentType

#### 3.19.3.6 Label add.dataflow.sync.GenericBin.getLabelNome ( )

**Returns**

the labelNome

#### 3.19.3.7 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortClk ( )

**Returns**

the portClk

#### 3.19.3.8 PortStdLogicVector add.dataflow.sync.GenericBin.getPortDin1 ( )

**Returns**

the portDin1

#### 3.19.3.9 PortStdLogicVector add.dataflow.sync.GenericBin.getPortDin2 ( )

**Returns**

the portDin2

#### 3.19.3.10 PortStdLogicVector add.dataflow.sync.GenericBin.getPortDout ( )

**Returns**

the portDout

#### 3.19.3.11 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortEn ( )

**Returns**

the portEn

#### 3.19.3.12 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortRin1 ( )

**Returns**

the portRin1

#### 3.19.3.13 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortRin2 ( )

**Returns**

the portRin2

#### 3.19.3.14 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortRout ( )

**Returns**

the portRout

#### 3.19.3.15 PortStdLogic1164 add.dataflow.sync.GenericBin.getPortRst ( )

**Returns**

the portRst

#### 3.19.3.16 String add.dataflow.sync.GenericBin.getS ( )

**Returns**

the s

#### 3.19.3.17 Label add.dataflow.sync.GenericBin.getStringLabel ( )

**Returns**

the stringLabel

#### 3.19.3.18 boolean add.dataflow.sync.GenericBin.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.19.3.19 boolean add.dataflow.sync.GenericBin.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- - TRUE means that the symbol will be built dynamically.

#### 3.19.3.20 void add.dataflow.sync.GenericBin.notCompute ( )

Method executed when computing is not performed. In this case it clears the text displayed by the component.

#### 3.19.3.21 void add.dataflow.sync.GenericBin.reseted ( )

Method executed when the signal from the reset input goes to high logic level.In this case it clears the text displayed by the component.

#### 3.19.3.22 void add.dataflow.sync.GenericBin.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

#### 3.19.3.23 void add.dataflow.sync.GenericBin.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.19.3.24 void add.dataflow.sync.GenericBin.setLabelNome ( Label *labelNome* )

**Parameters**

| | |
|---|---|
| *labelNome* | the labelNome to set |

#### 3.19.3.25 void add.dataflow.sync.GenericBin.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.19.3.26 void add.dataflow.sync.GenericBin.setPortDin1 ( PortStdLogicVector *portDin1* )

**Parameters**

| | |
|---|---|
| *portDin1* | the portDin1 to set |

#### 3.19.3.27 void add.dataflow.sync.GenericBin.setPortDin2 ( PortStdLogicVector *portDin2* )

**Parameters**

| | |
|---|---|
| *portDin2* | the portDin2 to set |

#### 3.19.3.28 void add.dataflow.sync.GenericBin.setPortDout ( PortStdLogicVector *portDout* )

**Parameters**

| | |
|---|---|
| *portDout* | the portDout to set |

#### 3.19.3.29 void add.dataflow.sync.GenericBin.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.19.3.30 void add.dataflow.sync.GenericBin.setPortRin1 ( PortStdLogic1164 *portRin1* )

**Parameters**

| | |
|---|---|
| *portRin1* | the portRin1 to set |

#### 3.19.3.31 void add.dataflow.sync.GenericBin.setPortRin2 ( PortStdLogic1164 *portRin2* )

**Parameters**

| | |
|---|---|
| *portRin2* | the portRin2 to set |

#### 3.19.3.32 void add.dataflow.sync.GenericBin.setPortRout ( PortStdLogic1164 *portRout* )

**Parameters**

| | |
|---|---|
| *portRout* | the portRout to set |

#### 3.19.3.33 void add.dataflow.sync.GenericBin.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.19.3.34 void add.dataflow.sync.GenericBin.setS ( String *s* )

**Parameters**

| | |
|---|---|
| *s* | the s to set |

#### 3.19.3.35 void add.dataflow.sync.GenericBin.setString ( String *s* )

Method responsible for updating the text displayed by the component.

**Parameters**

| | |
|---|---|
| *s* | - Text to be updated. |

#### 3.19.3.36 void add.dataflow.sync.GenericBin.setStringLabel ( Label *stringLabel* )

**Parameters**

| | |
|---|---|
| *stringLabel* | the stringLabel to set |

#### 3.19.3.37 void add.dataflow.sync.GenericBin.setSymbol ( Symbol *s* )

Method responsible for updating the component symbol.

**Parameters**

| | |
|---|---|
| *s* | |

#### 3.19.3.38 void add.dataflow.sync.GenericBin.tickDown ( )

Method executed when the clock signal goes to low logic level.

#### 3.19.3.39 void add.dataflow.sync.GenericBin.tickUp ( )

Method executed when the clock signal goes to high logic level.

#### 3.19.3.40 void add.dataflow.sync.GenericBin.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

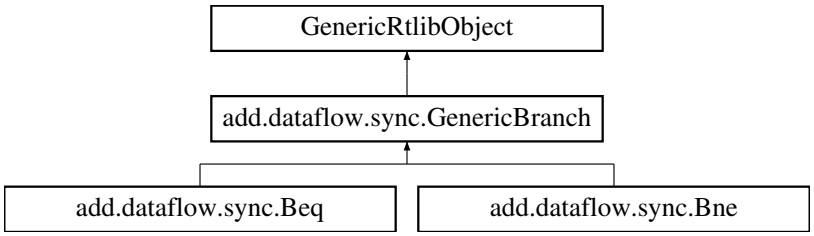
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericBin.java

## 3.20 add.dataflow.sync.GenericBranch Class Reference

Inheritance diagram for add.dataflow.sync.GenericBranch:




### Public Member Functions

- GenericBranch ()
- void constructPorts ()
- void setString (String s)
- void setSymbol (Symbol s)
- int compute (int data1, int data2)
- void reseted ()
- void tickUp ()
- void tickDown ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- Label getStringLabel ()
- void setStringLabel (Label stringLabel)
- Label getLabel_nome ()
- void setLabel_nome (Label label_nome)
- String getComponentType ()
- void setComponentType (String componentType)
- String getS ()
- void setS (String s)
- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRin1 ()
- void setPortRin1 (PortStdLogic1164 portRin1)
- PortStdLogic1164 getPortRin2 ()
- void setPortRin2 (PortStdLogic1164 portRin2)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector getPortDin1 ()
- void setPortDin1 (PortStdLogicVector portDin1)
- PortStdLogicVector getPortDin2 ()
- void setPortDin2 (PortStdLogicVector portDin2)
- PortStdLogic1164 getPortIf ()
- void setPortIf (PortStdLogic1164 portIf)
- PortStdLogic1164 getPortElse ()
- void setPortElse (PortStdLogic1164 portElse)

### 3.20.1 Detailed Description

GenericBranch component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components with an input and that make a comparison between the inputs.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.20.2 Constructor & Destructor Documentation

#### 3.20.2.1 add.dataflow.sync.GenericBranch.GenericBranch ( )

Object Constructor.

### 3.20.3 Member Function Documentation

#### 3.20.3.1 int add.dataflow.sync.GenericBranch.compute ( int *data1*, int *data2* )

Method responsible for the computation of the output.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 1. |

**Returns**

- Return of computation

#### 3.20.3.2 void add.dataflow.sync.GenericBranch.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.20.3.3 void add.dataflow.sync.GenericBranch.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.20.3.4 void add.dataflow.sync.GenericBranch.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events.In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN (1 and 2) inputs are high level. It will execute the reseted(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.20.3.5 String add.dataflow.sync.GenericBranch.getComponentType ( )

**Returns**

the componentType

#### 3.20.3.6 Label add.dataflow.sync.GenericBranch.getLabel_nome ( )

**Returns**

the label_nome

#### 3.20.3.7 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortClk ( )

**Returns**

the portClk

#### 3.20.3.8 PortStdLogicVector add.dataflow.sync.GenericBranch.getPortDin1 ( )

**Returns**

the portDin1

#### 3.20.3.9 PortStdLogicVector add.dataflow.sync.GenericBranch.getPortDin2 ( )

**Returns**

the portDin2

#### 3.20.3.10 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortElse ( )

**Returns**

the portElse

#### 3.20.3.11 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortEn ( )

**Returns**

the portEn

#### 3.20.3.12 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortIf ( )

**Returns**

the portIf

#### 3.20.3.13 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortRin1 ( )

**Returns**

the portRin1

#### 3.20.3.14 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortRin2 ( )

**Returns**

the portRin2

#### 3.20.3.15 PortStdLogic1164 add.dataflow.sync.GenericBranch.getPortRst ( )

**Returns**

the portRst

#### 3.20.3.16 String add.dataflow.sync.GenericBranch.getS ( )

**Returns**

the s

#### 3.20.3.17 Label add.dataflow.sync.GenericBranch.getStringLabel ( )

**Returns**

the stringLabel

#### 3.20.3.18 boolean add.dataflow.sync.GenericBranch.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.20.3.19 boolean add.dataflow.sync.GenericBranch.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be built dynamically.

#### 3.20.3.20 void add.dataflow.sync.GenericBranch.reseted ( )

Method executed when the signal from the reset input goes to high logic level.In this case it clears the text displayed by the component.

#### 3.20.3.21 void add.dataflow.sync.GenericBranch.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set to the component name. |

#### 3.20.3.22 void add.dataflow.sync.GenericBranch.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.20.3.23 void add.dataflow.sync.GenericBranch.setLabel_nome ( Label *label_nome* )

**Parameters**

| | |
|---|---|
| *label_nome* | the label_nome to set |

#### 3.20.3.24 void add.dataflow.sync.GenericBranch.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.20.3.25 void add.dataflow.sync.GenericBranch.setPortDin1 ( PortStdLogicVector *portDin1* )

**Parameters**

| | |
|---|---|
| *portDin1* | the portDin1 to set |

#### 3.20.3.26 void add.dataflow.sync.GenericBranch.setPortDin2 ( PortStdLogicVector *portDin2* )

**Parameters**

| | |
|---|---|
| *portDin2* | the portDin2 to set |

#### 3.20.3.27 void add.dataflow.sync.GenericBranch.setPortElse ( PortStdLogic1164 *portElse* )

**Parameters**

| | |
|---|---|
| *portElse* | the portElse to set |

#### 3.20.3.28 void add.dataflow.sync.GenericBranch.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.20.3.29 void add.dataflow.sync.GenericBranch.setPortIf ( PortStdLogic1164 *portIf* )

**Parameters**

| | |
|---|---|
| *portIf* | the portIf to set |

#### 3.20.3.30 void add.dataflow.sync.GenericBranch.setPortRin1 ( PortStdLogic1164 *portRin1* )

**Parameters**

| | |
|---|---|
| *portRin1* | the portRin1 to set |

#### 3.20.3.31 void add.dataflow.sync.GenericBranch.setPortRin2 ( PortStdLogic1164 *portRin2* )

**Parameters**

| | |
|---|---|
| *portRin2* | the portRin2 to set |

#### 3.20.3.32 void add.dataflow.sync.GenericBranch.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.20.3.33 void add.dataflow.sync.GenericBranch.setS ( String *s* )

**Parameters**

| | |
|---|---|
| *s* | the s to set |

#### 3.20.3.34 void add.dataflow.sync.GenericBranch.setString ( String *s* )

Method responsible for updating the text displayed by the component.

**Parameters**

| | |
|---|---|
| *s* | - Text to be updated. |

#### 3.20.3.35 void add.dataflow.sync.GenericBranch.setStringLabel ( Label *stringLabel* )

**Parameters**

| | |
|---|---|
| *stringLabel* | the stringLabel to set |

#### 3.20.3.36 void add.dataflow.sync.GenericBranch.setSymbol ( Symbol *s* )

Method responsible for updating the component symbol.

**Parameters**

| | |
|---|---|
| *s* | - Symbol passed automatically. |

#### 3.20.3.37 void add.dataflow.sync.GenericBranch.tickDown ( )

Method executed when the clock signal goes to low logic level.

#### 3.20.3.38 void add.dataflow.sync.GenericBranch.tickUp ( )

Method executed when the clock signal goes to high logic level.

#### 3.20.3.39 void add.dataflow.sync.GenericBranch.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

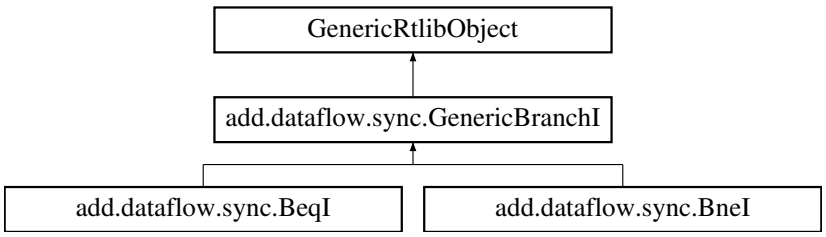
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericBranch.java

## 3.21 add.dataflow.sync.GenericBranchI Class Reference

Inheritance diagram for add.dataflow.sync.GenericBranchI:




### Public Member Functions

- GenericBranchI ()
- void constructPorts ()
- void setString (String componentId, String componentImmediate)
- void setSymbol (Symbol s)
- int compute (int data)
- void reseted ()

- void tickUp ()
- void tickDown ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- void mousePressed (java.awt.event.MouseEvent me)
- Label getStringLabelId ()
- void setStringLabelId (Label stringLabelId)
- Label getStringLabelImmediate ()
- void setStringLabelImmediate (Label stringLabelImmediate)
- Label getLabelNome ()
- void setLabelNome (Label labelNome)
- String getComponentId ()
- void setComponentId (String componentId)
- String getComponentImmediate ()
- void setComponentImmediate (String componentImmediate)
- String getComponentType ()
- void setComponentType (String componentType)
- Rectangle getBackground ()
- void setBackground (Rectangle background)
- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRin ()
- void setPortRin (PortStdLogic1164 portRin)
- PortStdLogic1164 getPortIf ()
- void setPortIf (PortStdLogic1164 portIf)
- PortStdLogic1164 getPortElse ()
- void setPortElse (PortStdLogic1164 portElse)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector getPortDin ()
- void setPortDin (PortStdLogicVector portDin)
- PortStdLogicVector getPortDconf ()
- void setPortDconf (PortStdLogicVector portDconf)
- int getId ()
- void setId (int id)
- int getImmediate ()
- void setImmediate (int immediate)

### 3.21.1 Detailed Description

GenericBranchI component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components with an input and that make a comparison with a (immediate) constant.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.21.2 Constructor & Destructor Documentation

#### 3.21.2.1 add.dataflow.sync.GenericBranchI.GenericBranchI ( )

Object Constructor.

