

HEBE SYNCHRONOUS DATAFLOW DOCUMENTATION

1.0

Wed Aug 23 2017 19:20:04

Contents

| | | |
|----------|---|----------|
| 1 | Hierarchical Index | 1 |
| 1.1 | Class Hierarchy | 1 |
| 2 | Class Index | 3 |
| 2.1 | Class List | 3 |
| 3 | Class Documentation | 5 |
| 3.1 | hebe.dataflow.sync.Abs Class Reference | 5 |
| 3.1.1 | Detailed Description | 5 |
| 3.1.2 | Constructor & Destructor Documentation | 5 |
| 3.1.2.1 | Abs | 5 |
| 3.1.3 | Member Function Documentation | 6 |
| 3.1.3.1 | compute | 6 |
| 3.2 | hebe.dataflow.sync.AccAdd Class Reference | 6 |
| 3.2.1 | Detailed Description | 6 |
| 3.2.2 | Constructor & Destructor Documentation | 7 |
| 3.2.2.1 | AccAdd | 7 |
| 3.3 | hebe.dataflow.sync.AccMax Class Reference | 7 |
| 3.3.1 | Detailed Description | 7 |
| 3.3.2 | Constructor & Destructor Documentation | 7 |
| 3.3.2.1 | AccMax | 7 |
| 3.3.3 | Member Function Documentation | 8 |
| 3.3.3.1 | accumulate | 8 |
| 3.3.3.2 | reset | 8 |
| 3.4 | hebe.dataflow.sync.AccMin Class Reference | 8 |
| 3.4.1 | Detailed Description | 8 |
| 3.4.2 | Constructor & Destructor Documentation | 9 |
| 3.4.2.1 | AccMin | 9 |
| 3.4.3 | Member Function Documentation | 9 |
| 3.4.3.1 | accumulate | 9 |
| 3.4.3.2 | reset | 9 |
| 3.5 | hebe.dataflow.sync.AccMul Class Reference | 9 |

| | | |
|----------|--|----|
| 3.5.1 | Detailed Description | 10 |
| 3.5.2 | Constructor & Destructor Documentation | 10 |
| 3.5.2.1 | AccMul | 10 |
| 3.5.3 | Member Function Documentation | 10 |
| 3.5.3.1 | accumulate | 10 |
| 3.5.3.2 | reset | 10 |
| 3.6 | hebe.dataflow.sync.Add Class Reference | 10 |
| 3.6.1 | Detailed Description | 11 |
| 3.6.2 | Constructor & Destructor Documentation | 11 |
| 3.6.2.1 | Add | 11 |
| 3.6.3 | Member Function Documentation | 11 |
| 3.6.3.1 | compute | 11 |
| 3.7 | hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulation Class Reference | 11 |
| 3.7.1 | Detailed Description | 11 |
| 3.8 | hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulationWithGeneratedHds Class Reference | 12 |
| 3.8.1 | Detailed Description | 12 |
| 3.9 | hebe.dataflow.sync.AddI Class Reference | 12 |
| 3.9.1 | Detailed Description | 13 |
| 3.9.2 | Constructor & Destructor Documentation | 13 |
| 3.9.2.1 | AddI | 13 |
| 3.9.3 | Member Function Documentation | 13 |
| 3.9.3.1 | compute | 13 |
| 3.10 | hebe.examples.dataflow_sync.AddIDataflowFpgaSimulation Class Reference | 13 |
| 3.10.1 | Detailed Description | 14 |
| 3.11 | hebe.dataflow.AFU Class Reference | 14 |
| 3.12 | hebe.dataflow.sync.And Class Reference | 14 |
| 3.12.1 | Detailed Description | 15 |
| 3.12.2 | Constructor & Destructor Documentation | 15 |
| 3.12.2.1 | And | 15 |
| 3.12.3 | Member Function Documentation | 15 |
| 3.12.3.1 | compute | 15 |
| 3.13 | hebe.dataflow.sync.AndI Class Reference | 15 |
| 3.13.1 | Detailed Description | 16 |
| 3.13.2 | Constructor & Destructor Documentation | 16 |
| 3.13.2.1 | AndI | 16 |
| 3.13.3 | Member Function Documentation | 16 |
| 3.13.3.1 | compute | 16 |
| 3.14 | hebe.dataflow.sync.Beq Class Reference | 16 |
| 3.14.1 | Detailed Description | 17 |

| | | |
|----------|---|----|
| 3.14.2 | Constructor & Destructor Documentation | 17 |
| 3.14.2.1 | Beq | 17 |
| 3.14.3 | Member Function Documentation | 17 |
| 3.14.3.1 | compute | 17 |
| 3.15 | hebe.dataflow.sync.BeqI Class Reference | 18 |
| 3.15.1 | Detailed Description | 18 |
| 3.15.2 | Constructor & Destructor Documentation | 18 |
| 3.15.2.1 | BeqI | 18 |
| 3.15.3 | Member Function Documentation | 18 |
| 3.15.3.1 | compute | 18 |
| 3.16 | hebe.dataflow.sync.Bne Class Reference | 19 |
| 3.16.1 | Detailed Description | 19 |
| 3.16.2 | Constructor & Destructor Documentation | 19 |
| 3.16.2.1 | Bne | 19 |
| 3.16.3 | Member Function Documentation | 20 |
| 3.16.3.1 | compute | 20 |
| 3.17 | hebe.dataflow.sync.BneI Class Reference | 20 |
| 3.17.1 | Detailed Description | 20 |
| 3.17.2 | Constructor & Destructor Documentation | 21 |
| 3.17.2.1 | BneI | 21 |
| 3.17.3 | Member Function Documentation | 21 |
| 3.17.3.1 | compute | 21 |
| 3.18 | hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulation Class Reference | 21 |
| 3.18.1 | Detailed Description | 21 |
| 3.19 | hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulationWithGeneratedHds Class Reference | 22 |
| 3.19.1 | Detailed Description | 22 |
| 3.20 | hebe.util.ConfReader Class Reference | 22 |
| 3.20.1 | Detailed Description | 22 |
| 3.20.2 | Member Function Documentation | 22 |
| 3.20.2.1 | ReadConfig | 22 |
| 3.21 | hebe.dataflow.DataflowSyncSimulBase Class Reference | 23 |
| 3.21.1 | Detailed Description | 23 |
| 3.21.2 | Member Function Documentation | 23 |
| 3.21.2.1 | execFpga | 23 |
| 3.21.2.2 | execHades | 23 |
| 3.21.2.3 | startFpgaJtag | 24 |
| 3.21.2.4 | startFpgaJtag | 24 |
| 3.21.2.5 | startSimulation | 24 |
| 3.21.2.6 | startSimulation | 25 |

| | | |
|----------|---|----|
| 3.22 | hebe.dataflow.sync.Div Class Reference | 25 |
| 3.22.1 | Detailed Description | 25 |
| 3.22.2 | Constructor & Destructor Documentation | 26 |
| 3.22.2.1 | Div | 26 |
| 3.22.3 | Member Function Documentation | 26 |
| 3.22.3.1 | compute | 26 |
| 3.23 | hebe.dataflow.sync.Div1 Class Reference | 26 |
| 3.23.1 | Detailed Description | 27 |
| 3.23.2 | Constructor & Destructor Documentation | 27 |
| 3.23.2.1 | Div1 | 27 |
| 3.23.3 | Member Function Documentation | 27 |
| 3.23.3.1 | compute | 27 |
| 3.24 | hebe.examples.dataflow_sync.Fir16DataflowFPGA Class Reference | 27 |
| 3.24.1 | Detailed Description | 27 |
| 3.25 | hebe.examples.dataflow_sync.Fir16DataflowSimulation Class Reference | 28 |
| 3.25.1 | Detailed Description | 28 |
| 3.26 | hebe.examples.dataflow_sync.Fir4DataflowHadesSimulation Class Reference | 28 |
| 3.26.1 | Detailed Description | 28 |
| 3.27 | hebe.examples.dataflow_sync.Fir4DataflowHadesSimulationWithGeneratedHds Class Reference | 29 |
| 3.27.1 | Detailed Description | 29 |
| 3.28 | hebe.examples.dataflow_sync.Fir8DataflowFPGA Class Reference | 29 |
| 3.28.1 | Detailed Description | 29 |
| 3.29 | hebe.examples.dataflow_sync.Fir8DataflowSimulation Class Reference | 30 |
| 3.29.1 | Detailed Description | 30 |
| 3.30 | hebe.dataflow.sync.GenericAcc Class Reference | 30 |
| 3.30.1 | Detailed Description | 31 |
| 3.30.2 | Constructor & Destructor Documentation | 31 |
| 3.30.2.1 | GenericAcc | 31 |
| 3.30.3 | Member Function Documentation | 31 |
| 3.30.3.1 | accumulate | 31 |
| 3.30.3.2 | evaluate | 31 |
| 3.30.3.3 | getAcc | 31 |
| 3.30.3.4 | getCounter | 31 |
| 3.30.3.5 | reset | 32 |
| 3.30.3.6 | setAcc | 32 |
| 3.30.3.7 | setCounter | 32 |
| 3.31 | hebe.dataflow.sync.GenericBin Class Reference | 32 |
| 3.31.1 | Detailed Description | 33 |
| 3.31.2 | Constructor & Destructor Documentation | 34 |
| 3.31.2.1 | GenericBin | 34 |

| | |
|--------------------------------------|----|
| 3.31.3 Member Function Documentation | 34 |
| 3.31.3.1 compute | 34 |
| 3.31.3.2 constructDynamicSymbol | 34 |
| 3.31.3.3 constructPorts | 34 |
| 3.31.3.4 evaluate | 34 |
| 3.31.3.5 getComponentType | 34 |
| 3.31.3.6 getLabelNome | 35 |
| 3.31.3.7 getPortClk | 35 |
| 3.31.3.8 getPortDin1 | 35 |
| 3.31.3.9 getPortDin2 | 35 |
| 3.31.3.10 getPortDout | 35 |
| 3.31.3.11 getPortEn | 35 |
| 3.31.3.12 getPortRin1 | 35 |
| 3.31.3.13 getPortRin2 | 35 |
| 3.31.3.14 getPortRout | 35 |
| 3.31.3.15 getPortRst | 36 |
| 3.31.3.16 getS | 36 |
| 3.31.3.17 getStringLabel | 36 |
| 3.31.3.18 initialize | 36 |
| 3.31.3.19 needsDynamicSymbol | 36 |
| 3.31.3.20 notCompute | 36 |
| 3.31.3.21 reseted | 36 |
| 3.31.3.22 setCompName | 36 |
| 3.31.3.23 setComponentType | 36 |
| 3.31.3.24 setLabelNome | 37 |
| 3.31.3.25 setPortClk | 37 |
| 3.31.3.26 setPortDin1 | 37 |
| 3.31.3.27 setPortDin2 | 37 |
| 3.31.3.28 setPortDout | 37 |
| 3.31.3.29 setPortEn | 37 |
| 3.31.3.30 setPortRin1 | 37 |
| 3.31.3.31 setPortRin2 | 37 |
| 3.31.3.32 setPortRout | 38 |
| 3.31.3.33 setPortRst | 38 |
| 3.31.3.34 setS | 38 |
| 3.31.3.35 setString | 38 |
| 3.31.3.36 setStringLabel | 38 |
| 3.31.3.37 setSymbol | 38 |
| 3.31.3.38 tickDown | 38 |
| 3.31.3.39 tickUp | 38 |

| | |
|---|----|
| 3.31.3.40 write | 39 |
| 3.32 hebe.dataflow.sync.GenericBranch Class Reference | 39 |
| 3.32.1 Detailed Description | 40 |
| 3.32.2 Constructor & Destructor Documentation | 40 |
| 3.32.2.1 GenericBranch | 40 |
| 3.32.3 Member Function Documentation | 40 |
| 3.32.3.1 compute | 40 |
| 3.32.3.2 constructDynamicSymbol | 40 |
| 3.32.3.3 constructPorts | 41 |
| 3.32.3.4 evaluate | 41 |
| 3.32.3.5 getComponentType | 41 |
| 3.32.3.6 getLabel_nome | 41 |
| 3.32.3.7 getPortClk | 41 |
| 3.32.3.8 getPortDin1 | 41 |
| 3.32.3.9 getPortDin2 | 41 |
| 3.32.3.10 getPortElse | 41 |
| 3.32.3.11 getPortEn | 42 |
| 3.32.3.12 getPortIf | 42 |
| 3.32.3.13 getPortRin1 | 42 |
| 3.32.3.14 getPortRin2 | 42 |
| 3.32.3.15 getPortRst | 42 |
| 3.32.3.16 getS | 42 |
| 3.32.3.17 getStringLabel | 42 |
| 3.32.3.18 initialize | 42 |
| 3.32.3.19 needsDynamicSymbol | 43 |
| 3.32.3.20 reseted | 43 |
| 3.32.3.21 setCompName | 43 |
| 3.32.3.22 setComponentType | 43 |
| 3.32.3.23 setLabel_nome | 43 |
| 3.32.3.24 setPortClk | 43 |
| 3.32.3.25 setPortDin1 | 43 |
| 3.32.3.26 setPortDin2 | 43 |
| 3.32.3.27 setPortElse | 44 |
| 3.32.3.28 setPortEn | 44 |
| 3.32.3.29 setPortIf | 44 |
| 3.32.3.30 setPortRin1 | 44 |
| 3.32.3.31 setPortRin2 | 44 |
| 3.32.3.32 setPortRst | 44 |
| 3.32.3.33 setS | 44 |
| 3.32.3.34 setString | 44 |

| | |
|--|----|
| 3.32.3.35 setStringLabel | 45 |
| 3.32.3.36 setSymbol | 45 |
| 3.32.3.37 tickDown | 45 |
| 3.32.3.38 tickUp | 45 |
| 3.32.3.39 write | 45 |
| 3.33 hebe.dataflow.sync.GenericBranchI Class Reference | 45 |
| 3.33.1 Detailed Description | 47 |
| 3.33.2 Constructor & Destructor Documentation | 47 |
| 3.33.2.1 GenericBranchI | 47 |
| 3.33.3 Member Function Documentation | 47 |
| 3.33.3.1 compute | 47 |
| 3.33.3.2 constructDynamicSymbol | 47 |
| 3.33.3.3 constructPorts | 47 |
| 3.33.3.4 evaluate | 47 |
| 3.33.3.5 getBackground | 48 |
| 3.33.3.6 getComponentId | 48 |
| 3.33.3.7 getComponentImmediate | 48 |
| 3.33.3.8 getComponentType | 48 |
| 3.33.3.9 getId | 48 |
| 3.33.3.10 getImmediate | 48 |
| 3.33.3.11 getLabelNome | 48 |
| 3.33.3.12 getPortClk | 48 |
| 3.33.3.13 getPortDconf | 49 |
| 3.33.3.14 getPortDin | 49 |
| 3.33.3.15 getPortElse | 49 |
| 3.33.3.16 getPortEn | 49 |
| 3.33.3.17 getPortIf | 49 |
| 3.33.3.18 getPortRin | 49 |
| 3.33.3.19 getPortRst | 49 |
| 3.33.3.20 getStringLabelId | 49 |
| 3.33.3.21 getStringLabelImmediate | 49 |
| 3.33.3.22 initialize | 50 |
| 3.33.3.23 mousePressed | 51 |
| 3.33.3.24 needsDynamicSymbol | 51 |
| 3.33.3.25 reseted | 51 |
| 3.33.3.26 setBackground | 51 |
| 3.33.3.27 setCompName | 51 |
| 3.33.3.28 setComponentId | 51 |
| 3.33.3.29 setComponentImmediate | 52 |
| 3.33.3.30 setComponentType | 52 |

| | | |
|-----------|---|----|
| 3.33.3.31 | setId | 52 |
| 3.33.3.32 | setImmediate | 52 |
| 3.33.3.33 | setLabelNome | 52 |
| 3.33.3.34 | setPortClk | 52 |
| 3.33.3.35 | setPortDconf | 52 |
| 3.33.3.36 | setPortDin | 52 |
| 3.33.3.37 | setPortElse | 53 |
| 3.33.3.38 | setPortEn | 53 |
| 3.33.3.39 | setPortIf | 53 |
| 3.33.3.40 | setPortRin | 53 |
| 3.33.3.41 | setPortRst | 53 |
| 3.33.3.42 | setString | 53 |
| 3.33.3.43 | setStringLabelId | 53 |
| 3.33.3.44 | setStringLabelImmediate | 53 |
| 3.33.3.45 | setSymbol | 54 |
| 3.33.3.46 | tickDown | 54 |
| 3.33.3.47 | tickUp | 54 |
| 3.33.3.48 | write | 54 |
| 3.34 | hebe.dataflow.sync.Genericl Class Reference | 54 |
| 3.34.1 | Detailed Description | 56 |
| 3.34.2 | Constructor & Destructor Documentation | 56 |
| 3.34.2.1 | Genericl | 56 |
| 3.34.3 | Member Function Documentation | 56 |
| 3.34.3.1 | compute | 56 |
| 3.34.3.2 | constructDynamicSymbol | 57 |
| 3.34.3.3 | constructPorts | 57 |
| 3.34.3.4 | evaluate | 57 |
| 3.34.3.5 | getBackground | 57 |
| 3.34.3.6 | getComponentId | 57 |
| 3.34.3.7 | getComponentImmediate | 57 |
| 3.34.3.8 | getComponentType | 57 |
| 3.34.3.9 | getId | 58 |
| 3.34.3.10 | getImmediate | 58 |
| 3.34.3.11 | setLabelNome | 58 |
| 3.34.3.12 | getPortClk | 58 |
| 3.34.3.13 | getPortDconf | 58 |
| 3.34.3.14 | getPortDin | 58 |
| 3.34.3.15 | getPortDout | 58 |
| 3.34.3.16 | getPortEn | 58 |
| 3.34.3.17 | getPortRin | 58 |