### 3.21.3 Member Function Documentation

#### 3.21.3.1 int add.dataflow.sync.GenericBranchI.compute ( int *data* )

Method responsible for the computation of the output.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for the computation. |

**Returns**

- Return of computation

#### 3.21.3.2 void add.dataflow.sync.GenericBranchI.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.21.3.3 void add.dataflow.sync.GenericBranchI.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.21.3.4 void add.dataflow.sync.GenericBranchI.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN input is high level. It will execute the reseted(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.21.3.5 Rectangle add.dataflow.sync.GenericBranchI.getBackground ( )

**Returns**

the background

#### 3.21.3.6 String add.dataflow.sync.GenericBranchI.getComponentId ( )

**Returns**

the componentId

#### 3.21.3.7 String add.dataflow.sync.GenericBranchI.getComponentImmediate ( )

**Returns**

the componentImmediate

#### 3.21.3.8 String add.dataflow.sync.GenericBranchI.getComponentType ( )

**Returns**

the componentType

#### 3.21.3.9 int add.dataflow.sync.GenericBranchI.getId ( )

**Returns**

the id

#### 3.21.3.10 int add.dataflow.sync.GenericBranchI.getImmediate ( )

**Returns**

the immediate

#### 3.21.3.11 Label add.dataflow.sync.GenericBranchI.getLabelNome ( )

**Returns**

the labelNome

#### 3.21.3.12 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortClk ( )

**Returns**

the portClk

#### 3.21.3.13 PortStdLogicVector add.dataflow.sync.GenericBranchI.getPortDconf ( )

**Returns**

the portDconf

#### 3.21.3.14 PortStdLogicVector add.dataflow.sync.GenericBranchI.getPortDin ( )

**Returns**

the portDin

#### 3.21.3.15 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortElse ( )

**Returns**

the portElse

#### 3.21.3.16 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortEn ( )

**Returns**

the portEn

#### 3.21.3.17 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortIf ( )

**Returns**

the portIf

#### 3.21.3.18 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortRin ( )

**Returns**

the portRin

#### 3.21.3.19 PortStdLogic1164 add.dataflow.sync.GenericBranchI.getPortRst ( )

**Returns**

the portRst

#### 3.21.3.20 Label add.dataflow.sync.GenericBranchI.getStringLabelId ( )

**Returns**

the stringLabelId

#### 3.21.3.21 Label add.dataflow.sync.GenericBranchI.getStringLabelImmediate ( )

**Returns**

the stringLabelImmediate

#### 3.21.3.22 boolean add.dataflow.sync.GenericBranchI.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.21.3.23 void add.dataflow.sync.GenericBranchI.mousePressed ( java.awt.event.MouseEvent *me* )

Method responsible for changing the value of the constant for more or less, depending on whether the mouse click is done by the right or left button respectively.

**Parameters**

| | |
|---|---|
| *me* | - Object where the event occurred. |

#### 3.21.3.24 boolean add.dataflow.sync.GenericBranchI.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component'componentId symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be built dynamically.

#### 3.21.3.25 void add.dataflow.sync.GenericBranchI.reseted ( )

Method executed when the signal from the reset input goes to high logic level.In this case it clears the text displayed by the component.

#### 3.21.3.26 void add.dataflow.sync.GenericBranchI.setBackground ( Rectangle *background* )

**Parameters**

| | |
|---|---|
| *background* | the background to set |

#### 3.21.3.27 void add.dataflow.sync.GenericBranchI.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set to the component name. |

#### 3.21.3.28 void add.dataflow.sync.GenericBranchI.setComponentId ( String *componentId* )

**Parameters**

| | |
|---|---|
| *componentId* | the componentId to set |

#### 3.21.3.29 void add.dataflow.sync.GenericBranchI.setComponentImmediate ( String *componentImmediate* )

**Parameters**

| | |
|---|---|
| *component-Immediate* | the componentImmediate to set |

#### 3.21.3.30 void add.dataflow.sync.GenericBranchI.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.21.3.31 void add.dataflow.sync.GenericBranchI.setId ( int *id* )

**Parameters**

| | |
|---|---|
| *id* | the id to set |

#### 3.21.3.32 void add.dataflow.sync.GenericBranchI.setImmediate ( int *immediate* )

**Parameters**

| | |
|---|---|
| *immediate* | the immediate to set |

#### 3.21.3.33 void add.dataflow.sync.GenericBranchI.setLabelNome ( Label *labelNome* )

**Parameters**

| | |
|---|---|
| *labelNome* | the labelNome to set |

#### 3.21.3.34 void add.dataflow.sync.GenericBranchI.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.21.3.35 void add.dataflow.sync.GenericBranchI.setPortDconf ( PortStdLogicVector *portDconf* )

**Parameters**

| | |
|---|---|
| *portDconf* | the portDconf to set |

#### 3.21.3.36 void add.dataflow.sync.GenericBranchI.setPortDin ( PortStdLogicVector *portDin* )

**Parameters**

| | |
|---|---|
| *portDin* | the portDin to set |

#### 3.21.3.37 void add.dataflow.sync.GenericBranchI.setPortElse ( PortStdLogic1164 *portElse* )

**Parameters**

| | |
|---|---|
| *portElse* | the portElse to set |

#### 3.21.3.38 void add.dataflow.sync.GenericBranchI.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.21.3.39 void add.dataflow.sync.GenericBranchI.setPortIf ( PortStdLogic1164 *portIf* )

**Parameters**

| | |
|---|---|
| *portIf* | the portIf to set |

#### 3.21.3.40 void add.dataflow.sync.GenericBranchI.setPortRin ( PortStdLogic1164 *portRin* )

**Parameters**

| | |
|---|---|
| *portRin* | the portRin to set |

#### 3.21.3.41 void add.dataflow.sync.GenericBranchI.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.21.3.42 void add.dataflow.sync.GenericBranchI.setString ( String *componentId,* String *componentImmediate* )

Method responsible for updating the text displayed by the component.

**Parameters**

| | |
|---|---|
| *componentId* | - Text to be updated. |

#### 3.21.3.43 void add.dataflow.sync.GenericBranchI.setStringLabelId ( Label *stringLabelId* )

**Parameters**

| | |
|---|---|
| *stringLabelId* | the stringLabelId to set |

#### 3.21.3.44 void add.dataflow.sync.GenericBranchI.setStringLabelImmediate ( Label *stringLabelImmediate* )

**Parameters**

| | |
|---|---|
| *stringLabel-Immediate* | the stringLabelImmediate to set |

#### 3.21.3.45 void add.dataflow.sync.GenericBranchI.setSymbol ( Symbol *s* )

Method responsible for updating the component symbol.

**Parameters**

| | |
|---|---|
| *s* | - Symbol passed automatically. |

#### 3.21.3.46 void add.dataflow.sync.GenericBranchI.tickDown ( )

Method executed when the clock signal goes to low logic level.

#### 3.21.3.47 void add.dataflow.sync.GenericBranchI.tickUp ( )

Method executed when the clock signal goes to high logic level.

#### 3.21.3.48 void add.dataflow.sync.GenericBranchI.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

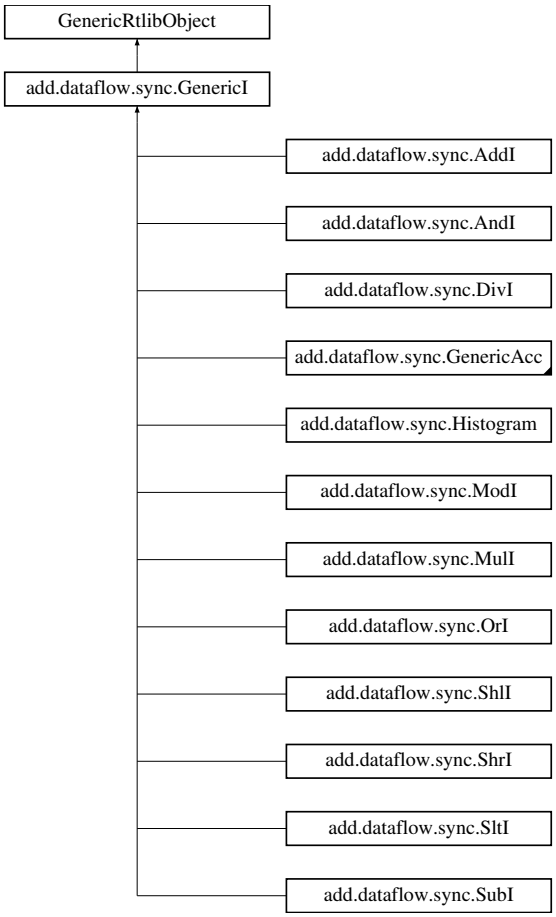
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericBranchI.java

## 3.22 add.dataflow.sync.GenericI Class Reference

Inheritance diagram for add.dataflow.sync.GenericI:




## Public Member Functions

- GenericI ()
- void constructPorts ()
- void setString (String componentId, String componentImmediate)
- void setSymbol (Symbol s)
- int compute (int data)
- void notCompute ()
- void reset ()
- void tickUp ()
- void tickDown ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- void mousePressed (java.awt.event.MouseEvent me)
- int getId ()
- void setId (int id)
- int getImmediate ()
- void setImmediate (int immediate)
- Label getStringLabelId ()
- void setStringLabelId (Label stringLabelId)
- Label getStringLabelImmediate ()

- void setStringLabelImmediate (Label stringLabelImmediate)
- Label getLabelNome ()
- void setLabelNome (Label labelNome)
- String getComponentId ()
- void setComponentId (String componentId)
- String getComponentImmediate ()
- void setComponentImmediate (String componentImmediate)
- String getComponentType ()
- void setComponentType (String componentType)
- Rectangle getBackground ()
- void setBackground (Rectangle background)
- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRin ()
- void setPortRin (PortStdLogic1164 portRin)
- PortStdLogic1164 getPortRout ()
- void setPortRout (PortStdLogic1164 portRout)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector getPortDin ()
- void setPortDin (PortStdLogicVector portDin)
- PortStdLogicVector getPortDout ()
- void setPortDout (PortStdLogicVector portDout)
- PortStdLogicVector getPortDconf ()
- void setPortDconf (PortStdLogicVector portDconf)

### 3.22.1 Detailed Description

GenericI component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components with an input and that perform the computation with an (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.22.2 Constructor & Destructor Documentation

#### 3.22.2.1 add.dataflow.sync.GenericI.GenericI ( )

Object Constructor.

### 3.22.3 Member Function Documentation

#### 3.22.3.1 int add.dataflow.sync.GenericI.compute ( int *data* )

Method responsible for the computation of the output and set the new text to be shown by the component. In this case the id.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for the computation. |

**Returns**

- Return of computation

#### 3.22.3.2 void add.dataflow.sync.GenericI.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.22.3.3 void add.dataflow.sync.GenericI.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.22.3.4 void add.dataflow.sync.GenericI.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN input is high level. It will execute the reset(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.22.3.5 Rectangle add.dataflow.sync.GenericI.getBackground ( )

**Returns**

the background

#### 3.22.3.6 String add.dataflow.sync.GenericI.getComponentId ( )

**Returns**

the componentId

#### 3.22.3.7 String add.dataflow.sync.GenericI.getComponentImmediate ( )

**Returns**

the componentImmediate

#### 3.22.3.8 String add.dataflow.sync.GenericI.getComponentType ( )

**Returns**

the componentType

#### 3.22.3.9 int add.dataflow.sync.GenericI.getId ( )

**Returns**

the id

#### 3.22.3.10 int add.dataflow.sync.GenericI.getImmediate ( )

**Returns**

the immediate

#### 3.22.3.11 Label add.dataflow.sync.GenericI.getLabelNome ( )

**Returns**

the labelNome

#### 3.22.3.12 PortStdLogic1164 add.dataflow.sync.GenericI.getPortClk ( )

**Returns**

the portClk

#### 3.22.3.13 PortStdLogicVector add.dataflow.sync.GenericI.getPortDconf ( )

**Returns**

the portDconf

#### 3.22.3.14 PortStdLogicVector add.dataflow.sync.GenericI.getPortDin ( )

**Returns**

the portDin

#### 3.22.3.15 PortStdLogicVector add.dataflow.sync.GenericI.getPortDout ( )

**Returns**

the portDout

#### 3.22.3.16 PortStdLogic1164 add.dataflow.sync.GenericI.getPortEn ( )

**Returns**

the portEn

#### 3.22.3.17 PortStdLogic1164 add.dataflow.sync.GenericI.getPortRin ( )

**Returns**

the portRin

#### 3.22.3.18 PortStdLogic1164 add.dataflow.sync.GenericI.getPortRout ( )

**Returns**

the portRout

#### 3.22.3.19 PortStdLogic1164 add.dataflow.sync.GenericI.getPortRst ( )

**Returns**

the portRst

#### 3.22.3.20 Label add.dataflow.sync.GenericI.getStringLabelId ( )

**Returns**

the stringLabelId

#### 3.22.3.21 Label add.dataflow.sync.GenericI.getStringLabelImmediate ( )

**Returns**

the stringLabelImmediate

#### 3.22.3.22 boolean add.dataflow.sync.GenericI.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.22.3.23 void add.dataflow.sync.GenericI.mousePressed ( java.awt.event.MouseEvent *me* )

Method responsible for changing the value of the id for more or less, depending on whether the mouse click is done by the right or left button respectively.

**Parameters**

| | |
|---|---|
| *me* | - Object where the event occurred. |

#### 3.22.3.24 boolean add.dataflow.sync.GenericI.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be made dynamically.

#### 3.22.3.25 void add.dataflow.sync.GenericI.notCompute ( )

Method executed when computing is not performed.

#### 3.22.3.26 void add.dataflow.sync.GenericI.reset ( )

Method executed when the signal from the reset input goes to high logic level. It sets the new text to be shown by the component. In this case the id.

#### 3.22.3.27 void add.dataflow.sync.GenericI.setBackground ( Rectangle *background* )

**Parameters**

| | |
|---|---|
| *background* | the background to set |

#### 3.22.3.28 void add.dataflow.sync.GenericI.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set in component name. |

#### 3.22.3.29 void add.dataflow.sync.GenericI.setComponentId ( String *componentId* )

**Parameters**

| | |
|---|---|
| *componentId* | the componentId to set |

#### 3.22.3.30 void add.dataflow.sync.GenericI.setComponentImmediate ( String *componentImmediate* )

**Parameters**

| | |
|---|---|
| *component-Immediate* | the componentImmediate to set |

#### 3.22.3.31 void add.dataflow.sync.GenericI.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.22.3.32 void add.dataflow.sync.GenericI.setId ( int *id* )

**Parameters**

| | |
|---|---|
| *id* | the id to set |

#### 3.22.3.33 void add.dataflow.sync.GenericI.setImmediate ( int *immediate* )

**Parameters**

| | |
|---|---|
| *immediate* | the immediate to set |

#### 3.22.3.34 void add.dataflow.sync.GenericI.setLabelNome ( Label *labelNome* )

**Parameters**

| | |
|---|---|
| *labelNome* | the labelNome to set |

#### 3.22.3.35 void add.dataflow.sync.GenericI.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.22.3.36 void add.dataflow.sync.GenericI.setPortDconf ( PortStdLogicVector *portDconf* )

**Parameters**

| | |
|---|---|
| *portDconf* | the portDconf to set |

#### 3.22.3.37 void add.dataflow.sync.GenericI.setPortDin ( PortStdLogicVector *portDin* )

**Parameters**

| | |
|---|---|
| *portDin* | the portDin to set |

#### 3.22.3.38 void add.dataflow.sync.GenericI.setPortDout ( PortStdLogicVector *portDout* )

**Parameters**

| | |
|---|---|
| *portDout* | the portDout to set |

#### 3.22.3.39 void add.dataflow.sync.GenericI.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.22.3.40 void add.dataflow.sync.GenericI.setPortRin ( PortStdLogic1164 *portRin* )

**Parameters**

| | |
|---|---|
| *portRin* | the portRin to set |

#### 3.22.3.41 void add.dataflow.sync.GenericI.setPortRout ( PortStdLogic1164 *portRout* )

**Parameters**

| | |
|---|---|
| *portRout* | the portRout to set |

#### 3.22.3.42 void add.dataflow.sync.GenericI.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.22.3.43 void add.dataflow.sync.GenericI.setString ( String *componentId,* String *componentImmediate* )

Method responsible for updating the text displayed by the component.

**Parameters**

| | |
|---|---|
| *componentId* | - Text to be updated. |
| *component-Immediate* | - Text to be updated. |

#### 3.22.3.44 void add.dataflow.sync.GenericI.setStringLabelId ( Label *stringLabelId* )

**Parameters**

| | |
|---|---|
| *stringLabelId* | the stringLabelId to set |

#### 3.22.3.45 void add.dataflow.sync.GenericI.setStringLabelImmediate ( Label *stringLabelImmediate* )

**Parameters**

| | |
|---|---|
| *stringLabel-Immediate* | the stringLabelImmediate to set |

#### 3.22.3.46 void add.dataflow.sync.GenericI.setSymbol ( Symbol *s* )

Method responsible for updating the component symbol.

**Parameters**

| | |
|---|---|
| *s* | - Symbol passed automatically. |

#### 3.22.3.47 void add.dataflow.sync.GenericI.tickDown ( )

Method executed when the clock signal goes to low logic level.

#### 3.22.3.48 void add.dataflow.sync.GenericI.tickUp ( )

Method executed when the clock signal goes to high logic level.

#### 3.22.3.49 void add.dataflow.sync.GenericI.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

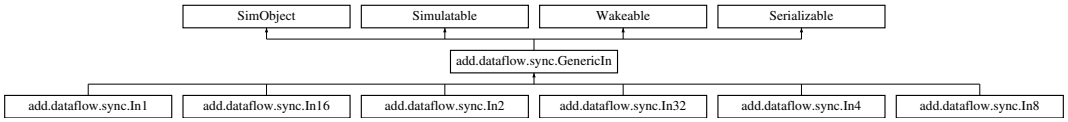
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericI.java

## 3.23 add.dataflow.sync.GenericIn Class Reference

Inheritance diagram for add.dataflow.sync.GenericIn:




### Public Member Functions

- GenericIn ()
- GenericIn (int QTDE_PORTS)
- void constructPorts ()
- void setVectorIn (int[] vectorIn)
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- double getDelay ()
- void setDelay (double _delay)
- void setDelay (String s)
- void wakeup (Object arg)
- void updateSymbol ()
- int getN_bits ()
- void setN_bits (int n_bits)
- StdLogicVector getVector ()
- void setVector (StdLogicVector vector)
- StdLogicVector getVector_UUU ()
- void setVector_UUU (StdLogicVector vector_UUU)
- StdLogicVector getVector_XXX ()
- void setVector_XXX (StdLogicVector vector_XXX)
- StdLogicVector getVector_ZZZ ()
- void setVector_ZZZ (StdLogicVector vector_ZZZ)
- StdLogicVector getVector_000 ()
- void setVector_000 (StdLogicVector vector_000)
- StdLogicVector getVector_111 ()
- void setVector_111 (StdLogicVector vector_111)
- PortStdLogicVector getVectorOutputPort ()
- void setVectorOutputPort (PortStdLogicVector vectorOutputPort)
- double getDefaultdelay ()
- void setDefaultdelay (double defaultdelay)
- boolean isEnableAnimationFlag ()

- void setEnableAnimationFlag (boolean enableAnimationFlag)
- ColoredValueLabel getValueLabel ()
- void setValueLabel (ColoredValueLabel valueLabel)
- FlexibleLabelFormatter getLabelFormatter ()
- void setLabelFormatter (FlexibleLabelFormatter labelFormatter)
- String getComponentType ()
- void setComponentType (String componentType)
- int getQTDE_PORTS ()
- int getTOT_PORTS ()
- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortEnOut ()
- void setPortEnOut (PortStdLogic1164 portEnOut)
- PortStdLogic1164 getPortRdy ()
- void setPortRdy (PortStdLogic1164 portRdy)
- PortStdLogicVector[] getPortDout ()
- void setPortDout (PortStdLogicVector[] portDout)
- PortStdLogic1164[] getPortRout ()
- void setPortRout (PortStdLogic1164[] portRout)
- PortStdLogicVector getPortDconf ()
- void setPortDconf (PortStdLogicVector portDconf)
- int[] getVectorIn ()
- int getIdxDin ()
- void setIdxDin (int idxDin)
- boolean isStart ()
- void setStart (boolean start)

### Protected Member Functions

- void constructStandardValues ()

### 3.23.1 Detailed Description

GenericIn component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components that implement input queues with 1, 2, 4, 8, 16, or 32 outputs.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.23.2 Constructor & Destructor Documentation

#### 3.23.2.1 add.dataflow.sync.GenericIn.GenericIn ( )

Object Constructor. By default, an input queue of an output is created.

#### 3.23.2.2 add.dataflow.sync.GenericIn.GenericIn ( int *QTDE_PORTS* )

Object Constructor. An input queue of N outputs is created.

**Parameters**

| | |
|---|---|
| *QTDE_PORTS* | - Number of queue outputs to be created |

### 3.23.3 Member Function Documentation

#### 3.23.3.1 void add.dataflow.sync.GenericIn.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.23.3.2 void add.dataflow.sync.GenericIn.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.23.3.3 void add.dataflow.sync.GenericIn.constructStandardValues ( ) [protected]

Method responsible for creating some auxiliary variables for working with bit vectors.

#### 3.23.3.4 void add.dataflow.sync.GenericIn.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected. It Will pass the vector data to the outputs.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.23.3.5 String add.dataflow.sync.GenericIn.getComponentType ( )

**Returns**

the componentType

#### 3.23.3.6 double add.dataflow.sync.GenericIn.getDefaultdelay ( )

**Returns**

the defaultdelay

#### 3.23.3.7 double add.dataflow.sync.GenericIn.getDelay ( )

Method responsible for returning the value of the delay variable that contains the response delay time of the component.

**Returns**

- Returns the dalay of the component.

#### 3.23.3.8 int add.dataflow.sync.GenericIn.getIdxDin ( )

**Returns**

the idxDin

#### 3.23.3.9 FlexibleLabelFormatter add.dataflow.sync.GenericIn.getLabelFormatter ( )

**Returns**

the labelFormatter

#### 3.23.3.10 int add.dataflow.sync.GenericIn.getN_bits ( )

**Returns**

the n_bits

#### 3.23.3.11 PortStdLogic1164 add.dataflow.sync.GenericIn.getPortClk ( )

**Returns**

the portClk

#### 3.23.3.12 PortStdLogicVector add.dataflow.sync.GenericIn.getPortDconf ( )

**Returns**

the portDconf

#### 3.23.3.13 PortStdLogicVector [ ] add.dataflow.sync.GenericIn.getPortDout ( )

**Returns**

the portDout

#### 3.23.3.14 PortStdLogic1164 add.dataflow.sync.GenericIn.getPortEnOut ( )

**Returns**

the portEnOut

#### 3.23.3.15 PortStdLogic1164 add.dataflow.sync.GenericIn.getPortRdy ( )

**Returns**

the portRdy

#### 3.23.3.16 PortStdLogic1164 [ ] add.dataflow.sync.GenericIn.getPortRout ( )

**Returns**

the portRout

#### 3.23.3.17 PortStdLogic1164 add.dataflow.sync.GenericIn.getPortRst ( )

**Returns**

the portRst

#### 3.23.3.18 int add.dataflow.sync.GenericIn.getQTDE_PORTS ( )

**Returns**

the QTDE_PORTS

#### 3.23.3.19 int add.dataflow.sync.GenericIn.getTOT_PORTS ( )

**Returns**

the TOT_PORTS

#### 3.23.3.20 ColoredValueLabel add.dataflow.sync.GenericIn.getValueLabel ( )

**Returns**

the valueLabel

#### 3.23.3.21 StdLogicVector add.dataflow.sync.GenericIn.getVector ( )

**Returns**

the vector

#### 3.23.3.22 StdLogicVector add.dataflow.sync.GenericIn.getVector_000 ( )

**Returns**

the vector_000

#### 3.23.3.23 StdLogicVector add.dataflow.sync.GenericIn.getVector_111 ( )

**Returns**

the vector_111

#### 3.23.3.24 StdLogicVector add.dataflow.sync.GenericIn.getVector_UUU ( )

**Returns**

the vector_UUU

#### 3.23.3.25 StdLogicVector add.dataflow.sync.GenericIn.getVector_XXX ( )

**Returns**

the vector_XXX

#### 3.23.3.26 StdLogicVector add.dataflow.sync.GenericIn.getVector_ZZZ ( )

**Returns**

the vector_ZZZ

#### 3.23.3.27 int [ ] add.dataflow.sync.GenericIn.getVectorIn ( )

**Returns**

the vectorIn

#### 3.23.3.28 PortStdLogicVector add.dataflow.sync.GenericIn.getVectorOutputPort ( )

**Returns**

the vectorOutputPort

#### 3.23.3.29 boolean add.dataflow.sync.GenericIn.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.23.3.30 boolean add.dataflow.sync.GenericIn.isEnableAnimationFlag ( )

**Returns**

the enableAnimationFlag

#### 3.23.3.31 boolean add.dataflow.sync.GenericIn.isStart ( )

**Returns**

the start

#### 3.23.3.32 boolean add.dataflow.sync.GenericIn.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be built dynamically.

#### 3.23.3.33 void add.dataflow.sync.GenericIn.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set to the component name. |

#### 3.23.3.34 void add.dataflow.sync.GenericIn.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.23.3.35 void add.dataflow.sync.GenericIn.setDefaultdelay ( double *defaultdelay* )

**Parameters**

| | |
|---|---|
| *defaultdelay* | the defaultdelay to set |

#### 3.23.3.36 void add.dataflow.sync.GenericIn.setDelay ( double *_delay* )

Method responsible for changing the value of the delay variable that contains the response delay time of the component.

**Parameters**

| | |
|---|---|
| *_delay* | |

#### 3.23.3.37 void add.dataflow.sync.GenericIn.setDelay ( String *s* )

Method responsible for changing the value of the delay variable that contains the response delay time of the component.