| | | |
|-----------|--|----|
| 3.34.3.18 | getPortRout | 59 |
| 3.34.3.19 | getPortRst | 59 |
| 3.34.3.20 | getStringLabelId | 59 |
| 3.34.3.21 | getStringLabelImmediate | 59 |
| 3.34.3.22 | initialize | 59 |
| 3.34.3.23 | mousePressed | 59 |
| 3.34.3.24 | needsDynamicSymbol | 59 |
| 3.34.3.25 | notCompute | 60 |
| 3.34.3.26 | reset | 60 |
| 3.34.3.27 | setBackground | 60 |
| 3.34.3.28 | setCompName | 60 |
| 3.34.3.29 | setComponentId | 60 |
| 3.34.3.30 | setComponentImmediate | 60 |
| 3.34.3.31 | setComponentType | 60 |
| 3.34.3.32 | setId | 60 |
| 3.34.3.33 | setImmediate | 60 |
| 3.34.3.34 | setLabelNome | 61 |
| 3.34.3.35 | setPortClk | 61 |
| 3.34.3.36 | setPortDconf | 61 |
| 3.34.3.37 | setPortDin | 61 |
| 3.34.3.38 | setPortDout | 61 |
| 3.34.3.39 | setPortEn | 61 |
| 3.34.3.40 | setPortRin | 61 |
| 3.34.3.41 | setPortRout | 61 |
| 3.34.3.42 | setPortRst | 62 |
| 3.34.3.43 | setString | 62 |
| 3.34.3.44 | getStringLabelId | 62 |
| 3.34.3.45 | getStringLabelImmediate | 62 |
| 3.34.3.46 | setSymbol | 62 |
| 3.34.3.47 | tickDown | 62 |
| 3.34.3.48 | tickUp | 62 |
| 3.34.3.49 | write | 63 |
| 3.35 | hebe.dataflow.sync.GenericIn Class Reference | 64 |
| 3.35.1 | Detailed Description | 65 |
| 3.35.2 | Constructor & Destructor Documentation | 66 |
| 3.35.2.1 | GenericIn | 66 |
| 3.35.2.2 | GenericIn | 66 |
| 3.35.3 | Member Function Documentation | 66 |
| 3.35.3.1 | constructDynamicSymbol | 66 |
| 3.35.3.2 | constructPorts | 66 |

| | | |
|-----------|---|----|
| 3.35.3.3 | constructStandardValues | 66 |
| 3.35.3.4 | evaluate | 66 |
| 3.35.3.5 | getComponentType | 66 |
| 3.35.3.6 | getDefaultdelay | 66 |
| 3.35.3.7 | getDelay | 67 |
| 3.35.3.8 | getIdxDin | 67 |
| 3.35.3.9 | getLabelFormatter | 67 |
| 3.35.3.10 | getN_bits | 67 |
| 3.35.3.11 | getPortClk | 67 |
| 3.35.3.12 | getPortDconf | 67 |
| 3.35.3.13 | getPortDout | 67 |
| 3.35.3.14 | getPortEnOut | 67 |
| 3.35.3.15 | getPortRdy | 68 |
| 3.35.3.16 | getPortRout | 68 |
| 3.35.3.17 | getPortRst | 68 |
| 3.35.3.18 | getQTDE_PORTS | 68 |
| 3.35.3.19 | getTOT_PORTS | 68 |
| 3.35.3.20 | getValueLabel | 68 |
| 3.35.3.21 | getVector | 68 |
| 3.35.3.22 | getVector_000 | 68 |
| 3.35.3.23 | getVector_111 | 68 |
| 3.35.3.24 | getVector_UUU | 69 |
| 3.35.3.25 | getVector_XXX | 69 |
| 3.35.3.26 | getVector_ZZZ | 69 |
| 3.35.3.27 | getVectorIn | 69 |
| 3.35.3.28 | getVectorOutputPort | 69 |
| 3.35.3.29 | initialize | 69 |
| 3.35.3.30 | isEnabledAnimationFlag | 69 |
| 3.35.3.31 | isStart | 69 |
| 3.35.3.32 | needsDynamicSymbol | 70 |
| 3.35.3.33 | setCompName | 70 |
| 3.35.3.34 | setComponentType | 70 |
| 3.35.3.35 | setDefaultdelay | 70 |
| 3.35.3.36 | setDelay | 70 |
| 3.35.3.37 | setDelay | 70 |
| 3.35.3.38 | setEnabledAnimationFlag | 70 |
| 3.35.3.39 | setIdxDin | 71 |
| 3.35.3.40 | setLabelFormatter | 71 |
| 3.35.3.41 | setN_bits | 71 |
| 3.35.3.42 | setPortClk | 71 |

| | |
|--|----|
| 3.35.3.43 setPortDconf | 71 |
| 3.35.3.44 setPortDout | 71 |
| 3.35.3.45 setPortEnOut | 71 |
| 3.35.3.46 setPortRdy | 71 |
| 3.35.3.47 setPortRout | 72 |
| 3.35.3.48 setPortRst | 72 |
| 3.35.3.49 setStart | 72 |
| 3.35.3.50 setValueLabel | 72 |
| 3.35.3.51 setVector | 72 |
| 3.35.3.52 setVector_000 | 72 |
| 3.35.3.53 setVector_111 | 72 |
| 3.35.3.54 setVector_UUU | 72 |
| 3.35.3.55 setVector_XXX | 73 |
| 3.35.3.56 setVector_ZZZ | 73 |
| 3.35.3.57 setVectorIn | 73 |
| 3.35.3.58 setVectorOutputPort | 73 |
| 3.35.3.59 updateSymbol | 73 |
| 3.35.3.60 wakeup | 73 |
| 3.35.3.61 write | 73 |
| 3.36 hebe.dataflow.sync.GenericOut Class Reference | 74 |
| 3.36.1 Detailed Description | 75 |
| 3.36.2 Constructor & Destructor Documentation | 76 |
| 3.36.2.1 GenericOut | 76 |
| 3.36.2.2 GenericOut | 76 |
| 3.36.3 Member Function Documentation | 76 |
| 3.36.3.1 constructDynamicSymbol | 76 |
| 3.36.3.2 constructPorts | 76 |
| 3.36.3.3 constructStandardValues | 76 |
| 3.36.3.4 evaluate | 76 |
| 3.36.3.5 getComponentType | 76 |
| 3.36.3.6 getDefaultdelay | 76 |
| 3.36.3.7 getDelay | 77 |
| 3.36.3.8 getDoneSignal | 77 |
| 3.36.3.9 getIdxDout | 77 |
| 3.36.3.10 getLabelFormatter | 77 |
| 3.36.3.11 getN_bits | 77 |
| 3.36.3.12 getPortClk | 77 |
| 3.36.3.13 getPortDin | 77 |
| 3.36.3.14 getPortEn | 77 |
| 3.36.3.15 getPortRdy | 78 |

| | |
|--|----|
| 3.36.3.16 getPortRin | 78 |
| 3.36.3.17 getPortRst | 78 |
| 3.36.3.18 getQTDE_PORTS | 78 |
| 3.36.3.19 getQtdeSave | 78 |
| 3.36.3.20 getTamVectorOut | 78 |
| 3.36.3.21 getTOT_PORTS | 78 |
| 3.36.3.22 getValueLabel | 78 |
| 3.36.3.23 getVector | 78 |
| 3.36.3.24 getVector_000 | 79 |
| 3.36.3.25 getVector_111 | 79 |
| 3.36.3.26 getVector_UUU | 79 |
| 3.36.3.27 getVector_XXX | 79 |
| 3.36.3.28 getVector_ZZZ | 79 |
| 3.36.3.29 getVectorOut | 79 |
| 3.36.3.30 getVectorOutputPort | 79 |
| 3.36.3.31 initialize | 79 |
| 3.36.3.32 isDone | 80 |
| 3.36.3.33 isEnableAnimationFlag | 80 |
| 3.36.3.34 needsDynamicSymbol | 80 |
| 3.36.3.35 setCompName | 80 |
| 3.36.3.36 setComponentType | 80 |
| 3.36.3.37 setDefaultdelay | 80 |
| 3.36.3.38 setDelay | 80 |
| 3.36.3.39 setDelay | 81 |
| 3.36.3.40 setDone | 81 |
| 3.36.3.41 setEnableAnimationFlag | 81 |
| 3.36.3.42 setIdxDout | 81 |
| 3.36.3.43 setLabelFormatter | 81 |
| 3.36.3.44 setN_bits | 81 |
| 3.36.3.45 setPortClk | 81 |
| 3.36.3.46 setPortDin | 82 |
| 3.36.3.47 setPortEn | 82 |
| 3.36.3.48 setPortRdy | 82 |
| 3.36.3.49 setPortRin | 82 |
| 3.36.3.50 setPortRst | 82 |
| 3.36.3.51 setQtdeSave | 82 |
| 3.36.3.52 setTamVectorOut | 82 |
| 3.36.3.53 setValueLabel | 82 |
| 3.36.3.54 setVector | 83 |
| 3.36.3.55 setVector | 83 |

| | |
|---|----|
| 3.36.3.56 setVector_000 | 83 |
| 3.36.3.57 setVector_111 | 83 |
| 3.36.3.58 setVector_UUU | 83 |
| 3.36.3.59 setVector_XXX | 83 |
| 3.36.3.60 setVector_ZZZ | 83 |
| 3.36.3.61 setVectorOut | 83 |
| 3.36.3.62 setVectorOutputPort | 84 |
| 3.36.3.63 updateSymbol | 84 |
| 3.36.3.64 wakeup | 84 |
| 3.36.3.65 write | 84 |
| 3.37 hebe.dataflow.sync.GenericUn Class Reference | 84 |
| 3.37.1 Detailed Description | 85 |
| 3.37.2 Constructor & Destructor Documentation | 86 |
| 3.37.2.1 GenericUn | 86 |
| 3.37.3 Member Function Documentation | 86 |
| 3.37.3.1 compute | 86 |
| 3.37.3.2 constructDynamicSymbol | 86 |
| 3.37.3.3 constructPorts | 86 |
| 3.37.3.4 evaluate | 86 |
| 3.37.3.5 getBackground | 86 |
| 3.37.3.6 getComponentType | 86 |
| 3.37.3.7 getLabelNome | 87 |
| 3.37.3.8 getPortClk | 87 |
| 3.37.3.9 getPortDin | 87 |
| 3.37.3.10 getPortDout | 87 |
| 3.37.3.11 getPortEn | 87 |
| 3.37.3.12 getPortRin | 87 |
| 3.37.3.13 getPortRout | 87 |
| 3.37.3.14 getPortRst | 87 |
| 3.37.3.15 getS | 87 |
| 3.37.3.16 getStringLabel | 88 |
| 3.37.3.17 initialize | 88 |
| 3.37.3.18 needsDynamicSymbol | 88 |
| 3.37.3.19 notCompute | 88 |
| 3.37.3.20 reseted | 88 |
| 3.37.3.21 setBackground | 88 |
| 3.37.3.22 setCompName | 88 |
| 3.37.3.23 setComponentType | 88 |
| 3.37.3.24 setLabelNome | 89 |
| 3.37.3.25 setPortClk | 89 |

| | | |
|-----------|---|----|
| 3.37.3.26 | setPortDin | 89 |
| 3.37.3.27 | setPortDout | 89 |
| 3.37.3.28 | setPortEn | 89 |
| 3.37.3.29 | setPortRin | 89 |
| 3.37.3.30 | setPortRout | 89 |
| 3.37.3.31 | setPortRst | 89 |
| 3.37.3.32 | setS | 90 |
| 3.37.3.33 | setString | 90 |
| 3.37.3.34 | setStringLabel | 90 |
| 3.37.3.35 | setSymbol | 90 |
| 3.37.3.36 | tickDown | 90 |
| 3.37.3.37 | tickUp | 90 |
| 3.37.3.38 | write | 90 |
| 3.38 | hebe.examples.dataflow_sync.GourandDataflowFPGA Class Reference | 91 |
| 3.38.1 | Detailed Description | 91 |
| 3.39 | hebe.examples.dataflow_sync.GourandDataflowSimulation Class Reference | 91 |
| 3.39.1 | Detailed Description | 91 |
| 3.40 | hebe.dataflow.sync.Histogram Class Reference | 92 |
| 3.40.1 | Detailed Description | 92 |
| 3.40.2 | Constructor & Destructor Documentation | 92 |
| 3.40.2.1 | Histogram | 92 |
| 3.40.3 | Member Function Documentation | 92 |
| 3.40.3.1 | compute | 92 |
| 3.40.3.2 | evaluate | 93 |
| 3.40.3.3 | getCounter | 93 |
| 3.40.3.4 | getDecr | 93 |
| 3.40.3.5 | getHistogram | 93 |
| 3.40.3.6 | getNUMBITS | 93 |
| 3.40.3.7 | reset | 93 |
| 3.40.3.8 | setCounter | 93 |
| 3.40.3.9 | setDecr | 94 |
| 3.40.3.10 | setHistogram | 94 |
| 3.41 | hebe.examples.dataflow_sync.HistogramDataflowFpgaSimulation Class Reference | 94 |
| 3.41.1 | Detailed Description | 94 |
| 3.42 | hebe.examples.dataflow_sync.HistogramDataflowSimulation Class Reference | 94 |
| 3.42.1 | Detailed Description | 95 |
| 3.43 | hebe.dataflow.sync.In1 Class Reference | 95 |
| 3.43.1 | Detailed Description | 95 |
| 3.43.2 | Constructor & Destructor Documentation | 96 |
| 3.43.2.1 | In1 | 96 |

| | | |
|----------|--|-----|
| 3.44 | hebe.dataflow.sync.In16 Class Reference | 96 |
| 3.44.1 | Detailed Description | 96 |
| 3.44.2 | Constructor & Destructor Documentation | 96 |
| 3.44.2.1 | In16 | 96 |
| 3.45 | hebe.dataflow.sync.In2 Class Reference | 97 |
| 3.45.1 | Detailed Description | 97 |
| 3.45.2 | Constructor & Destructor Documentation | 97 |
| 3.45.2.1 | In2 | 97 |
| 3.46 | hebe.dataflow.sync.In32 Class Reference | 97 |
| 3.46.1 | Detailed Description | 98 |
| 3.46.2 | Constructor & Destructor Documentation | 98 |
| 3.46.2.1 | In32 | 98 |
| 3.47 | hebe.dataflow.sync.In4 Class Reference | 98 |
| 3.47.1 | Detailed Description | 98 |
| 3.47.2 | Constructor & Destructor Documentation | 99 |
| 3.47.2.1 | In4 | 99 |
| 3.48 | hebe.dataflow.sync.In8 Class Reference | 99 |
| 3.48.1 | Detailed Description | 99 |
| 3.48.2 | Constructor & Destructor Documentation | 100 |
| 3.48.2.1 | In8 | 100 |
| 3.49 | hebe.dataflow.sync.Max Class Reference | 100 |
| 3.49.1 | Detailed Description | 100 |
| 3.49.2 | Constructor & Destructor Documentation | 100 |
| 3.49.2.1 | Max | 100 |
| 3.49.3 | Member Function Documentation | 100 |
| 3.49.3.1 | compute | 101 |
| 3.50 | hebe.dataflow.sync.Merge Class Reference | 102 |
| 3.50.1 | Detailed Description | 102 |
| 3.50.2 | Constructor & Destructor Documentation | 102 |
| 3.50.2.1 | Merge | 102 |
| 3.50.3 | Member Function Documentation | 103 |
| 3.50.3.1 | evaluate | 103 |
| 3.51 | hebe.dataflow.sync.Min Class Reference | 103 |
| 3.51.1 | Detailed Description | 103 |
| 3.51.2 | Constructor & Destructor Documentation | 103 |
| 3.51.2.1 | Min | 103 |
| 3.51.3 | Member Function Documentation | 104 |
| 3.51.3.1 | compute | 104 |
| 3.52 | hebe.dataflow.sync.Mod Class Reference | 104 |
| 3.52.1 | Detailed Description | 104 |

| | | |
|----------|---|-----|
| 3.52.2 | Constructor & Destructor Documentation | 105 |
| 3.52.2.1 | Mod | 105 |
| 3.52.3 | Member Function Documentation | 105 |
| 3.52.3.1 | compute | 105 |
| 3.53 | hebe.dataflow.sync.ModI Class Reference | 105 |
| 3.53.1 | Detailed Description | 105 |
| 3.53.2 | Constructor & Destructor Documentation | 106 |
| 3.53.2.1 | ModI | 106 |
| 3.53.3 | Member Function Documentation | 106 |
| 3.53.3.1 | compute | 106 |
| 3.54 | hebe.dataflow.sync.Mul Class Reference | 106 |
| 3.54.1 | Detailed Description | 106 |
| 3.54.2 | Constructor & Destructor Documentation | 107 |
| 3.54.2.1 | Mul | 107 |
| 3.54.3 | Member Function Documentation | 107 |
| 3.54.3.1 | compute | 107 |
| 3.55 | hebe.dataflow.sync.Mull Class Reference | 107 |
| 3.55.1 | Detailed Description | 108 |
| 3.55.2 | Constructor & Destructor Documentation | 108 |
| 3.55.2.1 | Mull | 108 |
| 3.55.3 | Member Function Documentation | 108 |
| 3.55.3.1 | compute | 108 |
| 3.56 | hebe.dataflow.sync.Not Class Reference | 108 |
| 3.56.1 | Detailed Description | 109 |
| 3.56.2 | Constructor & Destructor Documentation | 109 |
| 3.56.2.1 | Not | 109 |
| 3.56.3 | Member Function Documentation | 109 |
| 3.56.3.1 | compute | 109 |
| 3.57 | hebe.dataflow.sync.Or Class Reference | 109 |
| 3.57.1 | Detailed Description | 110 |
| 3.57.2 | Constructor & Destructor Documentation | 110 |
| 3.57.2.1 | Or | 110 |
| 3.57.3 | Member Function Documentation | 110 |
| 3.57.3.1 | compute | 110 |
| 3.58 | hebe.dataflow.sync.OrI Class Reference | 110 |
| 3.58.1 | Detailed Description | 111 |
| 3.58.2 | Constructor & Destructor Documentation | 111 |
| 3.58.2.1 | OrI | 111 |
| 3.58.3 | Member Function Documentation | 111 |
| 3.58.3.1 | compute | 111 |