**Parameters**

| | |
|---|---|
| *s* | |

#### 3.23.3.38 void add.dataflow.sync.GenericIn.setEnableAnimationFlag ( boolean *enableAnimationFlag* )

**Parameters**

| | |
|---|---|
| *enable-AnimationFlag* | the enableAnimationFlag to set |

#### 3.23.3.39 void add.dataflow.sync.GenericIn.setIdxDin ( int *idxDin* )

**Parameters**

| | |
|---|---|
| *idxDin* | the idxDin to set |

#### 3.23.3.40 void add.dataflow.sync.GenericIn.setLabelFormatter ( FlexibleLabelFormatter *labelFormatter* )

**Parameters**

| | |
|---|---|
| *labelFormatter* | the labelFormatter to set |

#### 3.23.3.41 void add.dataflow.sync.GenericIn.setN_bits ( int *n_bits* )

**Parameters**

| | |
|---|---|
| *n_bits* | the n_bits to set |

#### 3.23.3.42 void add.dataflow.sync.GenericIn.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.23.3.43 void add.dataflow.sync.GenericIn.setPortDconf ( PortStdLogicVector *portDconf* )

**Parameters**

| | |
|---|---|
| *portDconf* | the portDconf to set |

#### 3.23.3.44 void add.dataflow.sync.GenericIn.setPortDout ( PortStdLogicVector[] *portDout* )

**Parameters**

| | |
|---|---|
| *portDout* | the portDout to set |

#### 3.23.3.45 void add.dataflow.sync.GenericIn.setPortEnOut ( PortStdLogic1164 *portEnOut* )

**Parameters**

| | |
|---|---|
| *portEnOut* | the portEnOut to set |

#### 3.23.3.46 void add.dataflow.sync.GenericIn.setPortRdy ( PortStdLogic1164 *portRdy* )

**Parameters**

| | |
|---|---|
| *portRdy* | the portRdy to set |

#### 3.23.3.47 void add.dataflow.sync.GenericIn.setPortRout ( PortStdLogic1164[] *portRout* )

**Parameters**

| | |
|---|---|
| *portRout* | the portRout to set |

#### 3.23.3.48 void add.dataflow.sync.GenericIn.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.23.3.49 void add.dataflow.sync.GenericIn.setStart ( boolean *start* )

**Parameters**

| | |
|---|---|
| *start* | the start to set |

#### 3.23.3.50 void add.dataflow.sync.GenericIn.setValueLabel ( ColoredValueLabel *valueLabel* )

**Parameters**

| | |
|---|---|
| *valueLabel* | the valueLabel to set |

#### 3.23.3.51 void add.dataflow.sync.GenericIn.setVector ( StdLogicVector *vector* )

**Parameters**

| | |
|---|---|
| *vector* | the vector to set |

#### 3.23.3.52 void add.dataflow.sync.GenericIn.setVector_000 ( StdLogicVector *vector_000* )

**Parameters**

| | |
|---|---|
| *vector_000* | the vector_000 to set |

#### 3.23.3.53 void add.dataflow.sync.GenericIn.setVector_111 ( StdLogicVector *vector_111* )

**Parameters**

| | |
|---|---|
| *vector_111* | the vector_111 to set |

#### 3.23.3.54 void add.dataflow.sync.GenericIn.setVector_UUU ( StdLogicVector *vector_UUU* )

**Parameters**

| | |
|---|---|
| *vector_UUU* | the vector_UUU to set |

#### 3.23.3.55 void add.dataflow.sync.GenericIn.setVector_XXX ( StdLogicVector *vector_XXX* )

**Parameters**

| | |
|---|---|
| *vector_XXX* | the vector_XXX to set |

#### 3.23.3.56 void add.dataflow.sync.GenericIn.setVector_ZZZ ( StdLogicVector *vector_ZZZ* )

**Parameters**

| | |
|---|---|
| *vector_ZZZ* | the vector_ZZZ to set |

#### 3.23.3.57 void add.dataflow.sync.GenericIn.setVectorIn ( int[ ] *vectorIn* )

Method responsible to set the data vector to be delivered to the outputs.

**Parameters**

| | |
|---|---|
| *vectorIn* | - Vector that will be delivered to the outputs |

#### 3.23.3.58 void add.dataflow.sync.GenericIn.setVectorOutputPort ( PortStdLogicVector *vectorOutputPort* )

**Parameters**

| | |
|---|---|
| *vectorOutputPort* | the vectorOutputPort to set |

#### 3.23.3.59 void add.dataflow.sync.GenericIn.updateSymbol ( )

Method responsible for updating the component symbol.

#### 3.23.3.60 void add.dataflow.sync.GenericIn.wakeup ( Object *arg* )

wakeup(): Called by the simulator as a reaction to our own scheduleWakeup()-calls. For RTLIB components, a wakeup() is normally used to update the value label on its graphical symbol. A WakeupEvent for this purpose should have either 'null' or the current 'this' object as its payload.

A second use is to update our internal 'vector' variable at a specified simulation time, which is needed to implement the assign() method from interface hades.simulator.Assignable. A WakeupEvent for this purpose is expected to hold a StdLogicVector object (with the 'value' from the assign call) as its payload.

**Parameters**

| | |
|---|---|
| *arg* | - Object to be awakened. |

#### 3.23.3.61 void add.dataflow.sync.GenericIn.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

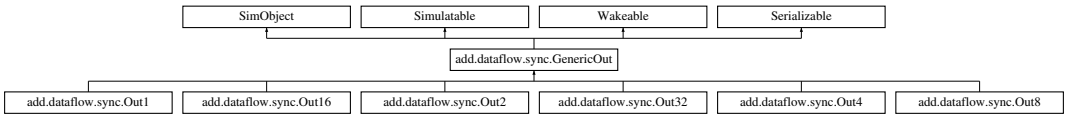
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericIn.java

## 3.24 add.dataflow.sync.GenericOut Class Reference

Inheritance diagram for add.dataflow.sync.GenericOut:




## Public Member Functions

- GenericOut ()
- GenericOut (int QTDE_PORTS)
- void constructPorts ()
- boolean getDoneSignal ()
- void setQtdeSave (int qtde_save)
- void setVector (int k)
- int[] getVectorOut ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)
- double getDelay ()
- void setDelay (double _delay)
- void setDelay (String s)
- void wakeup (Object arg)
- void updateSymbol ()
- int getN_bits ()
- void setN_bits (int n_bits)
- StdLogicVector getVector ()
- void setVector (StdLogicVector vector)
- StdLogicVector getVector_UUU ()
- void setVector_UUU (StdLogicVector vector_UUU)
- StdLogicVector getVector_XXX ()
- void setVector_XXX (StdLogicVector vector_XXX)
- StdLogicVector getVector_ZZZ ()
- void setVector_ZZZ (StdLogicVector vector_ZZZ)
- StdLogicVector getVector_000 ()
- void setVector_000 (StdLogicVector vector_000)
- StdLogicVector getVector_111 ()
- void setVector_111 (StdLogicVector vector_111)
- PortStdLogicVector getVectorOutputPort ()
- void setVectorOutputPort (PortStdLogicVector vectorOutputPort)
- double getDefaultdelay ()
- void setDefaultdelay (double defaultdelay)
- boolean isEnableAnimationFlag ()
- void setEnableAnimationFlag (boolean enableAnimationFlag)
- ColoredValueLabel getValueLabel ()
- void setValueLabel (ColoredValueLabel valueLabel)
- FlexibleLabelFormatter getLabelFormatter ()
- void setLabelFormatter (FlexibleLabelFormatter labelFormatter)
- String getComponentType ()
- void setComponentType (String componentType)
- int getQTDE_PORTS ()
- int getTOT_PORTS ()
- PortStdLogic1164 getPortClk ()

- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRdy ()
- void setPortRdy (PortStdLogic1164 portRdy)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector[] getPortDin ()
- void setPortDin (PortStdLogicVector[] portDin)
- PortStdLogic1164[] getPortRin ()
- void setPortRin (PortStdLogic1164[] portRin)
- void setVectorOut (int[] vectorOut)
- int getIdxDout ()
- void setIdxDout (int idxDout)
- int getTamVectorOut ()
- void setTamVectorOut (int tamVectorOut)
- boolean isDone ()
- void setDone (boolean done)
- int getQtdeSave ()

### Protected Member Functions

- void constructStandardValues ()

## 3.24.1 Detailed Description

GenericOut component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components that implement output queues with 1, 2, 4, 8, 16, or 32 inputs.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

## 3.24.2 Constructor & Destructor Documentation

### 3.24.2.1 add.dataflow.sync.GenericOut.GenericOut ( )

Object Constructor. By default, an input queue of an output is created.

### 3.24.2.2 add.dataflow.sync.GenericOut.GenericOut ( int *QTDE_PORTS* )

Object Constructor. An output queue of N inputs is created.

**Parameters**

| | |
|---|---|
| *QTDE_PORTS* | - Number of queue inputs to be created |

### 3.24.3 Member Function Documentation

#### 3.24.3.1 void add.dataflow.sync.GenericOut.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.24.3.2 void add.dataflow.sync.GenericOut.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.24.3.3 void add.dataflow.sync.GenericOut.constructStandardValues ( ) [protected]

Method responsible for creating some auxiliary variables for working with bit vectors.

#### 3.24.3.4 void add.dataflow.sync.GenericOut.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and if the R_IN inputs are high level. It Will pass the data from the inputs to the vector.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.24.3.5 String add.dataflow.sync.GenericOut.getComponentType ( )

**Returns**

the componentType

#### 3.24.3.6 double add.dataflow.sync.GenericOut.getDefaultdelay ( )

**Returns**

the defaultdelay

#### 3.24.3.7 double add.dataflow.sync.GenericOut.getDelay ( )

Method responsible for returning the value of the delay variable that contains the response delay time of the component.

**Returns**

- Returns component delay

#### 3.24.3.8 boolean add.dataflow.sync.GenericOut.getDoneSignal ( )

Method responsible for returning end of data entry.

**Returns**

- Returns the value of done signal.

#### 3.24.3.9 int add.dataflow.sync.GenericOut.getIdxDout ( )

**Returns**

the idxDout

#### 3.24.3.10 FlexibleLabelFormatter add.dataflow.sync.GenericOut.getLabelFormatter ( )

**Returns**

the labelFormatter

#### 3.24.3.11 int add.dataflow.sync.GenericOut.getN_bits ( )

**Returns**

the n_bits

#### 3.24.3.12 PortStdLogic1164 add.dataflow.sync.GenericOut.getPortClk ( )

**Returns**

the portClk

#### 3.24.3.13 PortStdLogicVector [ ] add.dataflow.sync.GenericOut.getPortDin ( )

**Returns**

the portDin

#### 3.24.3.14 PortStdLogic1164 add.dataflow.sync.GenericOut.getPortEn ( )

**Returns**

the portEn

#### 3.24.3.15 PortStdLogic1164 add.dataflow.sync.GenericOut.getPortRdy ( )

**Returns**

the portRdy

#### 3.24.3.16 PortStdLogic1164 [ ] add.dataflow.sync.GenericOut.getPortRin ( )

**Returns**

the portRin

#### 3.24.3.17 PortStdLogic1164 add.dataflow.sync.GenericOut.getPortRst ( )

**Returns**

the portRst

#### 3.24.3.18 int add.dataflow.sync.GenericOut.getQTDE_PORTS ( )

**Returns**

the QTDE_PORTS

#### 3.24.3.19 int add.dataflow.sync.GenericOut.getQtdeSave ( )

**Returns**

the qtdeSave

#### 3.24.3.20 int add.dataflow.sync.GenericOut.getTamVectorOut ( )

**Returns**

the tamVectorOut

#### 3.24.3.21 int add.dataflow.sync.GenericOut.getTOT_PORTS ( )

**Returns**

the TOT_PORTS

#### 3.24.3.22 ColoredValueLabel add.dataflow.sync.GenericOut.getValueLabel ( )

**Returns**

the valueLabel

#### 3.24.3.23 StdLogicVector add.dataflow.sync.GenericOut.getVector ( )

**Returns**

the vector

#### 3.24.3.24 StdLogicVector add.dataflow.sync.GenericOut.getVector_000 ( )

**Returns**

the vector_000

#### 3.24.3.25 StdLogicVector add.dataflow.sync.GenericOut.getVector_111 ( )

**Returns**

the vector_111

#### 3.24.3.26 StdLogicVector add.dataflow.sync.GenericOut.getVector_UUU ( )

**Returns**

the vector_UUU

#### 3.24.3.27 StdLogicVector add.dataflow.sync.GenericOut.getVector_XXX ( )

**Returns**

the vector_XXX

#### 3.24.3.28 StdLogicVector add.dataflow.sync.GenericOut.getVector_ZZZ ( )

**Returns**

the vector_ZZZ

#### 3.24.3.29 int [ ] add.dataflow.sync.GenericOut.getVectorOut ( )

Method responsible for returning the data vector received by the queue entries.

**Returns**

- Returns the vector with the processed data.

#### 3.24.3.30 PortStdLogicVector add.dataflow.sync.GenericOut.getVectorOutputPort ( )

**Returns**

the vectorOutputPort

#### 3.24.3.31 boolean add.dataflow.sync.GenericOut.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.24.3.32 boolean add.dataflow.sync.GenericOut.isDone ( )

**Returns**

the done

#### 3.24.3.33 boolean add.dataflow.sync.GenericOut.isEnableAnimationFlag ( )

**Returns**

the enableAnimationFlag

#### 3.24.3.34 boolean add.dataflow.sync.GenericOut.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be built dynamically.

#### 3.24.3.35 void add.dataflow.sync.GenericOut.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set to the component name. |

#### 3.24.3.36 void add.dataflow.sync.GenericOut.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.24.3.37 void add.dataflow.sync.GenericOut.setDefaultdelay ( double *defaultdelay* )

**Parameters**

| | |
|---|---|
| *defaultdelay* | the defaultdelay to set |

#### 3.24.3.38 void add.dataflow.sync.GenericOut.setDelay ( double *_delay* )

Method responsible for changing the value of the delay variable that contains the response delay time of the component.

**Parameters**

| | |
|---|---|
| *_delay* | |

#### 3.24.3.39 void add.dataflow.sync.GenericOut.setDelay ( String *s* )

Method responsible for changing the value of the delay variable that contains the response delay time of the component.

**Parameters**

| | |
|---|---|
| *s* | |

#### 3.24.3.40 void add.dataflow.sync.GenericOut.setDone ( boolean *done* )

**Parameters**

| | |
|---|---|
| *done* | the done to set |

#### 3.24.3.41 void add.dataflow.sync.GenericOut.setEnableAnimationFlag ( boolean *enableAnimationFlag* )

**Parameters**

| | |
|---|---|
| *enable-AnimationFlag* | the enableAnimationFlag to set |

#### 3.24.3.42 void add.dataflow.sync.GenericOut.setIdxDout ( int *idxDout* )

**Parameters**

| | |
|---|---|
| *idxDout* | the idxDout to set |

#### 3.24.3.43 void add.dataflow.sync.GenericOut.setLabelFormatter ( FlexibleLabelFormatter *labelFormatter* )

**Parameters**

| | |
|---|---|
| *labelFormatter* | the labelFormatter to set |

#### 3.24.3.44 void add.dataflow.sync.GenericOut.setN_bits ( int *n_bits* )

**Parameters**

| | |
|---|---|
| *n_bits* | the n_bits to set |

#### 3.24.3.45 void add.dataflow.sync.GenericOut.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.24.3.46 void add.dataflow.sync.GenericOut.setPortDin ( PortStdLogicVector[] *portDin* )

**Parameters**

| | |
|---|---|
| *portDin* | the portDin to set |

#### 3.24.3.47 void add.dataflow.sync.GenericOut.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.24.3.48 void add.dataflow.sync.GenericOut.setPortRdy ( PortStdLogic1164 *portRdy* )

**Parameters**

| | |
|---|---|
| *portRdy* | the portRdy to set |

#### 3.24.3.49 void add.dataflow.sync.GenericOut.setPortRin ( PortStdLogic1164[] *portRin* )

**Parameters**

| | |
|---|---|
| *portRin* | the portRin to set |

#### 3.24.3.50 void add.dataflow.sync.GenericOut.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.24.3.51 void add.dataflow.sync.GenericOut.setQtdeSave ( int *qtde_save* )

**Parameters**

| | |
|---|---|
| *qtde_save* | |

#### 3.24.3.52 void add.dataflow.sync.GenericOut.setTamVectorOut ( int *tamVectorOut* )

**Parameters**

| | |
|---|---|
| *tamVectorOut* | the tamVectorOut to set |

#### 3.24.3.53 void add.dataflow.sync.GenericOut.setValueLabel ( ColoredValueLabel *valueLabel* )

**Parameters**

| | |
|---|---|
| *valueLabel* | the valueLabel to set |

#### 3.24.3.54 void add.dataflow.sync.GenericOut.setVector ( int *k* )

Method responsible for inserting elements into the vector.

**Parameters**

| | |
|---|---|
| *k* | - Value to be inserted in vector. |

#### 3.24.3.55 void add.dataflow.sync.GenericOut.setVector ( StdLogicVector *vector* )

**Parameters**

| | |
|---|---|
| *vector* | the vector to set |

#### 3.24.3.56 void add.dataflow.sync.GenericOut.setVector_000 ( StdLogicVector *vector_000* )

**Parameters**

| | |
|---|---|
| *vector_000* | the vector_000 to set |

#### 3.24.3.57 void add.dataflow.sync.GenericOut.setVector_111 ( StdLogicVector *vector_111* )

**Parameters**

| | |
|---|---|
| *vector_111* | the vector_111 to set |

#### 3.24.3.58 void add.dataflow.sync.GenericOut.setVector_UUU ( StdLogicVector *vector_UUU* )

**Parameters**

| | |
|---|---|
| *vector_UUU* | the vector_UUU to set |

#### 3.24.3.59 void add.dataflow.sync.GenericOut.setVector_XXX ( StdLogicVector *vector_XXX* )

**Parameters**

| | |
|---|---|
| *vector_XXX* | the vector_XXX to set |

#### 3.24.3.60 void add.dataflow.sync.GenericOut.setVector_ZZZ ( StdLogicVector *vector_ZZZ* )

**Parameters**

| | |
|---|---|
| *vector_ZZZ* | the vector_ZZZ to set |

#### 3.24.3.61 void add.dataflow.sync.GenericOut.setVectorOut ( int[] *vectorOut* )

**Parameters**

| | |
|---|---|
| *vectorOut* | the vectorOut to set |

#### 3.24.3.62 void add.dataflow.sync.GenericOut.setVectorOutputPort ( PortStdLogicVector *vectorOutputPort* )

**Parameters**

| | |
|---|---|
| *vectorOutputPort* | the vectorOutputPort to set |

#### 3.24.3.63 void add.dataflow.sync.GenericOut.updateSymbol ( )

Method responsible for updating the component symbol.

#### 3.24.3.64 void add.dataflow.sync.GenericOut.wakeup ( Object *arg* )

wakeup(): Called by the simulator as a reaction to our own scheduleWakeup()-calls. For RTLIB components, a wakeup() is normally used to update the value label on its graphical symbol. A WakeupEvent for this purpose should have either 'null' or the current 'this' object as its payload.

A second use is to update our internal 'vector' variable at a specified simulation time, which is needed to implement the assign() method from interface hades.simulator.Assignable. A WakeupEvent for this purpose is expected to hold a StdLogicVector object (with the 'value' from the assign call) as its payload.

**Parameters**

| | |
|---|---|
| *arg* | - Object to be awakened. |

#### 3.24.3.65 void add.dataflow.sync.GenericOut.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

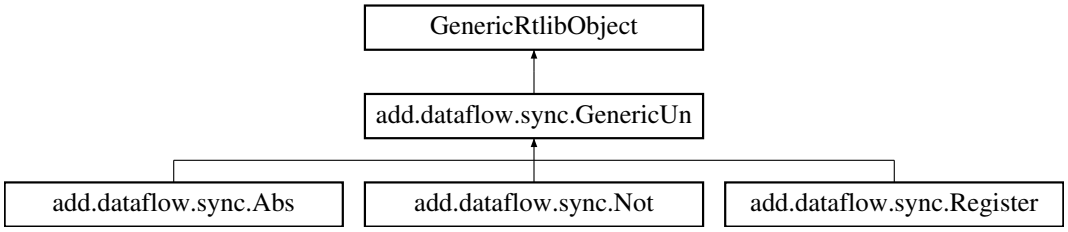
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericOut.java

## 3.25 add.dataflow.sync.GenericUn Class Reference

Inheritance diagram for add.dataflow.sync.GenericUn:




### Public Member Functions

- GenericUn ()
- void constructPorts ()
- void setString (String s)
- void setSymbol (Symbol s)
- int compute (int data)
- void notCompute ()
- void reseted ()
- void tickUp ()
- void tickDown ()
- void setCompName (String l)
- void evaluate (Object arg)
- boolean needsDynamicSymbol ()
- void constructDynamicSymbol ()
- void write (java.io.PrintWriter ps)
- boolean initialize (String s)

- Label getStringLabel ()
- void setStringLabel (Label stringLabel)
- Label getLabelNome ()
- void setLabelNome (Label labelNome)
- String getS ()
- void setS (String s)
- String getComponentType ()
- void setComponentType (String componentType)
- Rectangle getBackground ()
- void setBackground (Rectangle background)
- PortStdLogic1164 getPortClk ()
- void setPortClk (PortStdLogic1164 portClk)
- PortStdLogic1164 getPortRst ()
- void setPortRst (PortStdLogic1164 portRst)
- PortStdLogic1164 getPortRin ()
- void setPortRin (PortStdLogic1164 portRin)
- PortStdLogic1164 getPortRout ()
- void setPortRout (PortStdLogic1164 portRout)
- PortStdLogic1164 getPortEn ()
- void setPortEn (PortStdLogic1164 portEn)
- PortStdLogicVector getPortDin ()
- void setPortDin (PortStdLogicVector portDin)
- PortStdLogicVector getPortDout ()
- void setPortDout (PortStdLogicVector portDout)

### 3.25.1 Detailed Description

GenericUn component for the ADD Accelerator Design and Deploy.

The component creates the basis for other components with one input.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.25.2 Constructor & Destructor Documentation

#### 3.25.2.1 add.dataflow.sync.GenericUn.GenericUn ( )

Object Constructor.

### 3.25.3 Member Function Documentation

#### 3.25.3.1 int add.dataflow.sync.GenericUn.compute ( int *data* )

Method responsible for the computation of the output.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for the computation. |

**Returns**

- Return of computation

#### 3.25.3.2 void add.dataflow.sync.GenericUn.constructDynamicSymbol ( )

Method responsible for dynamically constructing the component symbol.

#### 3.25.3.3 void add.dataflow.sync.GenericUn.constructPorts ( )

Method responsible for initializing the component input and output ports.

#### 3.25.3.4 void add.dataflow.sync.GenericUn.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN input is high level. It will execute the reseted(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.25.3.5 Rectangle add.dataflow.sync.GenericUn.getBackground ( )

**Returns**

the background

#### 3.25.3.6 String add.dataflow.sync.GenericUn.getComponentType ( )

**Returns**

the componentType

#### 3.25.3.7 Label add.dataflow.sync.GenericUn.getLabelNome ( )

**Returns**

the labelNome

#### 3.25.3.8 PortStdLogic1164 add.dataflow.sync.GenericUn.getPortClk ( )

**Returns**

the portClk

#### 3.25.3.9 PortStdLogicVector add.dataflow.sync.GenericUn.getPortDin ( )

**Returns**

the portDin

#### 3.25.3.10 PortStdLogicVector add.dataflow.sync.GenericUn.getPortDout ( )

**Returns**

the portDout

#### 3.25.3.11 PortStdLogic1164 add.dataflow.sync.GenericUn.getPortEn ( )

**Returns**

the portEn

#### 3.25.3.12 PortStdLogic1164 add.dataflow.sync.GenericUn.getPortRin ( )

**Returns**

the portRin

#### 3.25.3.13 PortStdLogic1164 add.dataflow.sync.GenericUn.getPortRout ( )

**Returns**

the portRout

#### 3.25.3.14 PortStdLogic1164 add.dataflow.sync.GenericUn.getPortRst ( )

**Returns**

the portRst

#### 3.25.3.15 String add.dataflow.sync.GenericUn.getS ( )

**Returns**

the s

#### 3.25.3.16 Label add.dataflow.sync.GenericUn.getStringLabel ( )

**Returns**

the stringLabel

#### 3.25.3.17 boolean add.dataflow.sync.GenericUn.initialize ( String *s* )

Method responsible for reading the component settings in the file saved by the simulator.

**Parameters**

| | |
|---|---|
| *s* | - Settings for the component read from the file saved by the simulator. |

**Returns**

- Returns true if the settings are read successfully.

#### 3.25.3.18 boolean add.dataflow.sync.GenericUn.needsDynamicSymbol ( )

Method responsible for indicating to the simulator that the component's symbol will be constructed dynamically by the constructDynamicSymbol() method, or will be read from a file of the same name as the ".sym" extension.

**Returns**

- TRUE means that the symbol will be made dynamically.

#### 3.25.3.19 void add.dataflow.sync.GenericUn.notCompute ( )

Method executed when computing is not performed. In this case it clears the text displayed by the component.

#### 3.25.3.20 void add.dataflow.sync.GenericUn.reseted ( )

Method executed when the signal from the reset input goes to high logic level.In this case it clears the text displayed by the component.

#### 3.25.3.21 void add.dataflow.sync.GenericUn.setBackground ( Rectangle *background* )

**Parameters**

| | |
|---|---|
| *background* | the background to set |

#### 3.25.3.22 void add.dataflow.sync.GenericUn.setCompName ( String *l* )

Method responsible for changing the label that displays the name of the component.

**Parameters**

| | |
|---|---|
| *l* | - String to be set in component name. |

#### 3.25.3.23 void add.dataflow.sync.GenericUn.setComponentType ( String *componentType* )

**Parameters**

| | |
|---|---|
| *componentType* | the componentType to set |

#### 3.25.3.24 void add.dataflow.sync.GenericUn.setLabelNome ( Label *labelNome* )

**Parameters**

| | |
|---|---|
| *labelNome* | the labelNome to set |

#### 3.25.3.25 void add.dataflow.sync.GenericUn.setPortClk ( PortStdLogic1164 *portClk* )

**Parameters**

| | |
|---|---|
| *portClk* | the portClk to set |

#### 3.25.3.26 void add.dataflow.sync.GenericUn.setPortDin ( PortStdLogicVector *portDin* )

**Parameters**

| | |
|---|---|
| *portDin* | the portDin to set |

#### 3.25.3.27 void add.dataflow.sync.GenericUn.setPortDout ( PortStdLogicVector *portDout* )

**Parameters**

| | |
|---|---|
| *portDout* | the portDout to set |

#### 3.25.3.28 void add.dataflow.sync.GenericUn.setPortEn ( PortStdLogic1164 *portEn* )

**Parameters**

| | |
|---|---|
| *portEn* | the portEn to set |

#### 3.25.3.29 void add.dataflow.sync.GenericUn.setPortRin ( PortStdLogic1164 *portRin* )

**Parameters**

| | |
|---|---|
| *portRin* | the portRin to set |

#### 3.25.3.30 void add.dataflow.sync.GenericUn.setPortRout ( PortStdLogic1164 *portRout* )

**Parameters**

| | |
|---|---|
| *portRout* | the portRout to set |

#### 3.25.3.31 void add.dataflow.sync.GenericUn.setPortRst ( PortStdLogic1164 *portRst* )

**Parameters**

| | |
|---|---|
| *portRst* | the portRst to set |

#### 3.25.3.32 void add.dataflow.sync.GenericUn.setS ( String *s* )

**Parameters**

| | |
|---|---|
| *s* | the s to set |

#### 3.25.3.33 void add.dataflow.sync.GenericUn.setString ( String *s* )

Method responsible for updating the text displayed by the component.