| | | |
|----------|--|-----|
| 3.59 | hebe.dataflow.sync.Out1 Class Reference | 112 |
| 3.59.1 | Detailed Description | 112 |
| 3.59.2 | Constructor & Destructor Documentation | 112 |
| 3.59.2.1 | Out1 | 112 |
| 3.60 | hebe.dataflow.sync.Out16 Class Reference | 112 |
| 3.60.1 | Detailed Description | 113 |
| 3.60.2 | Constructor & Destructor Documentation | 113 |
| 3.60.2.1 | Out16 | 113 |
| 3.61 | hebe.dataflow.sync.Out2 Class Reference | 113 |
| 3.61.1 | Detailed Description | 113 |
| 3.61.2 | Constructor & Destructor Documentation | 114 |
| 3.61.2.1 | Out2 | 114 |
| 3.62 | hebe.dataflow.sync.Out32 Class Reference | 114 |
| 3.62.1 | Detailed Description | 114 |
| 3.62.2 | Constructor & Destructor Documentation | 115 |
| 3.62.2.1 | Out32 | 115 |
| 3.63 | hebe.dataflow.sync.Out4 Class Reference | 115 |
| 3.63.1 | Detailed Description | 115 |
| 3.63.2 | Constructor & Destructor Documentation | 115 |
| 3.63.2.1 | Out4 | 115 |
| 3.64 | hebe.dataflow.sync.Out8 Class Reference | 116 |
| 3.64.1 | Detailed Description | 116 |
| 3.64.2 | Constructor & Destructor Documentation | 116 |
| 3.64.2.1 | Out8 | 116 |
| 3.65 | hebe.examples.dataflow_sync.PaethDataflowFPGA Class Reference | 116 |
| 3.65.1 | Detailed Description | 116 |
| 3.66 | hebe.examples.dataflow_sync.PaethDataflowSimulation Class Reference | 117 |
| 3.66.1 | Detailed Description | 117 |
| 3.67 | hebe.examples.dataflow_sync.Reduce32DataflowFpga Class Reference | 117 |
| 3.67.1 | Detailed Description | 117 |
| 3.68 | hebe.examples.dataflow_sync.Reduce32DataflowSimulation Class Reference | 118 |
| 3.68.1 | Detailed Description | 118 |
| 3.69 | hebe.dataflow.sync.Register Class Reference | 118 |
| 3.69.1 | Detailed Description | 119 |
| 3.69.2 | Constructor & Destructor Documentation | 119 |
| 3.69.2.1 | Register | 119 |
| 3.70 | hebe.dataflow.sync.Shl Class Reference | 119 |
| 3.70.1 | Detailed Description | 119 |
| 3.70.2 | Constructor & Destructor Documentation | 120 |
| 3.70.2.1 | Shl | 120 |

| | | |
|----------|---|-----|
| 3.70.3 | Member Function Documentation | 120 |
| 3.70.3.1 | compute | 120 |
| 3.71 | hebe.dataflow.sync.Shll Class Reference | 120 |
| 3.71.1 | Detailed Description | 121 |
| 3.71.2 | Constructor & Destructor Documentation | 121 |
| 3.71.2.1 | Shll | 121 |
| 3.71.3 | Member Function Documentation | 121 |
| 3.71.3.1 | compute | 121 |
| 3.72 | hebe.dataflow.sync.Shr Class Reference | 121 |
| 3.72.1 | Detailed Description | 122 |
| 3.72.2 | Constructor & Destructor Documentation | 122 |
| 3.72.2.1 | Shr | 122 |
| 3.72.3 | Member Function Documentation | 122 |
| 3.72.3.1 | compute | 122 |
| 3.73 | hebe.dataflow.sync.Shrl Class Reference | 122 |
| 3.73.1 | Detailed Description | 123 |
| 3.73.2 | Constructor & Destructor Documentation | 123 |
| 3.73.2.1 | Shrl | 123 |
| 3.73.3 | Member Function Documentation | 123 |
| 3.73.3.1 | compute | 123 |
| 3.74 | hebe.dataflow.sync.Slt Class Reference | 123 |
| 3.74.1 | Detailed Description | 124 |
| 3.74.2 | Constructor & Destructor Documentation | 124 |
| 3.74.2.1 | Slt | 124 |
| 3.74.3 | Member Function Documentation | 124 |
| 3.74.3.1 | compute | 124 |
| 3.75 | hebe.dataflow.sync.Sltl Class Reference | 124 |
| 3.75.1 | Detailed Description | 125 |
| 3.75.2 | Constructor & Destructor Documentation | 125 |
| 3.75.2.1 | Sltl | 125 |
| 3.75.3 | Member Function Documentation | 125 |
| 3.75.3.1 | compute | 125 |
| 3.76 | hebe.dataflow.sync.Sub Class Reference | 126 |
| 3.76.1 | Detailed Description | 126 |
| 3.76.2 | Constructor & Destructor Documentation | 126 |
| 3.76.2.1 | Sub | 126 |
| 3.76.3 | Member Function Documentation | 126 |
| 3.76.3.1 | compute | 126 |
| 3.77 | hebe.dataflow.sync.Subl Class Reference | 127 |
| 3.77.1 | Detailed Description | 127 |

| | | |
|--------------|--|------------|
| 3.77.2 | Constructor & Destructor Documentation | 127 |
| 3.77.2.1 | Subl | 127 |
| 3.77.3 | Member Function Documentation | 127 |
| 3.77.3.1 | compute | 127 |
| Index | | 129 |

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

| | |
|--|-----|
| hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulation | 11 |
| hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulationWithGeneratedHds | 12 |
| hebe.examples.dataflow_sync.AddiDataflowFpgaSimulation | 13 |
| hebe.dataflow.AFU | 14 |
| hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulation | 21 |
| hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulationWithGeneratedHds | 22 |
| hebe.util.ConfReader | 22 |
| hebe.dataflow.DataflowSyncSimulBase | 23 |
| hebe.examples.dataflow_sync.Fir16DataflowFPGA | 27 |
| hebe.examples.dataflow_sync.Fir16DataflowSimulation | 28 |
| hebe.examples.dataflow_sync.Fir4DataflowHadesSimulation | 28 |
| hebe.examples.dataflow_sync.Fir4DataflowHadesSimulationWithGeneratedHds | 29 |
| hebe.examples.dataflow_sync.Fir8DataflowFPGA | 29 |
| hebe.examples.dataflow_sync.Fir8DataflowSimulation | 30 |
| GenericRtlibObject | |
| hebe.dataflow.sync.GenericBin | 32 |
| hebe.dataflow.sync.Add | 10 |
| hebe.dataflow.sync.And | 14 |
| hebe.dataflow.sync.Div | 25 |
| hebe.dataflow.sync.Max | 100 |
| hebe.dataflow.sync.Merge | 102 |
| hebe.dataflow.sync.Min | 103 |
| hebe.dataflow.sync.Mod | 104 |
| hebe.dataflow.sync.Mul | 106 |
| hebe.dataflow.sync.Or | 109 |
| hebe.dataflow.sync.Shl | 119 |
| hebe.dataflow.sync.Shr | 121 |
| hebe.dataflow.sync.Slt | 123 |
| hebe.dataflow.sync.Sub | 126 |
| hebe.dataflow.sync.GenericBranch | 39 |
| hebe.dataflow.sync.Beq | 16 |
| hebe.dataflow.sync.Bne | 19 |
| hebe.dataflow.sync.GenericBranchl | 45 |
| hebe.dataflow.sync.Beql | 18 |
| hebe.dataflow.sync.Bnel | 20 |
| hebe.dataflow.sync.Genericcl | 54 |
| hebe.dataflow.sync.Addl | 12 |
| hebe.dataflow.sync.Andl | 15 |

| | |
|---|-----|
| hebe.dataflow.sync.Divl | 26 |
| hebe.dataflow.sync.GenericAcc | 30 |
| hebe.dataflow.sync.AccAdd | 6 |
| hebe.dataflow.sync.AccMax | 7 |
| hebe.dataflow.sync.AccMin | 8 |
| hebe.dataflow.sync.AccMul | 9 |
| hebe.dataflow.sync.Histogram | 92 |
| hebe.dataflow.sync.Modl | 105 |
| hebe.dataflow.sync.Mull | 107 |
| hebe.dataflow.sync.Orl | 110 |
| hebe.dataflow.sync.Shll | 120 |
| hebe.dataflow.sync.Shrl | 122 |
| hebe.dataflow.sync.Sltl | 124 |
| hebe.dataflow.sync.Subl | 127 |
| hebe.dataflow.sync.GenericUn | 84 |
| hebe.dataflow.sync.Abs | 5 |
| hebe.dataflow.sync.Not | 108 |
| hebe.dataflow.sync.Register | 118 |
| hebe.examples.dataflow_sync.GourandDataflowFPGA | 91 |
| hebe.examples.dataflow_sync.GourandDataflowSimulation | 91 |
| hebe.examples.dataflow_sync.HistogramDataflowFpgaSimulation | 94 |
| hebe.examples.dataflow_sync.HistogramDataflowSimulation | 94 |
| hebe.examples.dataflow_sync.PaethDataflowFPGA | 116 |
| hebe.examples.dataflow_sync.PaethDataflowSimulation | 117 |
| hebe.examples.dataflow_sync.Reduce32DataflowFpga | 117 |
| hebe.examples.dataflow_sync.Reduce32DataflowSimulation | 118 |
| Serializable | |
| hebe.dataflow.sync.GenericIn | 64 |
| hebe.dataflow.sync.In1 | 95 |
| hebe.dataflow.sync.In16 | 96 |
| hebe.dataflow.sync.In2 | 97 |
| hebe.dataflow.sync.In32 | 97 |
| hebe.dataflow.sync.In4 | 98 |
| hebe.dataflow.sync.In8 | 99 |
| hebe.dataflow.sync.GenericOut | 74 |
| hebe.dataflow.sync.Out1 | 112 |
| hebe.dataflow.sync.Out16 | 112 |
| hebe.dataflow.sync.Out2 | 113 |
| hebe.dataflow.sync.Out32 | 114 |
| hebe.dataflow.sync.Out4 | 115 |
| hebe.dataflow.sync.Out8 | 116 |
| SimObject | |
| hebe.dataflow.sync.GenericIn | 64 |
| hebe.dataflow.sync.GenericOut | 74 |
| Simulatable | |
| hebe.dataflow.sync.GenericIn | 64 |
| hebe.dataflow.sync.GenericOut | 74 |
| Wakeable | |
| hebe.dataflow.sync.GenericIn | 64 |
| hebe.dataflow.sync.GenericOut | 74 |

Chapter 2

Class Index

2.1 Class List

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

| | |
|--|----|
| hebe.dataflow.sync.Abs | 5 |
| hebe.dataflow.sync.AccAdd | 6 |
| hebe.dataflow.sync.AccMax | 7 |
| hebe.dataflow.sync.AccMin | 8 |
| hebe.dataflow.sync.AccMul | 9 |
| hebe.dataflow.sync.Add | 10 |
| hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulation | 11 |
| hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulationWithGeneratedHds | 12 |
| hebe.dataflow.sync.AddI | 12 |
| hebe.examples.dataflow_sync.AddIDataflowFpgaSimulation | 13 |
| hebe.dataflow.AFU | 14 |
| hebe.dataflow.sync.And | 14 |
| hebe.dataflow.sync.AndI | 15 |
| hebe.dataflow.sync.Beq | 16 |
| hebe.dataflow.sync.BeqI | 18 |
| hebe.dataflow.sync.Bne | 19 |
| hebe.dataflow.sync.BneI | 20 |
| hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulation | 21 |
| hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulationWithGeneratedHds | 22 |
| hebe.util.ConfReader | 22 |
| hebe.dataflow.DataflowSyncSimulBase | 23 |
| hebe.dataflow.sync.Div | 25 |
| hebe.dataflow.sync.DivI | 26 |
| hebe.examples.dataflow_sync.Fir16DataflowFPGA | 27 |
| hebe.examples.dataflow_sync.Fir16DataflowSimulation | 28 |
| hebe.examples.dataflow_sync.Fir4DataflowHadesSimulation | 28 |
| hebe.examples.dataflow_sync.Fir4DataflowHadesSimulationWithGeneratedHds | 29 |
| hebe.examples.dataflow_sync.Fir8DataflowFPGA | 29 |
| hebe.examples.dataflow_sync.Fir8DataflowSimulation | 30 |
| hebe.dataflow.sync.GenericAcc | 30 |
| hebe.dataflow.sync.GenericBin | 32 |
| hebe.dataflow.sync.GenericBranch | 39 |
| hebe.dataflow.sync.GenericBranchI | 45 |
| hebe.dataflow.sync.GenericI | 54 |
| hebe.dataflow.sync.GenericIn | 64 |
| hebe.dataflow.sync.GenericOut | 74 |
| hebe.dataflow.sync.GenericUn | 84 |
| hebe.examples.dataflow_sync.GourandDataflowFPGA | 91 |

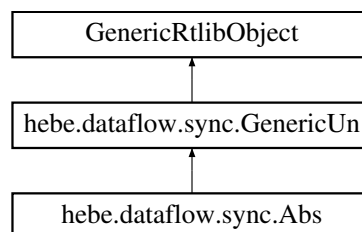
| | |
|---|-----|
| hebe.examples.dataflow_sync.GourandDataflowSimulation | 91 |
| hebe.dataflow.sync.Histogram | 92 |
| hebe.examples.dataflow_sync.HistogramDataflowFpgaSimulation | 94 |
| hebe.examples.dataflow_sync.HistogramDataflowSimulation | 94 |
| hebe.dataflow.sync.In1 | 95 |
| hebe.dataflow.sync.In16 | 96 |
| hebe.dataflow.sync.In2 | 97 |
| hebe.dataflow.sync.In32 | 97 |
| hebe.dataflow.sync.In4 | 98 |
| hebe.dataflow.sync.In8 | 99 |
| hebe.dataflow.sync.Max | 100 |
| hebe.dataflow.sync.Merge | 102 |
| hebe.dataflow.sync.Min | 103 |
| hebe.dataflow.sync.Mod | 104 |
| hebe.dataflow.sync.Modl | 105 |
| hebe.dataflow.sync.Mul | 106 |
| hebe.dataflow.sync.Mull | 107 |
| hebe.dataflow.sync.Not | 108 |
| hebe.dataflow.sync.Or | 109 |
| hebe.dataflow.sync.Orl | 110 |
| hebe.dataflow.sync.Out1 | 112 |
| hebe.dataflow.sync.Out16 | 112 |
| hebe.dataflow.sync.Out2 | 113 |
| hebe.dataflow.sync.Out32 | 114 |
| hebe.dataflow.sync.Out4 | 115 |
| hebe.dataflow.sync.Out8 | 116 |
| hebe.examples.dataflow_sync.PaethDataflowFPGA | 116 |
| hebe.examples.dataflow_sync.PaethDataflowSimulation | 117 |
| hebe.examples.dataflow_sync.Reduce32DataflowFpga | 117 |
| hebe.examples.dataflow_sync.Reduce32DataflowSimulation | 118 |
| hebe.dataflow.sync.Register | 118 |
| hebe.dataflow.sync.Shl | 119 |
| hebe.dataflow.sync.Shll | 120 |
| hebe.dataflow.sync.Shr | 121 |
| hebe.dataflow.sync.Shrl | 122 |
| hebe.dataflow.sync.Slt | 123 |
| hebe.dataflow.sync.Sltl | 124 |
| hebe.dataflow.sync.Sub | 126 |
| hebe.dataflow.sync.Subl | 127 |

Chapter 3

Class Documentation

3.1 hebe.dataflow.sync.Abs Class Reference

Inheritance diagram for hebe.dataflow.sync.Abs:



Public Member Functions

- [Abs](#) ()
- int [compute](#) (int data)

3.1.1 Detailed Description

[Abs](#) component for the UFV synchronous data flow simulator.

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 hebe.dataflow.sync.Abs.Abs ()

Object Constructor.

3.1.3 Member Function Documentation

3.1.3.1 `int hebe.dataflow.sync.Abs.compute (int data)`

Method responsible for the component computation.

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

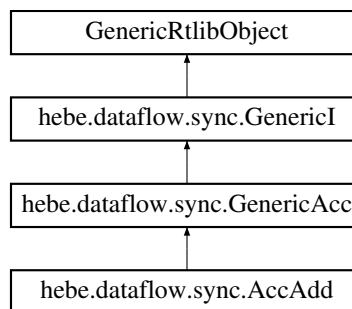
- Returns the result of the computation. In this case, returns the absolute value of the parameter.

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

- `hebe/dataflow/sync/Abs.java`

3.2 `hebe.dataflow.sync.AccAdd` Class Reference

Inheritance diagram for `hebe.dataflow.sync.AccAdd`:



Public Member Functions

- [AccAdd \(\)](#)

Additional Inherited Members

3.2.1 Detailed Description

[AccAdd](#) component for the UFV synchronous data flow simulator.

The component implements an adder accumulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.2.2 Constructor & Destructor Documentation

3.2.2.1 hebe.dataflow.sync.AccAdd.AccAdd ()

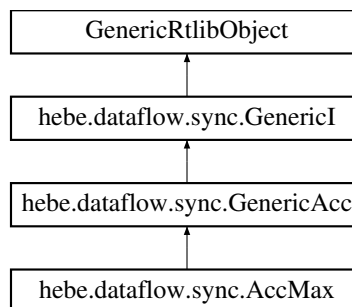
Object Constructor.

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

- hebe/dataflow/sync/AccAdd.java

3.3 hebe.dataflow.sync.AccMax Class Reference

Inheritance diagram for hebe.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 UFV synchronous data flow simulator.

The component implements a store for the highest input value.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.3.2 Constructor & Destructor Documentation

3.3.2.1 hebe.dataflow.sync.AccMax.AccMax ()

Object Constructor.

3.3.3 Member Function Documentation

3.3.3.1 void hebe.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

3.3.3.2 void hebe.dataflow.sync.AccMax.reset ()

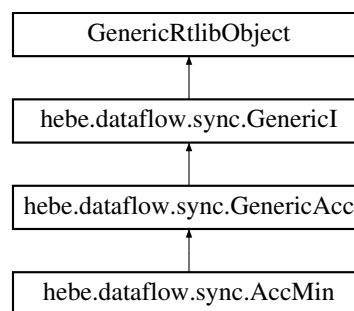
Method responsible for actions required when "Reset" occurs.

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

- hebe/dataflow/sync/AccMax.java

3.4 hebe.dataflow.sync.AccMin Class Reference

Inheritance diagram for hebe.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 UFV synchronous data flow simulator.

The component implements a store for the lowest input value.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.4.2 Constructor & Destructor Documentation

3.4.2.1 hebe.dataflow.sync.AccMin.AccMin ()

Object Constructor.

3.4.3 Member Function Documentation

3.4.3.1 void hebe.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

3.4.3.2 void hebe.dataflow.sync.AccMin.reset ()

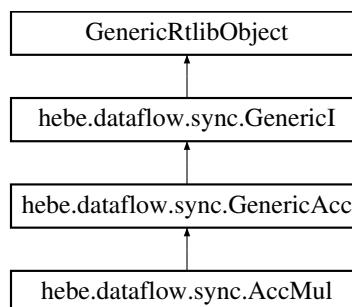
Method responsible for actions required when "Reset" occurs.