**Parameters**

| | |
|---|---|
| *s* | - Text to be updated. |

#### 3.25.3.34 void add.dataflow.sync.GenericUn.setStringLabel ( Label *stringLabel* )

**Parameters**

| | |
|---|---|
| *stringLabel* | the stringLabel to set |

#### 3.25.3.35 void add.dataflow.sync.GenericUn.setSymbol ( Symbol *s* )

Method responsible for updating the component symbol.

**Parameters**

| | |
|---|---|
| *s* | - Symbol passed automatically. |

#### 3.25.3.36 void add.dataflow.sync.GenericUn.tickDown ( )

Method executed when the clock signal goes to low logic level.

#### 3.25.3.37 void add.dataflow.sync.GenericUn.tickUp ( )

Method executed when the clock signal goes to high logic level.

#### 3.25.3.38 void add.dataflow.sync.GenericUn.write ( java.io.PrintWriter *ps* )

Method responsible for writing component settings to the file saved by the simulator.

**Parameters**

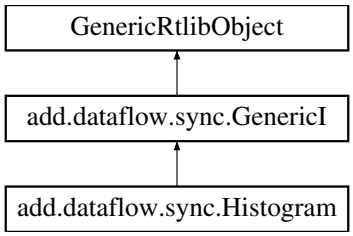
| | |
|---|---|
| *ps* | -Simulator writing object. |

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

- add/dataflow/sync/GenericUn.java

## 3.26 add.dataflow.sync.Histogram Class Reference

Inheritance diagram for add.dataflow.sync.Histogram:




### Public Member Functions

- Histogram ()
- int compute (int data)
- void reset ()
- void evaluate (Object arg)
- int[] getHistogram ()
- void setHistogram (int[] histogram)
- int getCounter ()
- void setCounter (int counter)
- int getDecr ()
- void setDecr (int decr)
- int getNUMBITS ()
- void setNUMBITS (int NUMBITS)

## 3.26.1 Detailed Description

Histogram component for the UFV synchronous data flow simulator.

The component is responsible for computing the amount of times a given value is delivered at its input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

## 3.26.2 Constructor & Destructor Documentation

### 3.26.2.1 add.dataflow.sync.Histogram.Histogram ( )

Object Constructor.

## 3.26.3 Member Function Documentation

### 3.26.3.1 int add.dataflow.sync.Histogram.compute ( int *data* )

Method responsible for the component computation: in this case it performs the logical operation "AND" between the parameter and the (immediate) id.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

- Returns the result of the computation. In this case the result of the logical operation "AND" between the parameter and the id.

#### 3.26.3.2 void add.dataflow.sync.Histogram.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events.

In this case, it will be checked whether the ports are connected and will execute the compute (int data) method if the R_IN input is high level. It will execute the reset(), tickUp(), and tickDown() methods if their respective entries order it. It will update the output with the compute(int data) method result.

**Parameters**

| | |
|---|---|
| *arg* | an arbitrary object argument |

#### 3.26.3.3 int add.dataflow.sync.Histogram.getCounter ( )

**Returns**

the counter

#### 3.26.3.4 int add.dataflow.sync.Histogram.getDecr ( )

**Returns**

the decr

#### 3.26.3.5 int [ ] add.dataflow.sync.Histogram.getHistogram ( )

**Returns**

the histogram

#### 3.26.3.6 int add.dataflow.sync.Histogram.getNUMBITS ( )

**Returns**

the NUMBITS

#### 3.26.3.7 void add.dataflow.sync.Histogram.reset ( )

Method executed when the signal from the reset input goes to high logic level. It sets the new text to be shown by the component. In this case the id.

#### 3.26.3.8 void add.dataflow.sync.Histogram.setCounter ( int *counter* )

**Parameters**

| | |
|---|---|
| *counter* | the counter to set |

#### 3.26.3.9 void add.dataflow.sync.Histogram.setDecr ( int *decr* )

**Parameters**

| | |
|---|---|
| *decr* | the decr to set |

#### 3.26.3.10 void add.dataflow.sync.Histogram.setHistogram ( int[ ] *histogram* )

**Parameters**

| | |
|---|---|
| *histogram* | the histogram to set |

#### 3.26.3.11 void add.dataflow.sync.Histogram.setNUMBITS ( int *NUMBITS* )

**Parameters**

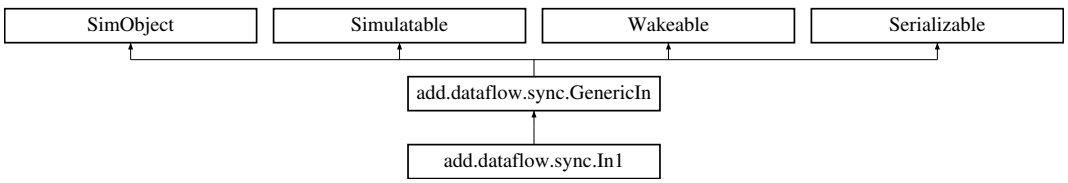
| | |
|---|---|
| *NUMBITS* | the NUMBITS to set |

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

- add/dataflow/sync/Histogram.java

## 3.27 add.dataflow.sync.In1 Class Reference

Inheritance diagram for add.dataflow.sync.In1:




### Public Member Functions

- In1 ()

### Additional Inherited Members

### 3.27.1 Detailed Description

In1 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 1 output.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.27.2 Constructor & Destructor Documentation

#### 3.27.2.1 add.dataflow.sync.In1.In1 ( )

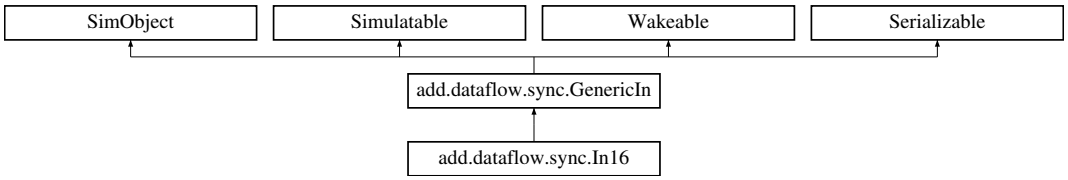
Object Constructor.

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

- add/dataflow/sync/In1.java

## 3.28 add.dataflow.sync.In16 Class Reference

Inheritance diagram for add.dataflow.sync.In16:




### Public Member Functions

- In16 ()

### Additional Inherited Members

### 3.28.1 Detailed Description

In16 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 16 output.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.28.2 Constructor & Destructor Documentation

#### 3.28.2.1 add.dataflow.sync.In16.In16 ( )

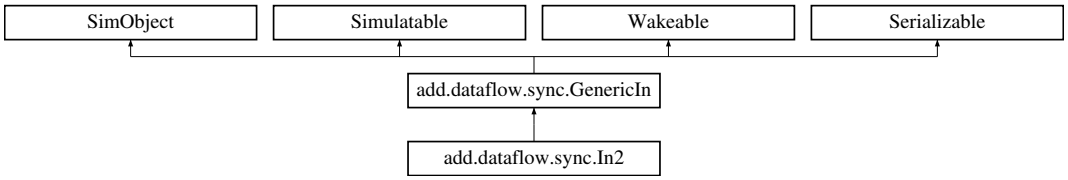
Object Constructor.

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

- add/dataflow/sync/In16.java

## 3.29 add.dataflow.sync.In2 Class Reference

Inheritance diagram for add.dataflow.sync.In2:




### Public Member Functions

- In2 ()

### Additional Inherited Members

### 3.29.1 Detailed Description

In2 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 2 output.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.29.2 Constructor & Destructor Documentation

#### 3.29.2.1 add.dataflow.sync.In2.In2 ( )

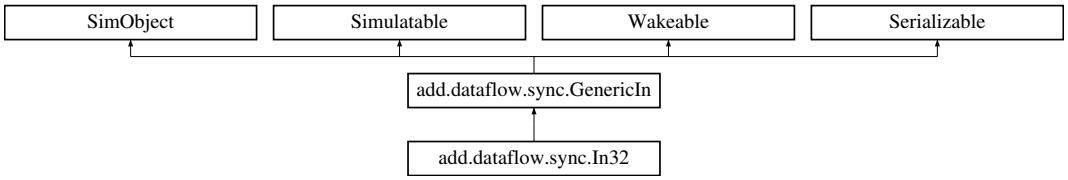
Object Constructor.

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

- add/dataflow/sync/In2.java

## 3.30 add.dataflow.sync.In32 Class Reference

Inheritance diagram for add.dataflow.sync.In32:




### Public Member Functions

- In32 ()

### Additional Inherited Members

### 3.30.1 Detailed Description

In32 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 32 output.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.30.2 Constructor & Destructor Documentation

#### 3.30.2.1 add.dataflow.sync.In32.In32 ( )

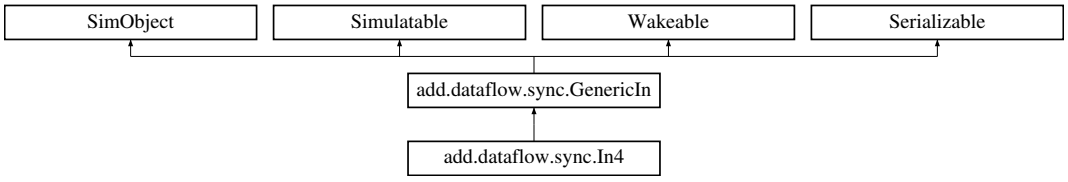
Object Constructor.

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

- add/dataflow/sync/In32.java

## 3.31 add.dataflow.sync.In4 Class Reference

Inheritance diagram for add.dataflow.sync.In4:



### Public Member Functions

- In4 ()

### Additional Inherited Members

#### 3.31.1 Detailed Description

In4 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 4 output.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

#### 3.31.2 Constructor & Destructor Documentation

##### 3.31.2.1 add.dataflow.sync.In4.In4 ( )

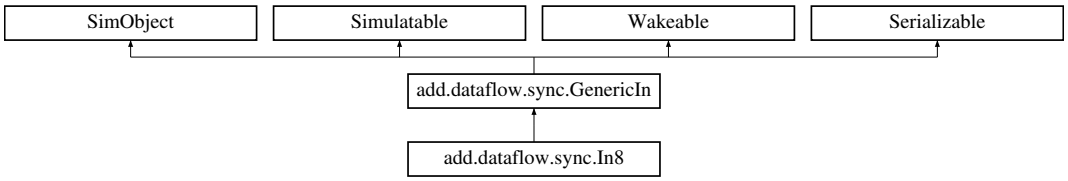
Object Constructor.

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

- add/dataflow/sync/In4.java

## 3.32 add.dataflow.sync.In8 Class Reference

Inheritance diagram for add.dataflow.sync.In8:




### Public Member Functions

- In8 ()

### Additional Inherited Members

#### 3.32.1 Detailed Description

In8 component for the ADD Accelerator Design and Deploy.

The component implements an input queue with 8 output.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.32.2 Constructor & Destructor Documentation

#### 3.32.2.1 add.dataflow.sync.In8.In8 ( )

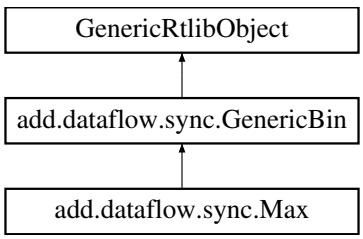
Object Constructor.

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

- add/dataflow/sync/In8.java

## 3.33 add.dataflow.sync.Max Class Reference

Inheritance diagram for add.dataflow.sync.Max:




### Public Member Functions

- Max ()
- int compute (int data1, int data2)

### 3.33.1 Detailed Description

Max component for the ADD Accelerator Design and Deploy.

The component is responsible for passing the output to the largest value input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.33.2 Constructor & Destructor Documentation

#### 3.33.2.1 add.dataflow.sync.Max.Max ( )

Object Constructor.

### 3.33.3 Member Function Documentation

#### 3.33.3.1 int add.dataflow.sync.Max.compute ( int *data1*, int *data2* )

Method responsible for the computation of components: in this case, it performs a comparison between the parameters and returns the largest between the two.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

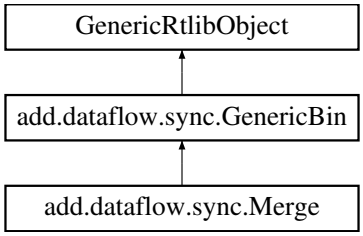
- Returns the result of the computation. In this case, the largest of the parameters.

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

- add/dataflow/sync/Max.java

## 3.34 add.dataflow.sync.Merge Class Reference

Inheritance diagram for add.dataflow.sync.Merge:




### Public Member Functions

- Merge ()
- void evaluate (Object arg)

### 3.34.1 Detailed Description

Merge component for the ADD Accelerator Design and Deploy.

The component is responsible for choosing which of the inputs to pass to the output depending on the value of R_IN1 and R_IN2.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.34.2 Constructor & Destructor Documentation

#### 3.34.2.1 add.dataflow.sync.Merge.Merge ( )

Object Constructor.