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

- hebe/dataflow/sync/AccMin.java

3.5 hebe.dataflow.sync.AccMul Class Reference

Inheritance diagram for hebe.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 UFV synchronous data flow simulator.

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 `hebe.dataflow.sync.AccMul.AccMul ()`

Object Constructor.

3.5.3 Member Function Documentation

3.5.3.1 `void hebe.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 hebe.dataflow.sync.AccMul.reset ()`

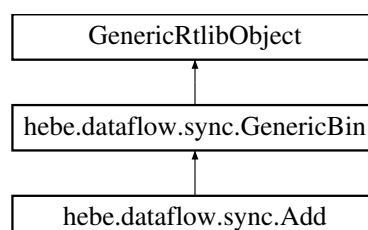
Method responsible for actions required when "Reset" occurs.

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

- `hebe/dataflow/sync/AccMul.java`

3.6 `hebe.dataflow.sync.Add` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Add`:



Public Member Functions

- [Add \(\)](#)
- `int compute (int data1, int data2)`

3.6.1 Detailed Description

Add component for the UFV synchronous data flow simulator.

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 hebe.dataflow.sync.Add.Add ()

Object Constructor.

3.6.3 Member Function Documentation

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

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

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - 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:

- hebe/dataflow/sync/Add.java

3.7 hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.7.1 Detailed Description

ADD_AB data flow example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/AddAbHelloWorldDataflowHadesSimulation.java

3.8 hebe.examples.dataflow_sync.AddAbHelloWorldDataflowHadesSimulationWith-GeneratedHds Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.8.1 Detailed Description

AddAb data flow (Automatically generated) example in the simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

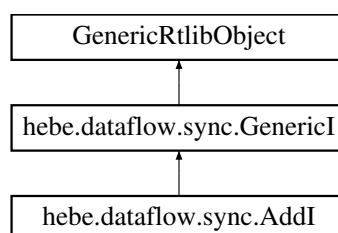
* 1.0

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

- hebe/examples/dataflow_sync/AddAbHelloWorldDataflowHadesSimulationWithGeneratedHds.java

3.9 hebe.dataflow.sync.AddI Class Reference

Inheritance diagram for hebe.dataflow.sync.AddI:



Public Member Functions

- [Addl](#) ()
- int [compute](#) (int data)

3.9.1 Detailed Description

[Addl](#) component for the UFV synchronous data flow simulator.

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

3.9.2 Constructor & Destructor Documentation

3.9.2.1 hebe.dataflow.sync.Addl.Addl ()

Object Constructor.

3.9.3 Member Function Documentation

3.9.3.1 int hebe.dataflow.sync.Addl.compute (int *data*)

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

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - 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:

- hebe/dataflow/sync/Addl.java

3.10 hebe.examples.dataflow_sync.AddiDataflowFpgaSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.10.1 Detailed Description

ADDI data flow example in the FPGA Board.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronimopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

* 1.0

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

- [hebe/examples/dataflow_sync/AddiDataflowFpgaSimulation.java](#)

3.11 hebe.dataflow.AFU Class Reference

Static Public Member Functions

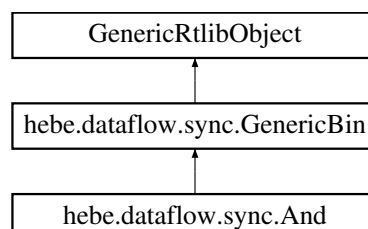
- static native int **process** (IntBuffer buffer_conf, int size_buffer_conf, ByteBuffer buffer_in, int size_buffer_in, ByteBuffer buffer_out, int size_buffer_out, int size_of_data)
- static native int **process** (IntBuffer buffer_conf, int size_buffer_conf, ShortBuffer buffer_in, int size_buffer_in, ShortBuffer buffer_out, int size_buffer_out, int size_of_data)
- static native int **process** (IntBuffer buffer_conf, int size_buffer_conf, IntBuffer buffer_in, int size_buffer_in, IntBuffer buffer_out, int size_buffer_out, int size_of_data)
- static native int **process** (IntBuffer buffer_conf, int size_buffer_conf, LongBuffer buffer_in, int size_buffer_in, LongBuffer buffer_out, int size_buffer_out, int size_of_data)
- static short[] **run** (int[] conf, short[] data_in, int size_data_out)
- static void **main** (String args[])

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

- [hebe/dataflow/AFU.java](#)

3.12 hebe.dataflow.sync.And Class Reference

Inheritance diagram for hebe.dataflow.sync.And:



Public Member Functions

- [And](#) ()
- int [compute](#) (int data1, int data2)

3.12.1 Detailed Description

[And](#) component for the UFV synchronous data flow simulator.

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

3.12.2 Constructor & Destructor Documentation

3.12.2.1 hebe.dataflow.sync.And.And ()

Object Constructor.

3.12.3 Member Function Documentation

3.12.3.1 int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

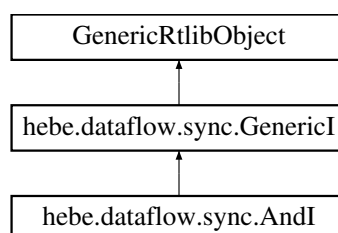
- Returns the result of the computation. In this case the result of the logical operation "And" between the parameters.

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

- hebe/dataflow/sync/And.java

3.13 hebe.dataflow.sync.AndI Class Reference

Inheritance diagram for hebe.dataflow.sync.AndI:



Public Member Functions

- [AndI](#) ()
- `int compute` (int data)

3.13.1 Detailed Description

[AndI](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.13.2 Constructor & Destructor Documentation

3.13.2.1 `hebe.dataflow.sync.AndI.AndI ()`

Object Constructor.

3.13.3 Member Function Documentation

3.13.3.1 `int hebe.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

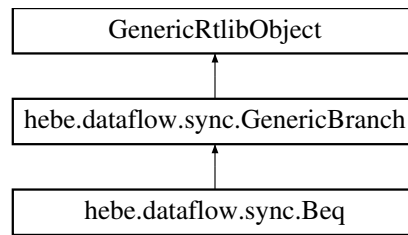
- Returns the result of the computation. In this case the result of the logical operation "AND" between the parameter and the id.

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

- `hebe/dataflow/sync/AndI.java`

3.14 `hebe.dataflow.sync.Beq` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Beq`:



Public Member Functions

- [Beq](#) ()
- [int compute](#) (int data1, int data2)

3.14.1 Detailed Description

[Beq](#) component for the synchronous data flow simulator of the UFV.

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

3.14.2 Constructor & Destructor Documentation

3.14.2.1 `hebe.dataflow.sync.Beq.Beq ()`

Object Constructor.

3.14.3 Member Function Documentation

3.14.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

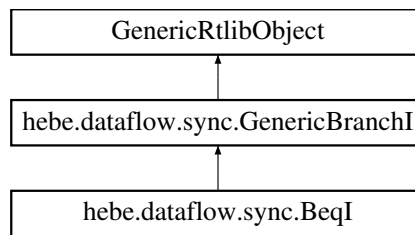
- Returns the result of the computation. In this case "1" if the parameters are equal or "0" if they are different.

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

- `hebe/dataflow/sync/Beq.java`

3.15 hebe.dataflow.sync.BeqI Class Reference

Inheritance diagram for hebe.dataflow.sync.BeqI:



Public Member Functions

- [BeqI](#) ()
- int [compute](#) (int data)

3.15.1 Detailed Description

[BeqI](#) component for the synchronous data flow simulator of the UFV.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.15.2 Constructor & Destructor Documentation

3.15.2.1 hebe.dataflow.sync.BeqI.BeqI ()

Object Constructor.

3.15.3 Member Function Documentation

3.15.3.1 int hebe.dataflow.sync.BeqI.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

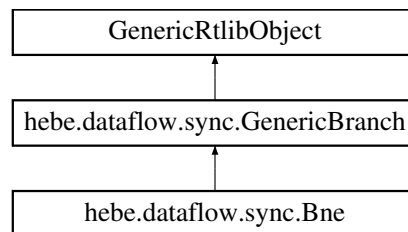
- Returns the result of the computation. In this case "1" if the parameter is equal to the constraint or "0" if they are different.

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

- hebe/dataflow/sync/Beql.java

3.16 hebe.dataflow.sync.Bne Class Reference

Inheritance diagram for hebe.dataflow.sync.Bne:



Public Member Functions

- [Bne](#) ()
- int [compute](#) (int data1, int data2)

3.16.1 Detailed Description

[Bne](#) component for the synchronous data flow simulator of the UFV.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.16.2 Constructor & Destructor Documentation

3.16.2.1 hebe.dataflow.sync.Bne.Bne ()

Object Constructor.

3.16.3 Member Function Documentation

3.16.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

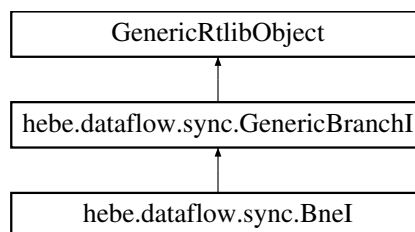
- Returns the result of the computation. In this case "1" if the parameters are different or "0" if they are equal.

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

- `hebe/dataflow/sync/Bne.java`

3.17 `hebe.dataflow.sync.BneI` Class Reference

Inheritance diagram for `hebe.dataflow.sync.BneI`:



Public Member Functions

- [BneI \(\)](#)
- `int compute (int data)`

3.17.1 Detailed Description

BEQI component for the synchronous data flow simulator of the UFV.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.17.2 Constructor & Destructor Documentation

3.17.2.1 hebe.dataflow.sync.Bnel.Bnel ()

Object Constructor.

3.17.3 Member Function Documentation

3.17.3.1 int hebe.dataflow.sync.Bnel.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - 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:

- hebe/dataflow/sync/Bnel.java

3.18 hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.18.1 Detailed Description

Branch_Test data flow example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/BranchTestDataflowHadesSimulation.java

3.19 hebe.examples.dataflow_sync.BranchTestDataflowHadesSimulationWithGeneratedHds Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.19.1 Detailed Description

Branch_Test data flow (Automatically generated) example in the simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronimopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

* 1.0

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

- hebe/examples/dataflow_sync/BranchTestDataflowHadesSimulationWithGeneratedHds.java

3.20 hebe.util.ConfReader Class Reference

Public Member Functions

- int[] [ReadConfig](#) (File file)

3.20.1 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

3.20.2 Member Function Documentation

3.20.2.1 int [] hebe.util.ConfReader.ReadConfig (File file)

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

Parameters

| | |
|-------------|----------------|
| <i>file</i> | - 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:

- hebe/util/ConfReader.java

3.21 hebe.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)

3.21.1 Detailed Description

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

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.21.2 Member Function Documentation

3.21.2.1 `int [] hebe.dataflow.DataflowSyncSimulBase.execFpga (int[] rawData, String quartusStpPath, int outSize)`

Execution in FPGA Boards

Parameters

| | |
|-----------------------|--|
| <i>rawData</i> | - Vector of data to be processed |
| <i>quartusStpPath</i> | - Path to the quartus_stp application. |
| <i>outSize</i> | - Output vector size. |

Returns

- Returns a vector with the processing results.

3.21.2.2 `int [] hebe.dataflow.DataflowSyncSimulBase.execHades (int[] rawData, String designPath, int outSize)`

Method responsible for executing the algorithm in the simulator.

Parameters

| | |
|----------------------|----------------------------------|
| <i>data_in</i> | - Input vector to be processed.. |
| <i>size_data_in</i> | - Size of the input vector. |
| <i>size_data_out</i> | - Size of the output vector. |

Returns

- Returns a vector with the processing results. Execution in HADES Simulator

Parameters

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

Returns

- Returns a vector with the processing results.

3.21.2.3 `int [] hebe.dataflow.DataflowSyncSimulBase.startFpgaJtag (int[] conf, String quartusStpPath, int outSize)`

Method responsible for running the algorithm on the FPGA board.

Parameters

| | |
|-----------------------|---|
| <i>conf</i> | - Configuration vector and data to be executed. |
| <i>quartusStpPath</i> | - Path to the quartus_stp application. |
| <i>outSize</i> | - Output vector size. |

Returns

- Returns a vector with the processing results.

3.21.2.4 `int [] hebe.dataflow.DataflowSyncSimulBase.startFpgaJtag (String confPath, String quartusStpPath, String desiredReturn, int outSize)`

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

Parameters

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

Returns

- Returns a vector with the processing results.

3.21.2.5 `int [] hebe.dataflow.DataflowSyncSimulBase.startSimulation (int[] conf, String designPath, int outSize)`

Method responsible for executing the algorithm in the simulator.

Parameters

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

Returns

- Returns a vector with the processing results.

3.21.2.6 `int [] hebe.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

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

Returns

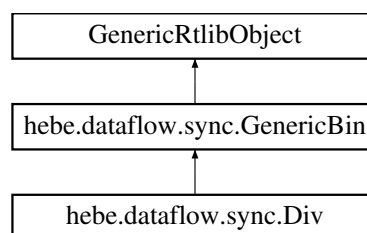
- Returns a vector with the processing results.

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

- hebe/dataflow/DataflowSyncSimulBase.java

3.22 hebe.dataflow.sync.Div Class Reference

Inheritance diagram for hebe.dataflow.sync.Div:



Public Member Functions

- [Div](#) ()
- `int compute (int data1, int data2)`

3.22.1 Detailed Description

[Div](#) component for the UFV synchronous data flow simulator.

The component is responsible for dividing the inputs.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicsa@gmail.com

Version

1.0

3.22.2 Constructor & Destructor Documentation**3.22.2.1 `hebe.dataflow.sync.Div.Div ()`**

Object Constructor.

3.22.3 Member Function Documentation**3.22.3.1 `int hebe.dataflow.sync.Div.compute (int data1, int data2)`**

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

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

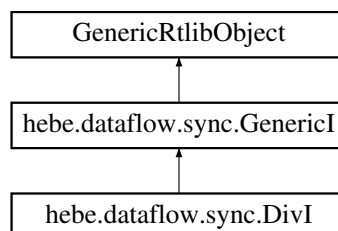
- Returns the result of the computation. In this case the value of the division of the parameters.

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

- `hebe/dataflow/sync/Div.java`

3.23 `hebe.dataflow.sync.DivI` Class Reference

Inheritance diagram for `hebe.dataflow.sync.DivI`:

**Public Member Functions**

- `DivI ()`
- `int compute (int data)`

3.23.1 Detailed Description

[Divl](#) component for the UFV synchronous data flow simulator.

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

3.23.2 Constructor & Destructor Documentation

3.23.2.1 hebe.dataflow.sync.Divl.Divl ()

Object Constructor.

3.23.3 Member Function Documentation

3.23.3.1 int hebe.dataflow.sync.Divl.compute (int *data*)

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

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - 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:

- hebe/dataflow/sync/Divl.java

3.24 hebe.examples.dataflow_sync.Fir16DataflowFPGA Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.24.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/Fir16DataflowFPGA.java

3.25 hebe.examples.dataflow_sync.Fir16DataflowSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.25.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/Fir16DataflowSimulation.java

3.26 hebe.examples.dataflow_sync.Fir4DataflowHadesSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.26.1 Detailed Description

FIR_4 data flow example in the simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/Fir4DataflowHadesSimulation.java

3.27 hebe.examples.dataflow_sync.Fir4DataflowHadesSimulationWithGeneratedHds Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.27.1 Detailed Description

FIR_4 data flow (Automatically generated) example in the simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

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

- hebe/examples/dataflow_sync/Fir4DataflowHadesSimulationWithGeneratedHds.java

3.28 hebe.examples.dataflow_sync.Fir8DataflowFPGA Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.28.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/Fir8DataflowFPGA.java

3.29 hebe.examples.dataflow_sync.Fir8DataflowSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.29.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronimopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

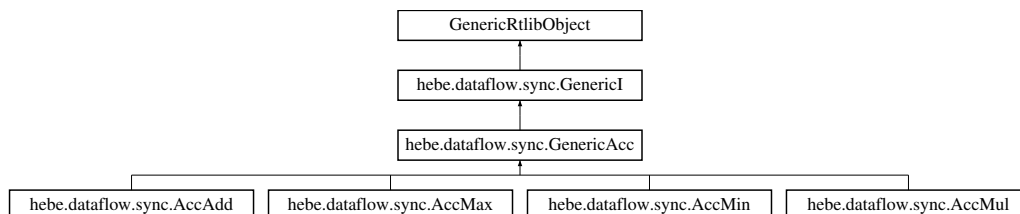
* 1.0

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

- hebe/examples/dataflow_sync/Fir8DataflowSimulation.java

3.30 hebe.dataflow.sync.GenericAcc Class Reference

Inheritance diagram for hebe.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)

3.30.1 Detailed Description

[GenericAcc](#) component for the UFV synchronous data flow simulator.

The component implements a generic accumulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.30.2 Constructor & Destructor Documentation

3.30.2.1 hebe.dataflow.sync.GenericAcc.GenericAcc ()

Object Constructor.

3.30.3 Member Function Documentation

3.30.3.1 void hebe.dataflow.sync.GenericAcc.accumulate (int *data*) [protected]

Method responsible for performing the accumulation or not.

Parameters

| | |
|-------------|---|
| <i>data</i> | - Value to be used for the computation. |
|-------------|---|

3.30.3.2 void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.30.3.3 int hebe.dataflow.sync.GenericAcc.getAcc ()

Returns

the acc

3.30.3.4 int hebe.dataflow.sync.GenericAcc.getCounter ()

Returns

the counter

3.30.3.5 void hebe.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.

3.30.3.6 void hebe.dataflow.sync.GenericAcc.setAcc (int *acc*)

Parameters

| | |
|------------|----------------|
| <i>acc</i> | the acc to set |
|------------|----------------|

3.30.3.7 void hebe.dataflow.sync.GenericAcc.setCounter (int *counter*)

Parameters

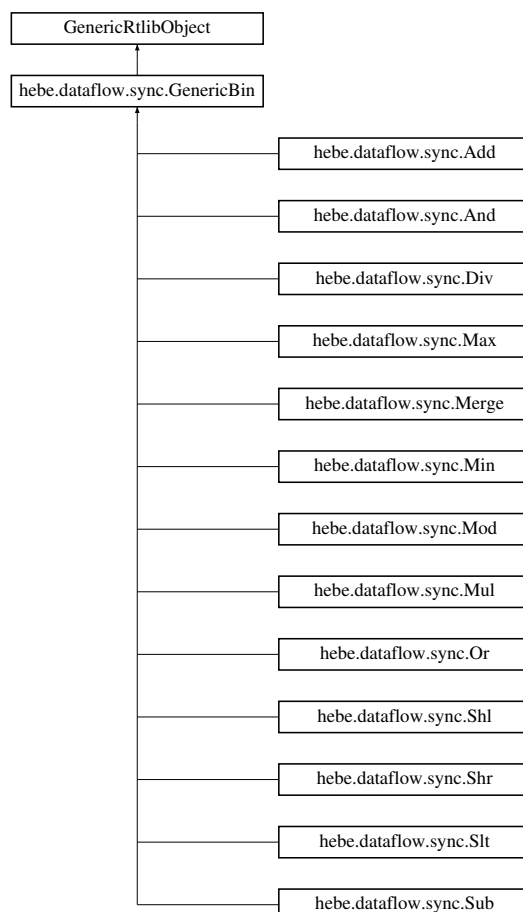
| | |
|----------------|--------------------|
| <i>counter</i> | the counter to set |
|----------------|--------------------|

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

- hebe/dataflow/sync/GenericAcc.java

3.31 hebe.dataflow.sync.GenericBin Class Reference

Inheritance diagram for hebe.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)

3.31.1 Detailed Description

[GenericBin](#) component for the UFV synchronous data flow simulator.

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

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.31.2 Constructor & Destructor Documentation

3.31.2.1 `hebe.dataflow.sync.GenericBin.GenericBin ()`

Object Constructor.

3.31.3 Member Function Documentation

3.31.3.1 `int hebe.dataflow.sync.GenericBin.compute (int data1, int data2)`

Method responsible for the computation of the output.

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 1. |

Returns

- Return of computation

3.31.3.2 `void hebe.dataflow.sync.GenericBin.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.31.3.3 `void hebe.dataflow.sync.GenericBin.constructPorts ()`

Method responsible for initializing the component input and output ports.

3.31.3.4 `void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.31.3.5 `String hebe.dataflow.sync.GenericBin.getComponentType ()`

Returns

the `componentType`

3.31.3.6 Label hebe.dataflow.sync.GenericBin.getLabelNome ()

Returns

the labelNome

3.31.3.7 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortClk ()

Returns

the portClk

3.31.3.8 PortStdLogicVector hebe.dataflow.sync.GenericBin.getPortDin1 ()

Returns

the portDin1

3.31.3.9 PortStdLogicVector hebe.dataflow.sync.GenericBin.getPortDin2 ()

Returns

the portDin2

3.31.3.10 PortStdLogicVector hebe.dataflow.sync.GenericBin.getPortDout ()

Returns

the portDout

3.31.3.11 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortEn ()

Returns

the portEn

3.31.3.12 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortRin1 ()

Returns

the portRin1

3.31.3.13 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortRin2 ()

Returns

the portRin2

3.31.3.14 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortRout ()

Returns

the portRout

3.31.3.15 PortStdLogic1164 hebe.dataflow.sync.GenericBin.getPortRst ()

Returns

the portRst

3.31.3.16 String hebe.dataflow.sync.GenericBin.getS ()

Returns

the s

3.31.3.17 Label hebe.dataflow.sync.GenericBin.getStringLabel ()

Returns

the stringLabel

3.31.3.18 boolean hebe.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.

3.31.3.19 boolean hebe.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.

3.31.3.20 void hebe.dataflow.sync.GenericBin.notCompute ()

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

3.31.3.21 void hebe.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.

3.31.3.22 void hebe.dataflow.sync.GenericBin.setCompName (String l)

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

3.31.3.23 void hebe.dataflow.sync.GenericBin.setComponentType (String *componentType*)

Parameters

| | |
|----------------------|--------------------------|
| <i>componentType</i> | the componentType to set |
|----------------------|--------------------------|

3.31.3.24 void hebe.dataflow.sync.GenericBin.setLabelNome (Label *labelNome*)

Parameters

| | |
|------------------|----------------------|
| <i>labelNome</i> | the labelNome to set |
|------------------|----------------------|

3.31.3.25 void hebe.dataflow.sync.GenericBin.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.31.3.26 void hebe.dataflow.sync.GenericBin.setPortDin1 (PortStdLogicVector *portDin1*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDin1</i> | the portDin1 to set |
|-----------------|---------------------|

3.31.3.27 void hebe.dataflow.sync.GenericBin.setPortDin2 (PortStdLogicVector *portDin2*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDin2</i> | the portDin2 to set |
|-----------------|---------------------|

3.31.3.28 void hebe.dataflow.sync.GenericBin.setPortDout (PortStdLogicVector *portDout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDout</i> | the portDout to set |
|-----------------|---------------------|

3.31.3.29 void hebe.dataflow.sync.GenericBin.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.31.3.30 void hebe.dataflow.sync.GenericBin.setPortRin1 (PortStdLogic1164 *portRin1*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRin1</i> | the portRin1 to set |
|-----------------|---------------------|

3.31.3.31 void hebe.dataflow.sync.GenericBin.setPortRin2 (PortStdLogic1164 *portRin2*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRin2</i> | the portRin2 to set |
|-----------------|---------------------|

3.31.3.32 void hebe.dataflow.sync.GenericBin.setPortRout (PortStdLogic1164 *portRout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRout</i> | the portRout to set |
|-----------------|---------------------|

3.31.3.33 void hebe.dataflow.sync.GenericBin.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.31.3.34 void hebe.dataflow.sync.GenericBin.setS (String *s*)

Parameters

| | |
|----------|--------------|
| <i>s</i> | the s to set |
|----------|--------------|

3.31.3.35 void hebe.dataflow.sync.GenericBin.setString (String *s*)

Method responsible for updating the text displayed by the component.

Parameters

| | |
|----------|-----------------------|
| <i>s</i> | - Text to be updated. |
|----------|-----------------------|

3.31.3.36 void hebe.dataflow.sync.GenericBin.setStringLabel (Label *stringLabel*)

Parameters

| | |
|--------------------|------------------------|
| <i>stringLabel</i> | the stringLabel to set |
|--------------------|------------------------|

3.31.3.37 void hebe.dataflow.sync.GenericBin.setSymbol (Symbol *s*)

Method responsible for updating the component symbol.

Parameters

| | |
|----------|--|
| <i>s</i> | |
|----------|--|

3.31.3.38 void hebe.dataflow.sync.GenericBin.tickDown ()

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

3.31.3.39 void hebe.dataflow.sync.GenericBin.tickUp ()

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

3.31.3.40 void hebe.dataflow.sync.GenericBin.write (java.io.PrintWriter *ps*)

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

Parameters

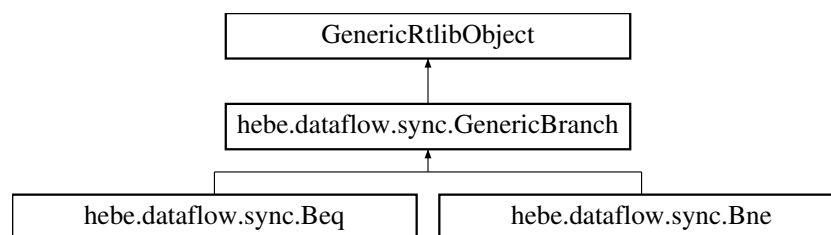
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericBin.java

3.32 hebe.dataflow.sync.GenericBranch Class Reference

Inheritance diagram for hebe.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)

3.32.1 Detailed Description

[GenericBranch](#) component for the UFV synchronous data flow simulator.

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

3.32.2 Constructor & Destructor Documentation

3.32.2.1 `hebe.dataflow.sync.GenericBranch.GenericBranch ()`

Object Constructor.

3.32.3 Member Function Documentation

3.32.3.1 `int hebe.dataflow.sync.GenericBranch.compute (int data1, int data2)`

Method responsible for the computation of the output.

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 1. |

Returns

- Return of computation

3.32.3.2 `void hebe.dataflow.sync.GenericBranch.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.32.3.3 void hebe.dataflow.sync.GenericBranch.constructPorts ()

Method responsible for initializing the component input and output ports.

3.32.3.4 void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.32.3.5 String hebe.dataflow.sync.GenericBranch.getComponentType ()

Returns

the componentType

3.32.3.6 Label hebe.dataflow.sync.GenericBranch.getLabel_nome ()

Returns

the label_nome

3.32.3.7 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortClk ()

Returns

the portClk

3.32.3.8 PortStdLogicVector hebe.dataflow.sync.GenericBranch.getPortDin1 ()

Returns

the portDin1

3.32.3.9 PortStdLogicVector hebe.dataflow.sync.GenericBranch.getPortDin2 ()

Returns

the portDin2

3.32.3.10 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortElse ()

Returns

the portElse

3.32.3.11 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortEn ()

Returns

the portEn

3.32.3.12 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortIf ()

Returns

the portIf

3.32.3.13 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortRin1 ()

Returns

the portRin1

3.32.3.14 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortRin2 ()

Returns

the portRin2

3.32.3.15 PortStdLogic1164 hebe.dataflow.sync.GenericBranch.getPortRst ()

Returns

the portRst

3.32.3.16 String hebe.dataflow.sync.GenericBranch.getS ()

Returns

the s

3.32.3.17 Label hebe.dataflow.sync.GenericBranch.getStringLabel ()

Returns

the stringLabel

3.32.3.18 boolean hebe.dataflow.sync.GenericBranch.initialize (String s)

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

Parameters

| | |
|----------|---|
| <i>s</i> | - Settings for the component read from the file saved by the simulator. |
|----------|---|

Returns

- Returns true if the settings are read successfully.

3.32.3.19 boolean hebe.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.

3.32.3.20 void hebe.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.

3.32.3.21 void hebe.dataflow.sync.GenericBranch.setCompName (String *l*)

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

Parameters

| | |
|----------|---|
| <i>l</i> | - String to be set to the component name. |
|----------|---|

3.32.3.22 void hebe.dataflow.sync.GenericBranch.setComponentType (String *componentType*)

Parameters

| | |
|----------------------|---------------------------------|
| <i>componentType</i> | the <i>componentType</i> to set |
|----------------------|---------------------------------|

3.32.3.23 void hebe.dataflow.sync.GenericBranch.setLabel_nome (Label *label_nome*)

Parameters

| | |
|-------------------|------------------------------|
| <i>label_nome</i> | the <i>label_nome</i> to set |
|-------------------|------------------------------|

3.32.3.24 void hebe.dataflow.sync.GenericBranch.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|---------------------------|
| <i>portClk</i> | the <i>portClk</i> to set |
|----------------|---------------------------|

3.32.3.25 void hebe.dataflow.sync.GenericBranch.setPortDin1 (PortStdLogicVector *portDin1*)

Parameters

| | |
|-----------------|----------------------------|
| <i>portDin1</i> | the <i>portDin1</i> to set |
|-----------------|----------------------------|

3.32.3.26 void hebe.dataflow.sync.GenericBranch.setPortDin2 (PortStdLogicVector *portDin2*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDin2</i> | the portDin2 to set |
|-----------------|---------------------|

3.32.3.27 void hebe.dataflow.sync.GenericBranch.setPortElse (PortStdLogic1164 *portElse*)

Parameters

| | |
|-----------------|---------------------|
| <i>portElse</i> | the portElse to set |
|-----------------|---------------------|

3.32.3.28 void hebe.dataflow.sync.GenericBranch.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.32.3.29 void hebe.dataflow.sync.GenericBranch.setPortIf (PortStdLogic1164 *portIf*)

Parameters

| | |
|---------------|-------------------|
| <i>portIf</i> | the portIf to set |
|---------------|-------------------|

3.32.3.30 void hebe.dataflow.sync.GenericBranch.setPortRin1 (PortStdLogic1164 *portRin1*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRin1</i> | the portRin1 to set |
|-----------------|---------------------|

3.32.3.31 void hebe.dataflow.sync.GenericBranch.setPortRin2 (PortStdLogic1164 *portRin2*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRin2</i> | the portRin2 to set |
|-----------------|---------------------|

3.32.3.32 void hebe.dataflow.sync.GenericBranch.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.32.3.33 void hebe.dataflow.sync.GenericBranch.setS (String *s*)

Parameters

| | |
|----------|--------------|
| <i>s</i> | the s to set |
|----------|--------------|

3.32.3.34 void hebe.dataflow.sync.GenericBranch.setString (String *s*)

Method responsible for updating the text displayed by the component.

Parameters

| | |
|----------|-----------------------|
| <i>s</i> | - Text to be updated. |
|----------|-----------------------|

3.32.3.35 void hebe.dataflow.sync.GenericBranch.setStringLabel (Label *stringLabel*)

Parameters

| | |
|--------------------|------------------------|
| <i>stringLabel</i> | the stringLabel to set |
|--------------------|------------------------|

3.32.3.36 void hebe.dataflow.sync.GenericBranch.setSymbol (Symbol *s*)

Method responsible for updating the component symbol.

Parameters

| | |
|----------|--------------------------------|
| <i>s</i> | - Symbol passed automatically. |
|----------|--------------------------------|

3.32.3.37 void hebe.dataflow.sync.GenericBranch.tickDown ()

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

3.32.3.38 void hebe.dataflow.sync.GenericBranch.tickUp ()

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

3.32.3.39 void hebe.dataflow.sync.GenericBranch.write (java.io.PrintWriter *ps*)

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

Parameters

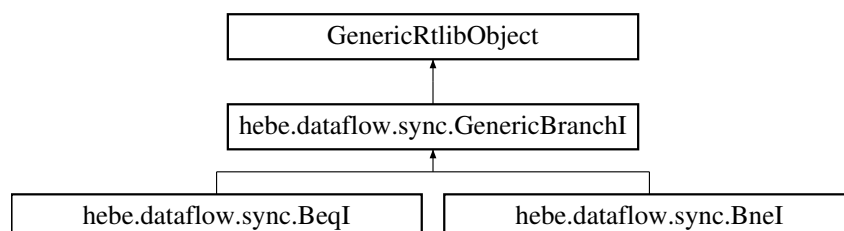
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericBranch.java

3.33 hebe.dataflow.sync.GenericBranchI Class Reference

Inheritance diagram for hebe.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)

3.33.1 Detailed Description

[GenericBranchI](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.33.2 Constructor & Destructor Documentation

3.33.2.1 `hebe.dataflow.sync.GenericBranchI.GenericBranchI ()`

Object Constructor.

3.33.3 Member Function Documentation

3.33.3.1 `int hebe.dataflow.sync.GenericBranchI.compute (int data)`

Method responsible for the computation of the output.

Parameters

| | |
|-------------|---|
| <i>data</i> | - Value to be used for the computation. |
|-------------|---|

Returns

- Return of computation

3.33.3.2 `void hebe.dataflow.sync.GenericBranchI.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.33.3.3 `void hebe.dataflow.sync.GenericBranchI.constructPorts ()`

Method responsible for initializing the component input and output ports.

3.33.3.4 `void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.33.3.5 `Rectangle hebe.dataflow.sync.GenericBranchI.getBackground ()`

Returns

the background

3.33.3.6 `String hebe.dataflow.sync.GenericBranchI.getComponentId ()`

Returns

the componentId

3.33.3.7 `String hebe.dataflow.sync.GenericBranchI.getComponentImmediate ()`

Returns

the componentImmediate

3.33.3.8 `String hebe.dataflow.sync.GenericBranchI.getComponentType ()`

Returns

the componentType

3.33.3.9 `int hebe.dataflow.sync.GenericBranchI.getId ()`

Returns

the id

3.33.3.10 `int hebe.dataflow.sync.GenericBranchI.getImmediate ()`

Returns

the immediate

3.33.3.11 `Label hebe.dataflow.sync.GenericBranchI.getLabelNome ()`

Returns

the labelNome

3.33.3.12 `PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortClk ()`

Returns

the portClk

3.33.3.13 PortStdLogicVector hebe.dataflow.sync.GenericBranchI.getPortDconf ()

Returns

the portDconf

3.33.3.14 PortStdLogicVector hebe.dataflow.sync.GenericBranchI.getPortDin ()

Returns

the portDin

3.33.3.15 PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortElse ()

Returns

the portElse

3.33.3.16 PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortEn ()

Returns

the portEn

3.33.3.17 PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortIf ()

Returns

the portIf

3.33.3.18 PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortRin ()

Returns

the portRin

3.33.3.19 PortStdLogic1164 hebe.dataflow.sync.GenericBranchI.getPortRst ()

Returns

the portRst

3.33.3.20 Label hebe.dataflow.sync.GenericBranchI.getStringLabelId ()

Returns

the stringLabelId

3.33.3.21 Label hebe.dataflow.sync.GenericBranchI.getStringLabelImmediate ()

Returns

the stringLabelImmediate

3.33.3.22 `boolean hebe.dataflow.sync.GenericBranchI.initialize (String s)`

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

Parameters

| | |
|----------|---|
| <i>s</i> | - Settings for the component read from the file saved by the simulator. |
|----------|---|

Returns

- Returns true if the settings are read successfully.

3.33.3.23 void hebe.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

| | |
|-----------|------------------------------------|
| <i>me</i> | - Object where the event occurred. |
|-----------|------------------------------------|

3.33.3.24 boolean hebe.dataflow.sync.GenericBranchI.needsDynamicSymbol ()

Method responsible for indicating to the simulator that the component's 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.

3.33.3.25 void hebe.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.