### 3.34.3 Member Function Documentation

#### 3.34.3.1 void add.dataflow.sync.Merge.evaluate ( Object *arg* )

evaluate(): called by the simulation engine on all events that concern this object. The object is responsible for updating its internal state and for scheduling all pending output events. In this case, it will be checked if any of the R_IN (1 or 2) inputs is at high level and put the respective input value in the output. If the two R_IN signals are at high level, the value of input 1 will be set to the output. If both are 0, nothing will be done.

**Parameters**

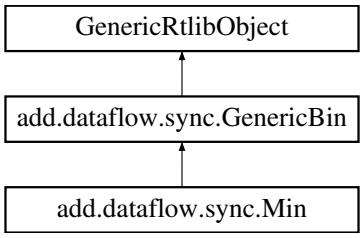
| | |
|---|---|
| *arg* | an arbitrary object argument |

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

- add/dataflow/sync/Merge.java

## 3.35 add.dataflow.sync.Min Class Reference

Inheritance diagram for add.dataflow.sync.Min:




### Public Member Functions

- Min ()
- int compute (int data1, int data2)

### 3.35.1 Detailed Description

Min component for the ADD Accelerator Design and Deploy.

The component is responsible for passing the output to the lowest value input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.35.2 Constructor & Destructor Documentation

#### 3.35.2.1 add.dataflow.sync.Min.Min ( )

Object Constructor.

### 3.35.3 Member Function Documentation

#### 3.35.3.1 int add.dataflow.sync.Min.compute ( int *data1*, int *data2* )

Method responsible for the computation of components: in this case, it performs a comparison between the parameters and returns the smaller between the two.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

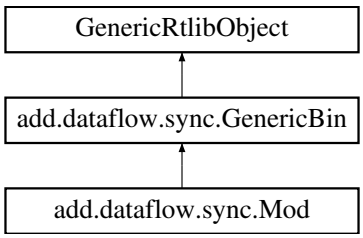
- Returns the result of the computation. In this case, the smallest of the parameters.

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

- add/dataflow/sync/Min.java

## 3.36 add.dataflow.sync.Mod Class Reference

Inheritance diagram for add.dataflow.sync.Mod:




### Public Member Functions

- Mod ()
- int compute (int data1, int data2)

### 3.36.1 Detailed Description

Mod component for the ADD Accelerator Design and Deploy.

The component is responsible for calculating the rest of the integer division of the first input by the second one.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.36.2 Constructor & Destructor Documentation

#### 3.36.2.1 add.dataflow.sync.Mod.Mod ( )

Object Constructor.

### 3.36.3 Member Function Documentation

#### 3.36.3.1 int add.dataflow.sync.Mod.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case, it returns the rest of the division between the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

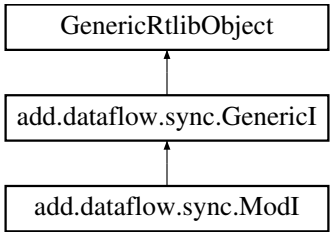
- Returns the result of the computation. In this case, it returns the rest of the division between the parameters.

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

- add/dataflow/sync/Mod.java

## 3.37 add.dataflow.sync.ModI Class Reference

Inheritance diagram for add.dataflow.sync.ModI:



### Public Member Functions

- ModI ()
- int compute (int data)

### 3.37.1 Detailed Description

ModI component for the ADD Accelerator Design and Deploy.

The component is responsible for calculating the rest of the integer division of the input by a id (immediate).

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.37.2 Constructor & Destructor Documentation

#### 3.37.2.1 add.dataflow.sync.ModI.ModI ( )

Object Constructor.

### 3.37.3 Member Function Documentation

#### 3.37.3.1 int add.dataflow.sync.ModI.compute ( int *data* )

Method responsible for the component computation: in this case, it returns the rest of the division of the parameter by the id.

#### Parameters

| | |
|---|---|
| *data* | - Value to be used for computing. |

#### Returns

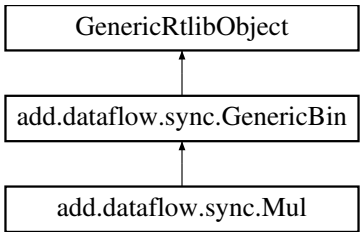
- Returns the result of the computation. In this case, it returns the rest of the division of the parameter by the id.

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

- add/dataflow/sync/ModI.java

## 3.38 add.dataflow.sync.Mul Class Reference

Inheritance diagram for add.dataflow.sync.Mul:




### Public Member Functions

- Mul ()
- int compute (int data1, int data2)

### 3.38.1 Detailed Description

Mul component for the ADD Accelerator Design and Deploy.

The component is responsible for multiplying the inputs.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.38.2 Constructor & Destructor Documentation

#### 3.38.2.1 add.dataflow.sync.Mul.Mul ( )

Object Constructor.

### 3.38.3 Member Function Documentation

#### 3.38.3.1 int add.dataflow.sync.Mul.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case performs a multiplication of the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

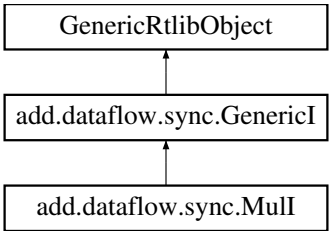
- Returns the result of the computation. In this case the value of the multiplication of the parameters.

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

- add/dataflow/sync/Mul.java

## 3.39 add.dataflow.sync.MulI Class Reference

Inheritance diagram for add.dataflow.sync.MulI:




### Public Member Functions

- MulI ()
- int compute (int data)

### 3.39.1 Detailed Description

MulI component for the ADD Accelerator Design and Deploy.

The component is responsible for multiplying the input by a (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.39.2 Constructor & Destructor Documentation

#### 3.39.2.1 add.dataflow.sync.MulI.MulI ( )

Object Constructor.

### 3.39.3 Member Function Documentation

#### 3.39.3.1 int add.dataflow.sync.MulI.compute ( int *data* )

Method responsible for the component computation: in this case performs a multiplying of the parameter by an (immediate) id.

#### Parameters

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

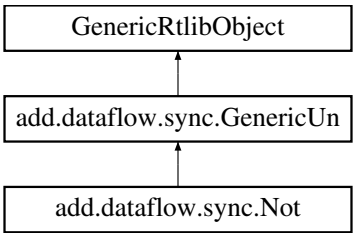
- Returns the result of the computation. In this case the value of the multiplication of the parameter by the id.

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

- add/dataflow/sync/Mull.java

## 3.40 add.dataflow.sync.Not Class Reference

Inheritance diagram for add.dataflow.sync.Not:




### Public Member Functions

- Not ()
- int compute (int data)

### 3.40.1 Detailed Description

Not component for the ADD Accelerator Design and Deploy.

The component is responsible for the bitwise inversion of the input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.40.2 Constructor & Destructor Documentation

#### 3.40.2.1 add.dataflow.sync.Not.Not ( )

Object Constructor.

### 3.40.3 Member Function Documentation

#### 3.40.3.1 int add.dataflow.sync.Not.compute ( int *data* )

Method responsible for the component computation: in this case performs a bitwise inversion of the parameter.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

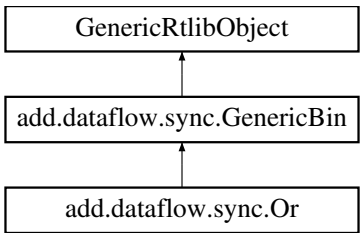
- Returns the result of the computation. In this case the value of the bitwise inversion of the parameter.

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

- add/dataflow/sync/Not.java

## 3.41 add.dataflow.sync.Or Class Reference

Inheritance diagram for add.dataflow.sync.Or:




### Public Member Functions

- Or ()
- int compute (int data1, int data2)

### 3.41.1 Detailed Description

Or component for the ADD Accelerator Design and Deploy.

The component is responsible for the logical operation "Or" between the input

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.41.2 Constructor & Destructor Documentation

#### 3.41.2.1 add.dataflow.sync.Or.Or ( )

Object Constructor.

### 3.41.3 Member Function Documentation

#### 3.41.3.1 int add.dataflow.sync.Or.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case it performs the logical operation "Or" between the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

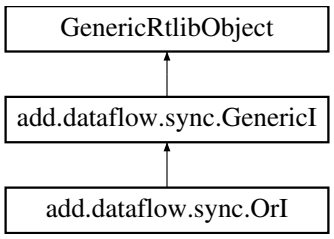
- Returns the result of the computation. In this case the result of the logical operation "Or" between the parameters.

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

- add/dataflow/sync/Or.java

## 3.42 add.dataflow.sync.OrI Class Reference

Inheritance diagram for add.dataflow.sync.OrI:




### Public Member Functions

- OrI ()
- int compute (int data)

### 3.42.1 Detailed Description

OrI component for the ADD Accelerator Design and Deploy.

The component is responsible for the logical operation "OR" between the input and a id (immediate)

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.42.2 Constructor & Destructor Documentation

#### 3.42.2.1 add.dataflow.sync.OrI.OrI ( )

Object Constructor.

### 3.42.3 Member Function Documentation

#### 3.42.3.1 int add.dataflow.sync.Orl.compute ( int *data* )

Method responsible for the component computation: in this case it performs the logical operation "OR" between the parameter and the (immediate) id.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

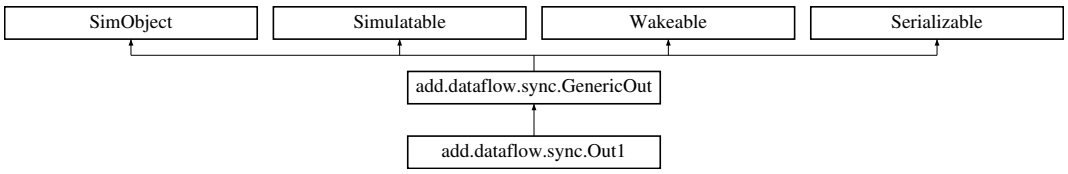
- Returns the result of the computation. In this case the result of the logical operation "OR" between the parameter and the id.

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

- add/dataflow/sync/OrI.java

## 3.43 add.dataflow.sync.Out1 Class Reference

Inheritance diagram for add.dataflow.sync.Out1:




### Public Member Functions

- Out1 ()

### Additional Inherited Members

### 3.43.1 Detailed Description

Out1 component for the ADD Accelerator Design and Deploy.

The component implements an output queue with 1 input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.43.2 Constructor & Destructor Documentation

#### 3.43.2.1 add.dataflow.sync.Out1.Out1 ( )

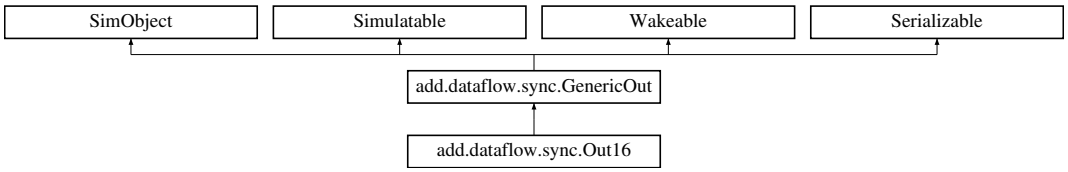
Object Constructor.

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

- add/dataflow/sync/Out1.java

## 3.44 add.dataflow.sync.Out16 Class Reference

Inheritance diagram for add.dataflow.sync.Out16:




### Public Member Functions

- Out16 ()

### Additional Inherited Members

### 3.44.1 Detailed Description

Out16 component for the ADD Accelerator Design and Deploy.

O implements an input queue with 16 output.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.44.2 Constructor & Destructor Documentation

#### 3.44.2.1 add.dataflow.sync.Out16.Out16 ( )

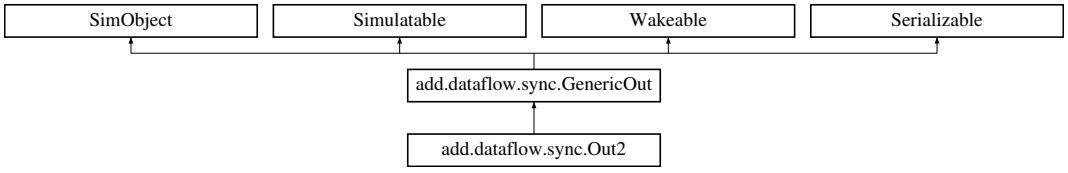
Object Constructor.

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

- add/dataflow/sync/Out16.java

## 3.45 add.dataflow.sync.Out2 Class Reference

Inheritance diagram for add.dataflow.sync.Out2:



### Public Member Functions

- Out2 ()

### Additional Inherited Members

#### 3.45.1 Detailed Description

Out2 component for the ADD Accelerator Design and Deploy.

The component implements an output queue with 2 input..

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

#### 3.45.2 Constructor & Destructor Documentation

##### 3.45.2.1 add.dataflow.sync.Out2.Out2 ( )

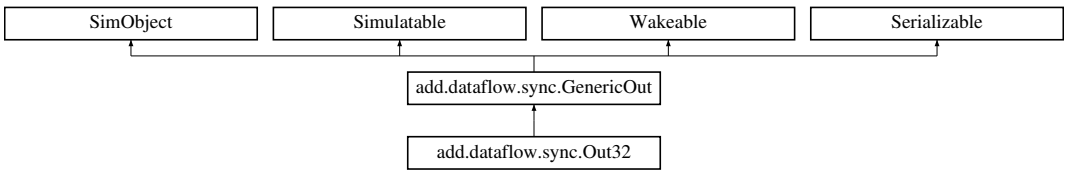
Object Constructor.