3.33.3.26 void hebe.dataflow.sync.GenericBranchI.setBackground (Rectangle *background*)

Parameters

| | |
|-------------------|-----------------------|
| <i>background</i> | the background to set |
|-------------------|-----------------------|

3.33.3.27 void hebe.dataflow.sync.GenericBranchI.setCompName (String *l*)

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

Parameters

| | |
|----------|---|
| <i>l</i> | - String to be set to the component name. |
|----------|---|

3.33.3.28 void hebe.dataflow.sync.GenericBranchI.setComponentId (String *componentId*)

Parameters

| | |
|--------------------|------------------------|
| <i>componentId</i> | the componentId to set |
|--------------------|------------------------|

3.33.3.29 void hebe.dataflow.sync.GenericBranchI.setComponentImmediate (String *componentImmediate*)

Parameters

| | |
|---------------------------|-------------------------------|
| <i>componentImmediate</i> | the componentImmediate to set |
|---------------------------|-------------------------------|

3.33.3.30 void hebe.dataflow.sync.GenericBranchI.setComponentType (String *componentType*)

Parameters

| | |
|----------------------|--------------------------|
| <i>componentType</i> | the componentType to set |
|----------------------|--------------------------|

3.33.3.31 void hebe.dataflow.sync.GenericBranchI.setId (int *id*)

Parameters

| | |
|-----------|---------------|
| <i>id</i> | the id to set |
|-----------|---------------|

3.33.3.32 void hebe.dataflow.sync.GenericBranchI.setImmediate (int *immediate*)

Parameters

| | |
|------------------|----------------------|
| <i>immediate</i> | the immediate to set |
|------------------|----------------------|

3.33.3.33 void hebe.dataflow.sync.GenericBranchI.setLabelNome (Label *labelNome*)

Parameters

| | |
|------------------|----------------------|
| <i>labelNome</i> | the labelNome to set |
|------------------|----------------------|

3.33.3.34 void hebe.dataflow.sync.GenericBranchI.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.33.3.35 void hebe.dataflow.sync.GenericBranchI.setPortDconf (PortStdLogicVector *portDconf*)

Parameters

| | |
|------------------|----------------------|
| <i>portDconf</i> | the portDconf to set |
|------------------|----------------------|

3.33.3.36 void hebe.dataflow.sync.GenericBranchI.setPortDin (PortStdLogicVector *portDin*)

Parameters

| | |
|----------------|--------------------|
| <i>portDin</i> | the portDin to set |
|----------------|--------------------|

3.33.3.37 void hebe.dataflow.sync.GenericBranchI.setPortElse (PortStdLogic1164 *portElse*)

Parameters

| | |
|-----------------|---------------------|
| <i>portElse</i> | the portElse to set |
|-----------------|---------------------|

3.33.3.38 void hebe.dataflow.sync.GenericBranchI.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.33.3.39 void hebe.dataflow.sync.GenericBranchI.setPortIf (PortStdLogic1164 *portIf*)

Parameters

| | |
|---------------|-------------------|
| <i>portIf</i> | the portIf to set |
|---------------|-------------------|

3.33.3.40 void hebe.dataflow.sync.GenericBranchI.setPortRin (PortStdLogic1164 *portRin*)

Parameters

| | |
|----------------|--------------------|
| <i>portRin</i> | the portRin to set |
|----------------|--------------------|

3.33.3.41 void hebe.dataflow.sync.GenericBranchI.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.33.3.42 void hebe.dataflow.sync.GenericBranchI.setString (String *componentId*, String *componentImmediate*)

Method responsible for updating the text displayed by the component.

Parameters

| | |
|--------------------|-----------------------|
| <i>componentId</i> | - Text to be updated. |
|--------------------|-----------------------|

3.33.3.43 void hebe.dataflow.sync.GenericBranchI.setStringLabelId (Label *stringLabelId*)

Parameters

| | |
|----------------------|--------------------------|
| <i>stringLabelId</i> | the stringLabelId to set |
|----------------------|--------------------------|

3.33.3.44 void hebe.dataflow.sync.GenericBranchI.setStringLabelImmediate (Label *stringLabelImmediate*)

Parameters

| | |
|-----------------------------|---------------------------------|
| <i>stringLabelImmediate</i> | the stringLabelImmediate to set |
|-----------------------------|---------------------------------|

3.33.3.45 void hebe.dataflow.sync.GenericBranchI.setSymbol (Symbol s)

Method responsible for updating the component symbol.

Parameters

| | |
|----------|--------------------------------|
| <i>s</i> | - Symbol passed automatically. |
|----------|--------------------------------|

3.33.3.46 void hebe.dataflow.sync.GenericBranchI.tickDown ()

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

3.33.3.47 void hebe.dataflow.sync.GenericBranchI.tickUp ()

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

3.33.3.48 void hebe.dataflow.sync.GenericBranchI.write (java.io.PrintWriter ps)

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

Parameters

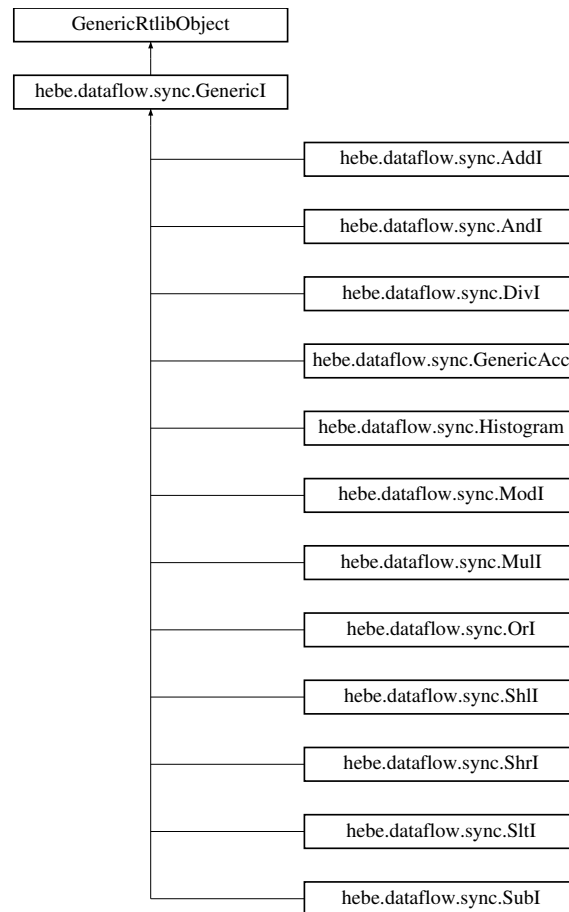
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericBranchI.java

3.34 hebe.dataflow.sync.GenericI Class Reference

Inheritance diagram for hebe.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)
- 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)
- int [getId](#) ()
- void [setId](#) (int id)
- int [getImmediate](#) ()
- void [setImmediate](#) (int immediate)

3.34.1 Detailed Description

[Genericl](#) component for the UFV synchronous data flow simulator.

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

3.34.2 Constructor & Destructor Documentation

3.34.2.1 `hebe.dataflow.sync.Genericl.Genericl ()`

Object Constructor.

3.34.3 Member Function Documentation

3.34.3.1 `int hebe.dataflow.sync.Genericl.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

| | |
|-------------|---|
| <i>data</i> | - Value to be used for the computation. |
|-------------|---|

Returns

- Return of computation

3.34.3.2 void hebe.dataflow.sync.Genericl.constructDynamicSymbol ()

Method responsible for dynamically constructing the component symbol.

3.34.3.3 void hebe.dataflow.sync.Genericl.constructPorts ()

Method responsible for initializing the component input and output ports.

3.34.3.4 void hebe.dataflow.sync.Genericl.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.34.3.5 Rectangle hebe.dataflow.sync.Genericl.getBackground ()

Returns

the background

3.34.3.6 String hebe.dataflow.sync.Genericl.getComponentId ()

Returns

the componentId

3.34.3.7 String hebe.dataflow.sync.Genericl.getComponentImmediate ()

Returns

the componentImmediate

3.34.3.8 String hebe.dataflow.sync.Genericl.getComponentType ()

Returns

the componentType

3.34.3.9 `int hebe.dataflow.sync.GenericI.getId ()`

Returns

the id

3.34.3.10 `int hebe.dataflow.sync.GenericI.getImmediate ()`

Returns

the immediate

3.34.3.11 `Label hebe.dataflow.sync.GenericI.getLabelNome ()`

Returns

the labelNome

3.34.3.12 `PortStdLogic1164 hebe.dataflow.sync.GenericI.getPortClk ()`

Returns

the portClk

3.34.3.13 `PortStdLogicVector hebe.dataflow.sync.GenericI.getPortDconf ()`

Returns

the portDconf

3.34.3.14 `PortStdLogicVector hebe.dataflow.sync.GenericI.getPortDin ()`

Returns

the portDin

3.34.3.15 `PortStdLogicVector hebe.dataflow.sync.GenericI.getPortDout ()`

Returns

the portDout

3.34.3.16 `PortStdLogic1164 hebe.dataflow.sync.GenericI.getPortEn ()`

Returns

the portEn

3.34.3.17 `PortStdLogic1164 hebe.dataflow.sync.GenericI.getPortRin ()`

Returns

the portRin

3.34.3.18 PortStdLogic1164 hebe.dataflow.sync.Genericl.getPortRout ()

Returns

the portRout

3.34.3.19 PortStdLogic1164 hebe.dataflow.sync.Genericl.getPortRst ()

Returns

the portRst

3.34.3.20 Label hebe.dataflow.sync.Genericl.getStringLabelId ()

Returns

the stringLabelId

3.34.3.21 Label hebe.dataflow.sync.Genericl.getStringLabelImmediate ()

Returns

the stringLabelImmediate

3.34.3.22 boolean hebe.dataflow.sync.Genericl.initialize (String s)

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

Parameters

| | |
|----------|---|
| <i>s</i> | - Settings for the component read from the file saved by the simulator. |
|----------|---|

Returns

- Returns true if the settings are read successfully.

3.34.3.23 void hebe.dataflow.sync.Genericl.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

| | |
|-----------|------------------------------------|
| <i>me</i> | - Object where the event occurred. |
|-----------|------------------------------------|

3.34.3.24 boolean hebe.dataflow.sync.Genericl.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.

3.34.3.25 void hebe.dataflow.sync.GenericI.notCompute ()

Method executed when computing is not performed.

3.34.3.26 void hebe.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.

3.34.3.27 void hebe.dataflow.sync.GenericI.setBackground (Rectangle *background*)

Parameters

| | |
|-------------------|-----------------------|
| <i>background</i> | the background to set |
|-------------------|-----------------------|

3.34.3.28 void hebe.dataflow.sync.GenericI.setCompName (String *I*)

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

Parameters

| | |
|----------|---------------------------------------|
| <i>I</i> | - String to be set in component name. |
|----------|---------------------------------------|

3.34.3.29 void hebe.dataflow.sync.GenericI.setComponentId (String *componentId*)

Parameters

| | |
|--------------------|------------------------|
| <i>componentId</i> | the componentId to set |
|--------------------|------------------------|

3.34.3.30 void hebe.dataflow.sync.GenericI.setComponentImmediate (String *componentImmediate*)

Parameters

| | |
|----------------------------|-------------------------------|
| <i>component-Immediate</i> | the componentImmediate to set |
|----------------------------|-------------------------------|

3.34.3.31 void hebe.dataflow.sync.GenericI.setComponentType (String *componentType*)

Parameters

| | |
|----------------------|--------------------------|
| <i>componentType</i> | the componentType to set |
|----------------------|--------------------------|

3.34.3.32 void hebe.dataflow.sync.GenericI.setId (int *id*)

Parameters

| | |
|-----------|---------------|
| <i>id</i> | the id to set |
|-----------|---------------|

3.34.3.33 void hebe.dataflow.sync.GenericI.setImmediate (int *immediate*)

Parameters

| | |
|------------------|----------------------|
| <i>immediate</i> | the immediate to set |
|------------------|----------------------|

3.34.3.34 void hebe.dataflow.sync.Genericl.setLabelNome (Label *labelNome*)

Parameters

| | |
|------------------|----------------------|
| <i>labelNome</i> | the labelNome to set |
|------------------|----------------------|

3.34.3.35 void hebe.dataflow.sync.Genericl.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.34.3.36 void hebe.dataflow.sync.Genericl.setPortDconf (PortStdLogicVector *portDconf*)

Parameters

| | |
|------------------|----------------------|
| <i>portDconf</i> | the portDconf to set |
|------------------|----------------------|

3.34.3.37 void hebe.dataflow.sync.Genericl.setPortDin (PortStdLogicVector *portDin*)

Parameters

| | |
|----------------|--------------------|
| <i>portDin</i> | the portDin to set |
|----------------|--------------------|

3.34.3.38 void hebe.dataflow.sync.Genericl.setPortDout (PortStdLogicVector *portDout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDout</i> | the portDout to set |
|-----------------|---------------------|

3.34.3.39 void hebe.dataflow.sync.Genericl.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.34.3.40 void hebe.dataflow.sync.Genericl.setPortRin (PortStdLogic1164 *portRin*)

Parameters

| | |
|----------------|--------------------|
| <i>portRin</i> | the portRin to set |
|----------------|--------------------|

3.34.3.41 void hebe.dataflow.sync.Genericl.setPortRout (PortStdLogic1164 *portRout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRout</i> | the portRout to set |
|-----------------|---------------------|

3.34.3.42 void hebe.dataflow.sync.GenericI.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.34.3.43 void hebe.dataflow.sync.GenericI.setString (String *componentId*, String *componentImmediate*)

Method responsible for updating the text displayed by the component.

Parameters

| | |
|----------------------------|-----------------------|
| <i>componentId</i> | - Text to be updated. |
| <i>component-Immediate</i> | - Text to be updated. |

3.34.3.44 void hebe.dataflow.sync.GenericI.setStringLabelId (Label *stringLabelId*)

Parameters

| | |
|----------------------|--------------------------|
| <i>stringLabelId</i> | the stringLabelId to set |
|----------------------|--------------------------|

3.34.3.45 void hebe.dataflow.sync.GenericI.setStringLabelImmediate (Label *stringLabelImmediate*)

Parameters

| | |
|------------------------------|---------------------------------|
| <i>stringLabel-Immediate</i> | the stringLabelImmediate to set |
|------------------------------|---------------------------------|

3.34.3.46 void hebe.dataflow.sync.GenericI.setSymbol (Symbol *s*)

Method responsible for updating the component symbol.

Parameters

| | |
|----------|--------------------------------|
| <i>s</i> | - Symbol passed automatically. |
|----------|--------------------------------|

3.34.3.47 void hebe.dataflow.sync.GenericI.tickDown ()

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

3.34.3.48 void hebe.dataflow.sync.GenericI.tickUp ()

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

3.34.3.49 void hebe.dataflow.sync.Genericl.write (java.io.PrintWriter *ps*)

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

Parameters

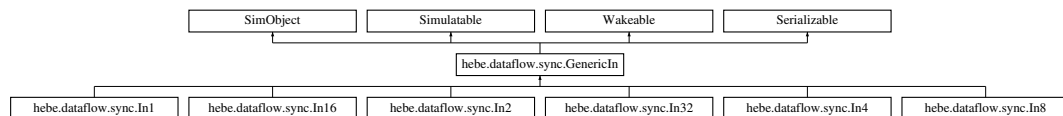
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericIn.java

3.35 hebe.dataflow.sync.GenericIn Class Reference

Inheritance diagram for hebe.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 [isEnabledAnimationFlag](#) ()

- void [setEnabledAnimationFlag](#) (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](#) ()

Protected Attributes

- double **delay**

3.35.1 Detailed Description

[GenericIn](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.35.2 Constructor & Destructor Documentation

3.35.2.1 `hebe.dataflow.sync.GenericIn.GenericIn ()`

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

3.35.2.2 `hebe.dataflow.sync.GenericIn.GenericIn (int QTDE_PORTS)`

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

Parameters

| | |
|-------------------|---|
| <i>QTDE_PORTS</i> | - Number of queue outputs to be created |
|-------------------|---|

3.35.3 Member Function Documentation

3.35.3.1 `void hebe.dataflow.sync.GenericIn.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.35.3.2 `void hebe.dataflow.sync.GenericIn.constructPorts ()`

Method responsible for initializing the component input and output ports.

3.35.3.3 `void hebe.dataflow.sync.GenericIn.constructStandardValues ()` *[protected]*

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

3.35.3.4 `void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.35.3.5 `String hebe.dataflow.sync.GenericIn.getComponentType ()`

Returns

the `componentType`

3.35.3.6 `double hebe.dataflow.sync.GenericIn.getDefaultdelay ()`

Returns

the `defaultdelay`

3.35.3.7 double hebe.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 delay of the component.

3.35.3.8 int hebe.dataflow.sync.GenericIn.getIdxDin ()**Returns**

the idxDin

3.35.3.9 FlexibleLabelFormatter hebe.dataflow.sync.GenericIn.getLabelFormatter ()**Returns**

the labelFormatter

3.35.3.10 int hebe.dataflow.sync.GenericIn.getN_bits ()**Returns**

the n_bits

3.35.3.11 PortStdLogic1164 hebe.dataflow.sync.GenericIn.getPortClk ()**Returns**

the portClk

3.35.3.12 PortStdLogicVector hebe.dataflow.sync.GenericIn.getPortDconf ()**Returns**

the portDconf

3.35.3.13 PortStdLogicVector [] hebe.dataflow.sync.GenericIn.getPortDout ()**Returns**

the portDout

3.35.3.14 PortStdLogic1164 hebe.dataflow.sync.GenericIn.getPortEnOut ()**Returns**

the portEnOut

3.35.3.15 PortStdLogic1164 hebe.dataflow.sync.GenericIn.getPortRdy ()

Returns

the portRdy

3.35.3.16 PortStdLogic1164 [] hebe.dataflow.sync.GenericIn.getPortRout ()

Returns

the portRout

3.35.3.17 PortStdLogic1164 hebe.dataflow.sync.GenericIn.getPortRst ()

Returns

the portRst

3.35.3.18 int hebe.dataflow.sync.GenericIn.getQTDE_PORTS ()

Returns

the QTDE_PORTS

3.35.3.19 int hebe.dataflow.sync.GenericIn.getTOT_PORTS ()

Returns

the TOT_PORTS

3.35.3.20 ColoredValueLabel hebe.dataflow.sync.GenericIn.getValueLabel ()

Returns

the valueLabel

3.35.3.21 StdLogicVector hebe.dataflow.sync.GenericIn.getVector ()

Returns

the vector

3.35.3.22 StdLogicVector hebe.dataflow.sync.GenericIn.getVector_000 ()

Returns

the vector_000

3.35.3.23 StdLogicVector hebe.dataflow.sync.GenericIn.getVector_111 ()

Returns

the vector_111

3.35.3.24 StdLogicVector hebe.dataflow.sync.GenericIn.getVector_UUU ()

Returns

the vector_UUU

3.35.3.25 StdLogicVector hebe.dataflow.sync.GenericIn.getVector_XXX ()

Returns

the vector_XXX

3.35.3.26 StdLogicVector hebe.dataflow.sync.GenericIn.getVector_ZZZ ()

Returns

the vector_ZZZ

3.35.3.27 int [] hebe.dataflow.sync.GenericIn.getVectorIn ()

Returns

the vectorIn

3.35.3.28 PortStdLogicVector hebe.dataflow.sync.GenericIn.getVectorOutputPort ()

Returns

the vectorOutputPort

3.35.3.29 boolean hebe.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.

3.35.3.30 boolean hebe.dataflow.sync.GenericIn.isEnableAnimationFlag ()

Returns

the enableAnimationFlag

3.35.3.31 boolean hebe.dataflow.sync.GenericIn.isStart ()

Returns

the start

3.35.3.32 boolean hebe.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.

3.35.3.33 void hebe.dataflow.sync.GenericIn.setCompName (String *l*)

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

Parameters

| | |
|----------|---|
| <i>l</i> | - String to be set to the component name. |
|----------|---|

3.35.3.34 void hebe.dataflow.sync.GenericIn.setComponentType (String *componentType*)

Parameters

| | |
|----------------------|---------------------------------|
| <i>componentType</i> | the <i>componentType</i> to set |
|----------------------|---------------------------------|

3.35.3.35 void hebe.dataflow.sync.GenericIn.setDefaultdelay (double *defaultdelay*)

Parameters

| | |
|---------------------|--------------------------------|
| <i>defaultdelay</i> | the <i>defaultdelay</i> to set |
|---------------------|--------------------------------|

3.35.3.36 void hebe.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

| | |
|---------------|--|
| <i>_delay</i> | |
|---------------|--|

3.35.3.37 void hebe.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

| | |
|----------|--|
| <i>s</i> | |
|----------|--|

3.35.3.38 void hebe.dataflow.sync.GenericIn.setEnableAnimationFlag (boolean *enableAnimationFlag*)

Parameters

| | |
|-----------------------------|--------------------------------|
| <i>enable-AnimationFlag</i> | the enableAnimationFlag to set |
|-----------------------------|--------------------------------|

3.35.3.39 void hebe.dataflow.sync.GenericIn.setIdxDin (int *idxDin*)

Parameters

| | |
|---------------|-------------------|
| <i>idxDin</i> | the idxDin to set |
|---------------|-------------------|

3.35.3.40 void hebe.dataflow.sync.GenericIn.setLabelFormatter (FlexibleLabelFormatter *labelFormatter*)

Parameters

| | |
|-----------------------|---------------------------|
| <i>labelFormatter</i> | the labelFormatter to set |
|-----------------------|---------------------------|

3.35.3.41 void hebe.dataflow.sync.GenericIn.setN_bits (int *n_bits*)

Parameters

| | |
|---------------|-------------------|
| <i>n_bits</i> | the n_bits to set |
|---------------|-------------------|

3.35.3.42 void hebe.dataflow.sync.GenericIn.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.35.3.43 void hebe.dataflow.sync.GenericIn.setPortDconf (PortStdLogicVector *portDconf*)

Parameters

| | |
|------------------|----------------------|
| <i>portDconf</i> | the portDconf to set |
|------------------|----------------------|

3.35.3.44 void hebe.dataflow.sync.GenericIn.setPortDout (PortStdLogicVector[] *portDout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDout</i> | the portDout to set |
|-----------------|---------------------|

3.35.3.45 void hebe.dataflow.sync.GenericIn.setPortEnOut (PortStdLogic1164 *portEnOut*)

Parameters

| | |
|------------------|----------------------|
| <i>portEnOut</i> | the portEnOut to set |
|------------------|----------------------|

3.35.3.46 void hebe.dataflow.sync.GenericIn.setPortRdy (PortStdLogic1164 *portRdy*)

Parameters

| | |
|----------------|--------------------|
| <i>portRdy</i> | the portRdy to set |
|----------------|--------------------|

3.35.3.47 void hebe.dataflow.sync.GenericIn.setPortRout (PortStdLogic1164[] *portRout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRout</i> | the portRout to set |
|-----------------|---------------------|

3.35.3.48 void hebe.dataflow.sync.GenericIn.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.35.3.49 void hebe.dataflow.sync.GenericIn.setStart (boolean *start*)

Parameters

| | |
|--------------|------------------|
| <i>start</i> | the start to set |
|--------------|------------------|

3.35.3.50 void hebe.dataflow.sync.GenericIn.setValueLabel (ColoredValueLabel *valueLabel*)

Parameters

| | |
|-------------------|-----------------------|
| <i>valueLabel</i> | the valueLabel to set |
|-------------------|-----------------------|

3.35.3.51 void hebe.dataflow.sync.GenericIn.setVector (StdLogicVector *vector*)

Parameters

| | |
|---------------|-------------------|
| <i>vector</i> | the vector to set |
|---------------|-------------------|

3.35.3.52 void hebe.dataflow.sync.GenericIn.setVector_000 (StdLogicVector *vector_000*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_000</i> | the vector_000 to set |
|-------------------|-----------------------|

3.35.3.53 void hebe.dataflow.sync.GenericIn.setVector_111 (StdLogicVector *vector_111*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_111</i> | the vector_111 to set |
|-------------------|-----------------------|

3.35.3.54 void hebe.dataflow.sync.GenericIn.setVector_UUU (StdLogicVector *vector_UUU*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_UUU</i> | the vector_UUU to set |
|-------------------|-----------------------|

3.35.3.55 void hebe.dataflow.sync.GenericIn.setVector_XXX (StdLogicVector *vector_XXX*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_XXX</i> | the vector_XXX to set |
|-------------------|-----------------------|

3.35.3.56 void hebe.dataflow.sync.GenericIn.setVector_ZZZ (StdLogicVector *vector_ZZZ*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_ZZZ</i> | the vector_ZZZ to set |
|-------------------|-----------------------|

3.35.3.57 void hebe.dataflow.sync.GenericIn.setVectorIn (int[] *vectorIn*)

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

Parameters

| | |
|-----------------|--|
| <i>vectorIn</i> | - Vector that will be delivered to the outputs |
|-----------------|--|

3.35.3.58 void hebe.dataflow.sync.GenericIn.setVectorOutputPort (PortStdLogicVector *vectorOutputPort*)

Parameters

| | |
|-------------------------|-----------------------------|
| <i>vectorOutputPort</i> | the vectorOutputPort to set |
|-------------------------|-----------------------------|

3.35.3.59 void hebe.dataflow.sync.GenericIn.updateSymbol ()

Method responsible for updating the component symbol.

3.35.3.60 void hebe.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

| | |
|------------|--------------------------|
| <i>arg</i> | - Object to be awakened. |
|------------|--------------------------|

3.35.3.61 void hebe.dataflow.sync.GenericIn.write (java.io.PrintWriter *ps*)

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

Parameters

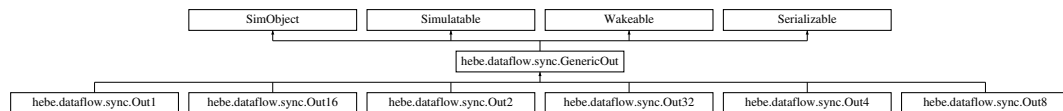
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericIn.java

3.36 hebe.dataflow.sync.GenericOut Class Reference

Inheritance diagram for hebe.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 [isEnabledAnimationFlag](#) ()
- void [setEnabledAnimationFlag](#) (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](#) ()

Protected Attributes

- double [delay](#)

3.36.1 Detailed Description

[GenericOut](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.36.2 Constructor & Destructor Documentation

3.36.2.1 `hebe.dataflow.sync.GenericOut.GenericOut ()`

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

3.36.2.2 `hebe.dataflow.sync.GenericOut.GenericOut (int QTDE_PORTS)`

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

Parameters

| | |
|-------------------|--|
| <i>QTDE_PORTS</i> | - Number of queue inputs to be created |
|-------------------|--|

3.36.3 Member Function Documentation

3.36.3.1 `void hebe.dataflow.sync.GenericOut.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.36.3.2 `void hebe.dataflow.sync.GenericOut.constructPorts ()`

Method responsible for initializing the component input and output ports.

3.36.3.3 `void hebe.dataflow.sync.GenericOut.constructStandardValues ()` [protected]

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

3.36.3.4 `void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.36.3.5 `String hebe.dataflow.sync.GenericOut.getComponentType ()`

Returns

the componentType

3.36.3.6 `double hebe.dataflow.sync.GenericOut.getDefaultdelay ()`

Returns

the defaultdelay

3.36.3.7 double hebe.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

3.36.3.8 boolean hebe.dataflow.sync.GenericOut.getDoneSignal ()

Method responsible for returning end of data entry.

Returns

- Returns the value of done signal.

3.36.3.9 int hebe.dataflow.sync.GenericOut.getIdxDout ()**Returns**

- the idxDout

3.36.3.10 FlexibleLabelFormatter hebe.dataflow.sync.GenericOut.getLabelFormatter ()**Returns**

- the labelFormatter

3.36.3.11 int hebe.dataflow.sync.GenericOut.getN_bits ()**Returns**

- the n_bits

3.36.3.12 PortStdLogic1164 hebe.dataflow.sync.GenericOut.getPortClk ()**Returns**

- the portClk

3.36.3.13 PortStdLogicVector [] hebe.dataflow.sync.GenericOut.getPortDin ()**Returns**

- the portDin

3.36.3.14 PortStdLogic1164 hebe.dataflow.sync.GenericOut.getPortEn ()**Returns**

- the portEn

3.36.3.15 PortStdLogic1164 hebe.dataflow.sync.GenericOut.getPortRdy ()

Returns

the portRdy

3.36.3.16 PortStdLogic1164 [] hebe.dataflow.sync.GenericOut.getPortRin ()

Returns

the portRin

3.36.3.17 PortStdLogic1164 hebe.dataflow.sync.GenericOut.getPortRst ()

Returns

the portRst

3.36.3.18 int hebe.dataflow.sync.GenericOut.getQTDE_PORTS ()

Returns

the QTDE_PORTS

3.36.3.19 int hebe.dataflow.sync.GenericOut.getQtdeSave ()

Returns

the qtdeSave

3.36.3.20 int hebe.dataflow.sync.GenericOut.getTamVectorOut ()

Returns

the tamVectorOut

3.36.3.21 int hebe.dataflow.sync.GenericOut.getTOT_PORTS ()

Returns

the TOT_PORTS

3.36.3.22 ColoredValueLabel hebe.dataflow.sync.GenericOut.getValueLabel ()

Returns

the valueLabel

3.36.3.23 StdLogicVector hebe.dataflow.sync.GenericOut.getVector ()

Returns

the vector

3.36.3.24 StdLogicVector hebe.dataflow.sync.GenericOut.getVector_000 ()

Returns

the vector_000

3.36.3.25 StdLogicVector hebe.dataflow.sync.GenericOut.getVector_111 ()

Returns

the vector_111

3.36.3.26 StdLogicVector hebe.dataflow.sync.GenericOut.getVector_UUU ()

Returns

the vector_UUU

3.36.3.27 StdLogicVector hebe.dataflow.sync.GenericOut.getVector_XXX ()

Returns

the vector_XXX

3.36.3.28 StdLogicVector hebe.dataflow.sync.GenericOut.getVector_ZZZ ()

Returns

the vector_ZZZ

3.36.3.29 int [] hebe.dataflow.sync.GenericOut.getVectorOut ()

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

Returns

- Returns the vector with the processed data.

3.36.3.30 PortStdLogicVector hebe.dataflow.sync.GenericOut.getVectorOutputPort ()

Returns

the vectorOutputPort

3.36.3.31 boolean hebe.dataflow.sync.GenericOut.initialize (String s)

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

Parameters

| | |
|----------|---|
| <i>s</i> | - Settings for the component read from the file saved by the simulator. |
|----------|---|

Returns

- Returns true if the settings are read successfully.

3.36.3.32 `boolean hebe.dataflow.sync.GenericOut.isDone ()`

Returns

the done

3.36.3.33 `boolean hebe.dataflow.sync.GenericOut.isEnableAnimationFlag ()`

Returns

the enableAnimationFlag

3.36.3.34 `boolean hebe.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.

3.36.3.35 `void hebe.dataflow.sync.GenericOut.setCompName (String /)`

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

Parameters

| | |
|----------|---|
| <i>/</i> | - String to be set to the component name. |
|----------|---|

3.36.3.36 `void hebe.dataflow.sync.GenericOut.setComponentType (String componentType)`

Parameters

| | |
|----------------------|--------------------------|
| <i>componentType</i> | the componentType to set |
|----------------------|--------------------------|

3.36.3.37 `void hebe.dataflow.sync.GenericOut.setDefaultdelay (double defaultdelay)`

Parameters

| | |
|---------------------|-------------------------|
| <i>defaultdelay</i> | the defaultdelay to set |
|---------------------|-------------------------|

3.36.3.38 `void hebe.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

| | |
|---------------|--|
| <i>_delay</i> | |
|---------------|--|

3.36.3.39 void hebe.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

| | |
|----------|--|
| <i>s</i> | |
|----------|--|

3.36.3.40 void hebe.dataflow.sync.GenericOut.setDone (boolean *done*)

Parameters

| | |
|-------------|-----------------|
| <i>done</i> | the done to set |
|-------------|-----------------|

3.36.3.41 void hebe.dataflow.sync.GenericOut.setEnableAnimationFlag (boolean *enableAnimationFlag*)

Parameters

| | |
|-----------------------------|--------------------------------|
| <i>enable-AnimationFlag</i> | the enableAnimationFlag to set |
|-----------------------------|--------------------------------|

3.36.3.42 void hebe.dataflow.sync.GenericOut.setIdxDout (int *idxDout*)

Parameters

| | |
|----------------|--------------------|
| <i>idxDout</i> | the idxDout to set |
|----------------|--------------------|

3.36.3.43 void hebe.dataflow.sync.GenericOut.setLabelFormatter (FlexibleLabelFormatter *labelFormatter*)

Parameters

| | |
|-----------------------|---------------------------|
| <i>labelFormatter</i> | the labelFormatter to set |
|-----------------------|---------------------------|

3.36.3.44 void hebe.dataflow.sync.GenericOut.setN_bits (int *n_bits*)

Parameters

| | |
|---------------|-------------------|
| <i>n_bits</i> | the n_bits to set |
|---------------|-------------------|

3.36.3.45 void hebe.dataflow.sync.GenericOut.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|--|--|
| | |
|--|--|

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.36.3.46 void hebe.dataflow.sync.GenericOut.setPortDin (PortStdLogicVector[] *portDin*)

Parameters

| | |
|----------------|--------------------|
| <i>portDin</i> | the portDin to set |
|----------------|--------------------|

3.36.3.47 void hebe.dataflow.sync.GenericOut.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.36.3.48 void hebe.dataflow.sync.GenericOut.setPortRdy (PortStdLogic1164 *portRdy*)

Parameters

| | |
|----------------|--------------------|
| <i>portRdy</i> | the portRdy to set |
|----------------|--------------------|

3.36.3.49 void hebe.dataflow.sync.GenericOut.setPortRin (PortStdLogic1164[] *portRin*)

Parameters

| | |
|----------------|--------------------|
| <i>portRin</i> | the portRin to set |
|----------------|--------------------|

3.36.3.50 void hebe.dataflow.sync.GenericOut.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.36.3.51 void hebe.dataflow.sync.GenericOut.setQtdeSave (int *qtde_save*)

Parameters

| | |
|------------------|--|
| <i>qtde_save</i> | |
|------------------|--|

3.36.3.52 void hebe.dataflow.sync.GenericOut.setTamVectorOut (int *tamVectorOut*)

Parameters

| | |
|---------------------|-------------------------|
| <i>tamVectorOut</i> | the tamVectorOut to set |
|---------------------|-------------------------|

3.36.3.53 void hebe.dataflow.sync.GenericOut.setValueLabel (ColoredValueLabel *valueLabel*)

Parameters

| | |
|-------------------|-----------------------|
| <i>valueLabel</i> | the valueLabel to set |
|-------------------|-----------------------|

3.36.3.54 void hebe.dataflow.sync.GenericOut.setVector (int *k*)

Method responsible for inserting elements into the vector.

Parameters

| | |
|----------|-----------------------------------|
| <i>k</i> | - Value to be inserted in vector. |
|----------|-----------------------------------|

3.36.3.55 void hebe.dataflow.sync.GenericOut.setVector (StdLogicVector *vector*)

Parameters

| | |
|---------------|-------------------|
| <i>vector</i> | the vector to set |
|---------------|-------------------|

3.36.3.56 void hebe.dataflow.sync.GenericOut.setVector_000 (StdLogicVector *vector_000*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_000</i> | the vector_000 to set |
|-------------------|-----------------------|

3.36.3.57 void hebe.dataflow.sync.GenericOut.setVector_111 (StdLogicVector *vector_111*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_111</i> | the vector_111 to set |
|-------------------|-----------------------|

3.36.3.58 void hebe.dataflow.sync.GenericOut.setVector_UUU (StdLogicVector *vector_UUU*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_UUU</i> | the vector_UUU to set |
|-------------------|-----------------------|

3.36.3.59 void hebe.dataflow.sync.GenericOut.setVector_XXX (StdLogicVector *vector_XXX*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_XXX</i> | the vector_XXX to set |
|-------------------|-----------------------|

3.36.3.60 void hebe.dataflow.sync.GenericOut.setVector_ZZZ (StdLogicVector *vector_ZZZ*)

Parameters

| | |
|-------------------|-----------------------|
| <i>vector_ZZZ</i> | the vector_ZZZ to set |
|-------------------|-----------------------|

3.36.3.61 void hebe.dataflow.sync.GenericOut.setVectorOut (int[] *vectorOut*)

Parameters

| | |
|------------------|----------------------|
| <i>vectorOut</i> | the vectorOut to set |
|------------------|----------------------|

3.36.3.62 void hebe.dataflow.sync.GenericOut.setVectorOutputPort (PortStdLogicVector *vectorOutputPort*)

Parameters

| | |
|-------------------------|-----------------------------|
| <i>vectorOutputPort</i> | the vectorOutputPort to set |
|-------------------------|-----------------------------|

3.36.3.63 void hebe.dataflow.sync.GenericOut.updateSymbol ()

Method responsible for updating the component symbol.

3.36.3.64 void hebe.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

| | |
|------------|--------------------------|
| <i>arg</i> | - Object to be awakened. |
|------------|--------------------------|

3.36.3.65 void hebe.dataflow.sync.GenericOut.write (java.io.PrintWriter *ps*)

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

Parameters

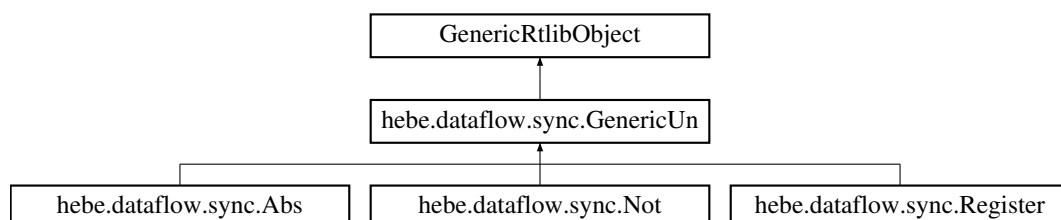
| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericOut.java

3.37 hebe.dataflow.sync.GenericUn Class Reference

Inheritance diagram for hebe.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)

3.37.1 Detailed Description

[GenericUn](#) component for the UFV synchronous data flow simulator.