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

- add/dataflow/sync/Out2.java

## 3.46 add.dataflow.sync.Out32 Class Reference

Inheritance diagram for add.dataflow.sync.Out32:




### Public Member Functions

- Out32 ()

### Additional Inherited Members

#### 3.46.1 Detailed Description

Out32 component for the ADD Accelerator Design and Deploy.

The component implements an output queue with 32 input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.46.2 Constructor & Destructor Documentation

#### 3.46.2.1 add.dataflow.sync.Out32.Out32 ( )

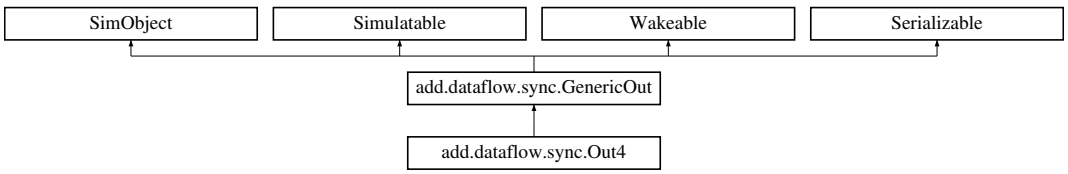
Object Constructor.

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

- add/dataflow/sync/Out32.java

## 3.47 add.dataflow.sync.Out4 Class Reference

Inheritance diagram for add.dataflow.sync.Out4:




### Public Member Functions

- Out4 ()

### Additional Inherited Members

### 3.47.1 Detailed Description

Out4 component for the ADD Accelerator Design and Deploy.

The component implements an output queue with 4 input.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.47.2 Constructor & Destructor Documentation

#### 3.47.2.1 add.dataflow.sync.Out4.Out4 ( )

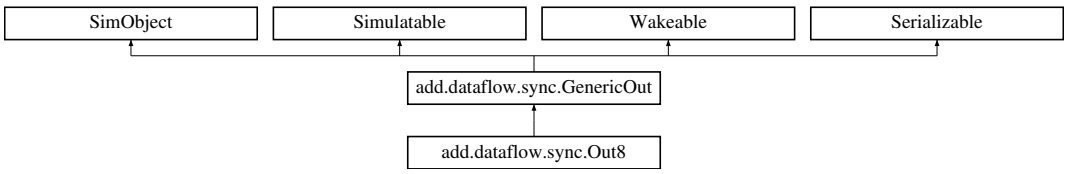
Object Constructor.

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

- add/dataflow/sync/Out4.java

## 3.48 add.dataflow.sync.Out8 Class Reference

Inheritance diagram for add.dataflow.sync.Out8:




### Public Member Functions

- Out8 ()

### Additional Inherited Members

### 3.48.1 Detailed Description

Out8 component for the ADD Accelerator Design and Deploy.

The component implements an output queue with 8 input.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.48.2 Constructor & Destructor Documentation

#### 3.48.2.1 add.dataflow.sync.Out8.Out8 ( )

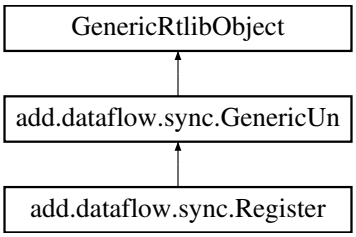
Object Constructor.

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

- add/dataflow/sync/Out8.java

## 3.49 add.dataflow.sync.Register Class Reference

Inheritance diagram for add.dataflow.sync.Register:




### Public Member Functions

- Register ()

### 3.49.1 Detailed Description

Register component for the ADD Accelerator Design and Deploy.

The component is responsible for pass the input to the output when a clock pulse occurs.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.49.2 Constructor & Destructor Documentation

#### 3.49.2.1 add.dataflow.sync.Register.Register ( )

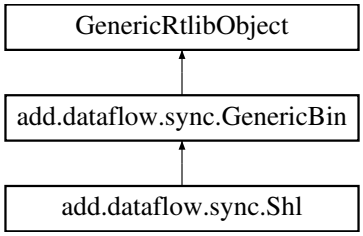
Object Constructor.

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

- add/dataflow/sync/Register.java

## 3.50 add.dataflow.sync.Shl Class Reference

Inheritance diagram for add.dataflow.sync.Shl:



### Public Member Functions

- Shl ()
- int compute (int data1, int data2)

### 3.50.1 Detailed Description

Shl component for the ADD Accelerator Design and Deploy.

The component is responsible for moving all the bits of the first input to the left N times, where N is equal to the value of the second input.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.50.2 Constructor & Destructor Documentation

#### 3.50.2.1 add.dataflow.sync.Shl.Shl ( )

Object Constructor.

### 3.50.3 Member Function Documentation

#### 3.50.3.1 int add.dataflow.sync.Shl.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case, it moves all the bits of the first parameter to the left N times, where N is equal to the value of the second parameter.

**Parameters**

| | |
|---|---|
| *data1* | - Value 1 to be used for computing. |
| *data2* | - Value 2 to be used for computing. |

**Returns**

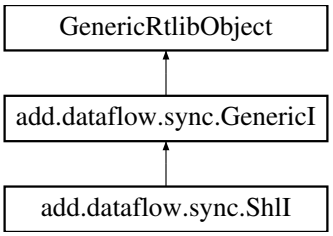
- Returns the result of the computation. In this case, it moves all the bits of the first parameter to the left N times, where N is equal to the value of second parameter.

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

- add/dataflow/sync/Shl.java

## 3.51 add.dataflow.sync.ShlI Class Reference

Inheritance diagram for add.dataflow.sync.ShlI:




### Public Member Functions

- ShlI ()
- int compute (int data)

### 3.51.1 Detailed Description

ShlI component for the ADD Accelerator Design and Deploy.

The component is responsible for moving all the bits of the input to the left N times, where N is equal to the value of a (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.51.2 Constructor & Destructor Documentation

#### 3.51.2.1 add.dataflow.sync.ShlI.ShlI ( )

Object Constructor.

### 3.51.3 Member Function Documentation

#### 3.51.3.1 int add.dataflow.sync.ShlI.compute ( int *data* )

Method responsible for the component computation: in this case, it moves all the bits of the parameter to the left N times, where N is equal to the value of a (immediate) id.

```
@param data - Value to be used for computing.
@return - Returns the result of the computation. In this case, it moves
```

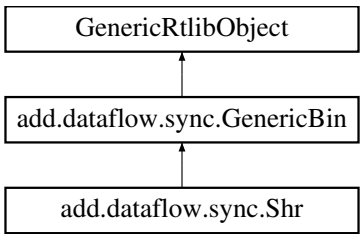
all the bits of the parameter to the left N times, where N is equal to the value of a (immediate) id.

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

- add/dataflow/sync/ShlI.java

## 3.52 add.dataflow.sync.Shr Class Reference

Inheritance diagram for add.dataflow.sync.Shr:




### Public Member Functions

- Shr ()
- int compute (int data1, int data2)

### 3.52.1 Detailed Description

Shr component for the ADD Accelerator Design and Deploy.

The component is responsible for moving all the bits of the first input to the right N times, where N is equal to the value of the value from the second input.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.52.2 Constructor & Destructor Documentation

#### 3.52.2.1 add.dataflow.sync.Shr.Shr ( )

Object Constructor.

### 3.52.3 Member Function Documentation

#### 3.52.3.1 int add.dataflow.sync.Shr.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case, it moves all the bits of the first parameter to the right N times, where N is equal to the value of the second parameter.

#### Parameters

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |

| | |
|---|---|
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

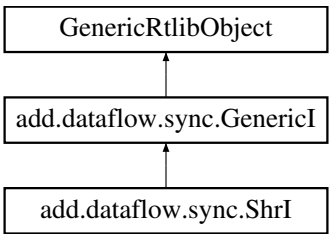
- Returns the result of the computation. In this case, it moves all the bits of the first parameter to the right N times, where N is equal to the value of the second parameter.

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

- add/dataflow/sync/Shr.java

## 3.53 add.dataflow.sync.ShrI Class Reference

Inheritance diagram for add.dataflow.sync.ShrI:




### Public Member Functions

- ShrI ()
- int compute (int data)

### 3.53.1 Detailed Description

ShrI component for the ADD Accelerator Design and Deploy.

The component is responsible for moving all the bits of the input to the right N times, where N is equal to the value of a (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.53.2 Constructor & Destructor Documentation

#### 3.53.2.1 add.dataflow.sync.ShrI.ShrI ( )

Object Constructor.

### 3.53.3 Member Function Documentation

#### 3.53.3.1 int add.dataflow.sync.Shrl.compute ( int *data* )

Method responsible for the component computation: in this case, it moves all the bits of the parameter to the right N times, where N is equal to the value of a (immediate) id.

```
@param data - Value to be used for computing.
@return - Returns the result of the computation. In this case, it moves
```

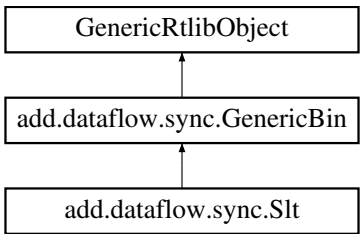
all the bits of the parameter to the right N times, where N is equal to the value of a (immediate) id.

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

- add/dataflow/sync/Shrl.java

## 3.54 add.dataflow.sync.Slt Class Reference

Inheritance diagram for add.dataflow.sync.Slt:




### Public Member Functions

- Slt ()
- int compute (int data1, int data2)

### 3.54.1 Detailed Description

Slt component for the ADD Accelerator Design and Deploy.

The component is responsible for returning the value 1 if the first input is less than the second one.

Universidade Federal de Viçosa - MG - Brasil.

#### Author

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

#### Version

1.0

### 3.54.2 Constructor & Destructor Documentation

#### 3.54.2.1 add.dataflow.sync.Slt.Slt ( )

Object Constructor.

### 3.54.3 Member Function Documentation

#### 3.54.3.1 int add.dataflow.sync.Slt.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case performs a comparison if the first parameter is less than the other one.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

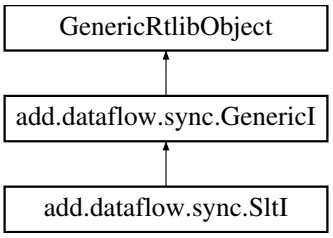
- Returns the result of the computation. In this case 1 or 0 depending on the comparison between the parameters.

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

- add/dataflow/sync/Slt.java

## 3.55 add.dataflow.sync.SltI Class Reference

Inheritance diagram for add.dataflow.sync.SltI:




### Public Member Functions

- SltI ()
- int compute (int data)

### 3.55.1 Detailed Description

SltI component for the ADD Accelerator Design and Deploy.

The component is responsible for returning the value 1 if the input is less than the (immediate) id.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.55.2 Constructor & Destructor Documentation

#### 3.55.2.1 add.dataflow.sync.SltI.SltI ( )

Object Constructor.

### 3.55.3 Member Function Documentation

#### 3.55.3.1 int add.dataflow.sync.SltI.compute ( int *data* )

Method responsible for the component computation: in this case performs a comparison if parameter is less than the (immediate) id.

**Parameters**

| | |
|---|---|
| *data* | - Value to be used for computing. |

**Returns**

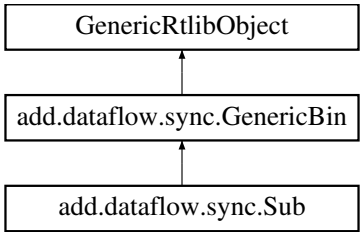
- Returns the result of the computation. In this case 1 or 0 depending on the comparison between the parameter and the id.

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

- add/dataflow/sync/SltI.java

## 3.56 add.dataflow.sync.Sub Class Reference

Inheritance diagram for add.dataflow.sync.Sub:




### Public Member Functions

- Sub ()
- int compute (int data1, int data2)

### 3.56.1 Detailed Description

Sub component for the ADD Accelerator Design and Deploy.

The component is responsible for subtracting the inputs.

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.56.2 Constructor & Destructor Documentation

#### 3.56.2.1 add.dataflow.sync.Sub.Sub ( )

Object Constructor.

### 3.56.3 Member Function Documentation

#### 3.56.3.1 int add.dataflow.sync.Sub.compute ( int *data1*, int *data2* )

Method responsible for the component computation: in this case performs a subtraction of the parameters.

**Parameters**

| | |
|---|---|
| *data1* | - Value to be used for the computation related to input 1. |
| *data2* | - Value to be used for the computation related to input 2. |

**Returns**

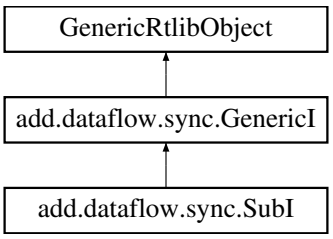
- Returns the result of the computation. In this case the value of the subtraction of the parameters.

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

- add/dataflow/sync/Sub.java

## 3.57 add.dataflow.sync.SubI Class Reference

Inheritance diagram for add.dataflow.sync.SubI:




### Public Member Functions

- SubI ()
- int compute (int data)

### 3.57.1 Detailed Description

SubI component for the ADD Accelerator Design and Deploy.

The component is responsible for subtracting the input by a id (immediate).

Universidade Federal de Viçosa - MG - Brasil.

**Author**

Jeronimo Costa Penha - jeronimopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

**Version**

1.0

### 3.57.2 Constructor & Destructor Documentation

#### 3.57.2.1 add.dataflow.sync.SubI.SubI ( )

Object Constructor.

### 3.57.3 Member Function Documentation

#### 3.57.3.1 int add.dataflow.sync.SubI.compute ( int *data* )

Method responsible for the component computation: in this case performs a subtraction of the parameter by an (immediate) id.

```
@param data - Value to be used for computing.
@return - Returns the result of the computation. In this case the value
```

of the subtraction of the parameter by the id.

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

- add/dataflow/sync/SubI.java

# Index