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

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.37.2 Constructor & Destructor Documentation

3.37.2.1 `hebe.dataflow.sync.GenericUn.GenericUn ()`

Object Constructor.

3.37.3 Member Function Documentation

3.37.3.1 `int hebe.dataflow.sync.GenericUn.compute (int data)`

Method responsible for the computation of the output.

Parameters

| | |
|-------------|---|
| <i>data</i> | - Value to be used for the computation. |
|-------------|---|

Returns

- Return of computation

3.37.3.2 `void hebe.dataflow.sync.GenericUn.constructDynamicSymbol ()`

Method responsible for dynamically constructing the component symbol.

3.37.3.3 `void hebe.dataflow.sync.GenericUn.constructPorts ()`

Method responsible for initializing the component input and output ports.

3.37.3.4 `void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.37.3.5 `Rectangle hebe.dataflow.sync.GenericUn.getBackground ()`

Returns

the background

3.37.3.6 `String hebe.dataflow.sync.GenericUn.getComponentType ()`

Returns

the componentType

3.37.3.7 Label hebe.dataflow.sync.GenericUn.getLabelNome ()

Returns

the labelNome

3.37.3.8 PortStdLogic1164 hebe.dataflow.sync.GenericUn.getPortClk ()

Returns

the portClk

3.37.3.9 PortStdLogicVector hebe.dataflow.sync.GenericUn.getPortDin ()

Returns

the portDin

3.37.3.10 PortStdLogicVector hebe.dataflow.sync.GenericUn.getPortDout ()

Returns

the portDout

3.37.3.11 PortStdLogic1164 hebe.dataflow.sync.GenericUn.getPortEn ()

Returns

the portEn

3.37.3.12 PortStdLogic1164 hebe.dataflow.sync.GenericUn.getPortRin ()

Returns

the portRin

3.37.3.13 PortStdLogic1164 hebe.dataflow.sync.GenericUn.getPortRout ()

Returns

the portRout

3.37.3.14 PortStdLogic1164 hebe.dataflow.sync.GenericUn.getPortRst ()

Returns

the portRst

3.37.3.15 String hebe.dataflow.sync.GenericUn.getS ()

Returns

the s

3.37.3.16 Label `hebe.dataflow.sync.GenericUn.getStringLabel ()`

Returns

the `stringLabel`

3.37.3.17 `boolean hebe.dataflow.sync.GenericUn.initialize (String s)`

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

Parameters

| | |
|----------------|---|
| <code>s</code> | - Settings for the component read from the file saved by the simulator. |
|----------------|---|

Returns

- Returns true if the settings are read successfully.

3.37.3.18 `boolean hebe.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.

3.37.3.19 `void hebe.dataflow.sync.GenericUn.notCompute ()`

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

3.37.3.20 `void hebe.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.

3.37.3.21 `void hebe.dataflow.sync.GenericUn.setBackground (Rectangle background)`

Parameters

| | |
|-------------------------|-----------------------|
| <code>background</code> | the background to set |
|-------------------------|-----------------------|

3.37.3.22 `void hebe.dataflow.sync.GenericUn.setCompName (String /)`

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

Parameters

| | |
|----------------|---------------------------------------|
| <code>/</code> | - String to be set in component name. |
|----------------|---------------------------------------|

3.37.3.23 `void hebe.dataflow.sync.GenericUn.setComponentType (String componentType)`

Parameters

| | |
|----------------------|--------------------------|
| <i>componentType</i> | the componentType to set |
|----------------------|--------------------------|

3.37.3.24 void hebe.dataflow.sync.GenericUn.setLabelNome (Label *labelNome*)

Parameters

| | |
|------------------|----------------------|
| <i>labelNome</i> | the labelNome to set |
|------------------|----------------------|

3.37.3.25 void hebe.dataflow.sync.GenericUn.setPortClk (PortStdLogic1164 *portClk*)

Parameters

| | |
|----------------|--------------------|
| <i>portClk</i> | the portClk to set |
|----------------|--------------------|

3.37.3.26 void hebe.dataflow.sync.GenericUn.setPortDin (PortStdLogicVector *portDin*)

Parameters

| | |
|----------------|--------------------|
| <i>portDin</i> | the portDin to set |
|----------------|--------------------|

3.37.3.27 void hebe.dataflow.sync.GenericUn.setPortDout (PortStdLogicVector *portDout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portDout</i> | the portDout to set |
|-----------------|---------------------|

3.37.3.28 void hebe.dataflow.sync.GenericUn.setPortEn (PortStdLogic1164 *portEn*)

Parameters

| | |
|---------------|-------------------|
| <i>portEn</i> | the portEn to set |
|---------------|-------------------|

3.37.3.29 void hebe.dataflow.sync.GenericUn.setPortRin (PortStdLogic1164 *portRin*)

Parameters

| | |
|----------------|--------------------|
| <i>portRin</i> | the portRin to set |
|----------------|--------------------|

3.37.3.30 void hebe.dataflow.sync.GenericUn.setPortRout (PortStdLogic1164 *portRout*)

Parameters

| | |
|-----------------|---------------------|
| <i>portRout</i> | the portRout to set |
|-----------------|---------------------|

3.37.3.31 void hebe.dataflow.sync.GenericUn.setPortRst (PortStdLogic1164 *portRst*)

Parameters

| | |
|----------------|--------------------|
| <i>portRst</i> | the portRst to set |
|----------------|--------------------|

3.37.3.32 void hebe.dataflow.sync.GenericUn.setS (String s)

Parameters

| | |
|----------|--------------|
| <i>s</i> | the s to set |
|----------|--------------|

3.37.3.33 void hebe.dataflow.sync.GenericUn.setString (String s)

Method responsible for updating the text displayed by the component.

Parameters

| | |
|----------|-----------------------|
| <i>s</i> | - Text to be updated. |
|----------|-----------------------|

3.37.3.34 void hebe.dataflow.sync.GenericUn.setStringLabel (Label stringLabel)

Parameters

| | |
|--------------------|------------------------|
| <i>stringLabel</i> | the stringLabel to set |
|--------------------|------------------------|

3.37.3.35 void hebe.dataflow.sync.GenericUn.setSymbol (Symbol s)

Method responsible for updating the component symbol.

Parameters

| | |
|----------|--------------------------------|
| <i>s</i> | - Symbol passed automatically. |
|----------|--------------------------------|

3.37.3.36 void hebe.dataflow.sync.GenericUn.tickDown ()

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

3.37.3.37 void hebe.dataflow.sync.GenericUn.tickUp ()

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

3.37.3.38 void hebe.dataflow.sync.GenericUn.write (java.io.PrintWriter ps)

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

Parameters

| | |
|-----------|----------------------------|
| <i>ps</i> | -Simulator writing object. |
|-----------|----------------------------|

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

- hebe/dataflow/sync/GenericUn.java

3.38 hebe.examples.dataflow_sync.GourandDataflowFPGA Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.38.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/GourandDataflowFPGA.java

3.39 hebe.examples.dataflow_sync.GourandDataflowSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.39.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

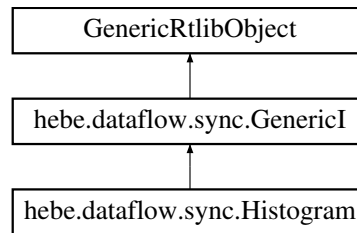
* 1.0

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

- hebe/examples/dataflow_sync/GourandDataflowSimulation.java

3.40 hebe.dataflow.sync.Histogram Class Reference

Inheritance diagram for hebe.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](#) ()

3.40.1 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

3.40.2 Constructor & Destructor Documentation

3.40.2.1 hebe.dataflow.sync.Histogram.Histogram ()

Object Constructor.

3.40.3 Member Function Documentation

3.40.3.1 int hebe.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - 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.

3.40.3.2 void hebe.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

| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

3.40.3.3 int hebe.dataflow.sync.Histogram.getCounter ()

Returns

the counter

3.40.3.4 int hebe.dataflow.sync.Histogram.getDecr ()

Returns

the decr

3.40.3.5 int [] hebe.dataflow.sync.Histogram.getHistogram ()

Returns

the histogram

3.40.3.6 int hebe.dataflow.sync.Histogram.getNUMBITS ()

Returns

the NUMBITS

3.40.3.7 void hebe.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.

3.40.3.8 void hebe.dataflow.sync.Histogram.setCounter (int *counter*)

Parameters

| | |
|----------------|--------------------|
| <i>counter</i> | the counter to set |
|----------------|--------------------|

3.40.3.9 void `hebe.dataflow.sync.Histogram.setDecr (int decr)`

Parameters

| | |
|-------------|-----------------|
| <i>decr</i> | the decr to set |
|-------------|-----------------|

3.40.3.10 void `hebe.dataflow.sync.Histogram.setHistogram (int[] histogram)`

Parameters

| | |
|------------------|----------------------|
| <i>histogram</i> | the histogram to set |
|------------------|----------------------|

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

- `hebe/dataflow/sync/Histogram.java`

3.41 `hebe.examples.dataflow_sync.HistogramDataflowFpgaSimulation` Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.41.1 Detailed Description

[HistogramDataflowSimulation](#) example in FPGA Board.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- `hebe/examples/dataflow_sync/HistogramDataflowFpgaSimulation.java`

3.42 `hebe.examples.dataflow_sync.HistogramDataflowSimulation` Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.42.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

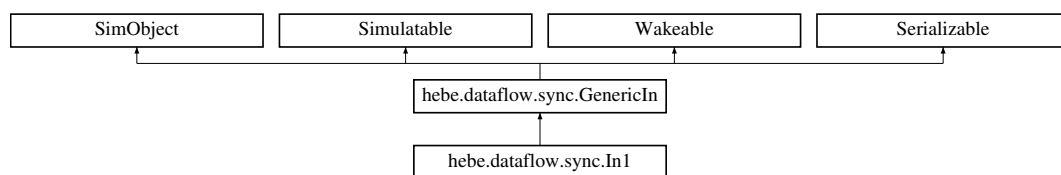
* 1.0

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

- [hebe/examples/dataflow_sync/HistogramDataflowSimulation.java](#)

3.43 hebe.dataflow.sync.In1 Class Reference

Inheritance diagram for hebe.dataflow.sync.In1:



Public Member Functions

- [In1](#) ()

Additional Inherited Members

3.43.1 Detailed Description

[In1](#) component for the UFV synchronous data flow simulator.

The component implements an input queue with 1 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.43.2 Constructor & Destructor Documentation

3.43.2.1 `hebe.dataflow.sync.In1.In1 ()`

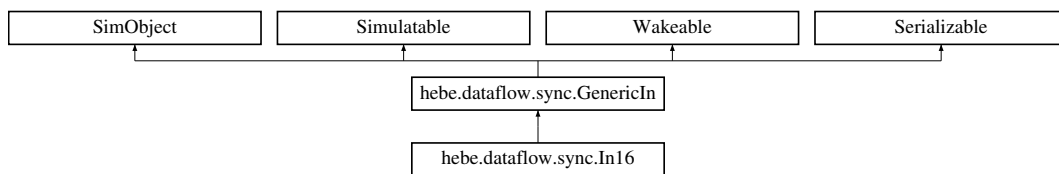
Object Constructor.

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

- `hebe/dataflow/sync/In1.java`

3.44 `hebe.dataflow.sync.In16` Class Reference

Inheritance diagram for `hebe.dataflow.sync.In16`:



Public Member Functions

- [In16 \(\)](#)

Additional Inherited Members

3.44.1 Detailed Description

`In16` component for the UFV synchronous data flow simulator.

The component implements an input queue with 16 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.44.2 Constructor & Destructor Documentation

3.44.2.1 `hebe.dataflow.sync.In16.In16 ()`

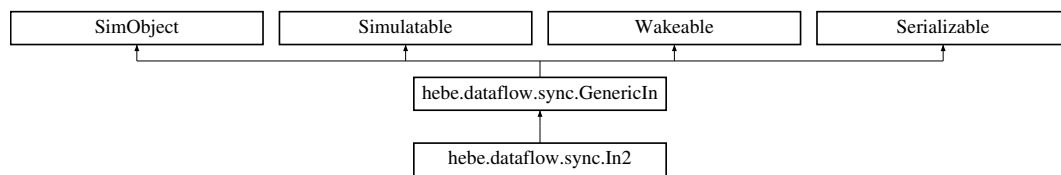
Object Constructor.

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

- `hebe/dataflow/sync/In16.java`

3.45 hebe.dataflow.sync.In2 Class Reference

Inheritance diagram for hebe.dataflow.sync.In2:



Public Member Functions

- [In2\(\)](#)

Additional Inherited Members

3.45.1 Detailed Description

[In2](#) component for the UFV synchronous data flow simulator.

The component implements an input queue with 2 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.45.2 Constructor & Destructor Documentation

3.45.2.1 hebe.dataflow.sync.In2.In2()

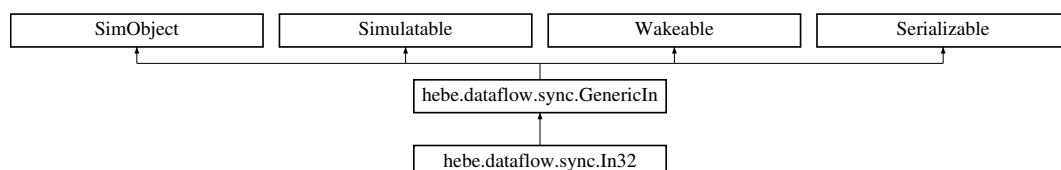
Object Constructor.

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

- hebe/dataflow/sync/In2.java

3.46 hebe.dataflow.sync.In32 Class Reference

Inheritance diagram for hebe.dataflow.sync.In32:



Public Member Functions

- [In32](#) ()

Additional Inherited Members

3.46.1 Detailed Description

[In32](#) component for the UFV synchronous data flow simulator.

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

3.46.2 Constructor & Destructor Documentation

3.46.2.1 `hebe.dataflow.sync.In32.In32 ()`

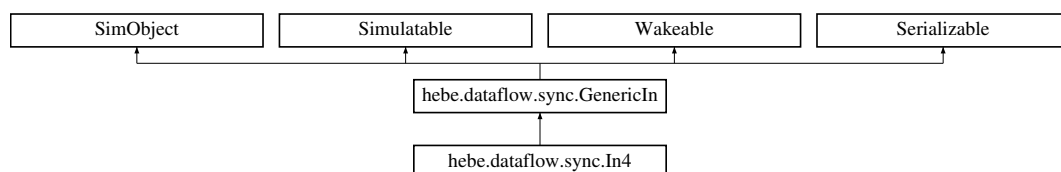
Object Constructor.

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

- `hebe/dataflow/sync/In32.java`

3.47 `hebe.dataflow.sync.In4` Class Reference

Inheritance diagram for `hebe.dataflow.sync.In4`:



Public Member Functions

- [In4](#) ()

Additional Inherited Members

3.47.1 Detailed Description

[In4](#) component for the UFV synchronous data flow simulator.

The component implements an input queue with 4 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.47.2 Constructor & Destructor Documentation**3.47.2.1 hebe.dataflow.sync.In4.In4 ()**

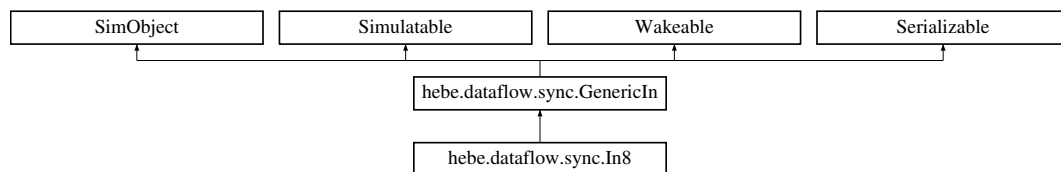
Object Constructor.

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

- hebe/dataflow/sync/In4.java

3.48 hebe.dataflow.sync.In8 Class Reference

Inheritance diagram for hebe.dataflow.sync.In8:

**Public Member Functions**

- [In8 \(\)](#)

Additional Inherited Members**3.48.1 Detailed Description**

[In8](#) component for the UFV synchronous data flow simulator.

The component implements an input queue with 8 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.48.2 Constructor & Destructor Documentation

3.48.2.1 `hebe.dataflow.sync.In8.In8 ()`

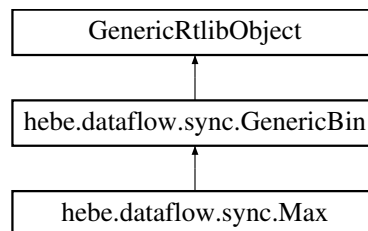
Object Constructor.

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

- `hebe/dataflow/sync/In8.java`

3.49 `hebe.dataflow.sync.Max` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Max`:



Public Member Functions

- `Max ()`
- `int compute (int data1, int data2)`

3.49.1 Detailed Description

`Max` component for the UFV synchronous data flow simulator.

The component is responsible for passing the output to the largest value input.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.49.2 Constructor & Destructor Documentation

3.49.2.1 `hebe.dataflow.sync.Max.Max ()`

Object Constructor.

3.49.3 Member Function Documentation

3.49.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

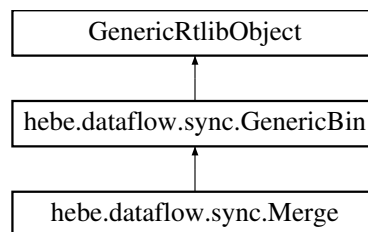
- Returns the result of the computation. In this case, the largest of the parameters.

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

- hebe/dataflow/sync/Max.java

3.50 hebe.dataflow.sync.Merge Class Reference

Inheritance diagram for hebe.dataflow.sync.Merge:

**Public Member Functions**

- [Merge](#) ()
- void [evaluate](#) (Object arg)

3.50.1 Detailed Description

[Merge](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.50.2 Constructor & Destructor Documentation

3.50.2.1 hebe.dataflow.sync.Merge.Merge ()

Object Constructor.

3.50.3 Member Function Documentation

3.50.3.1 void hebe.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

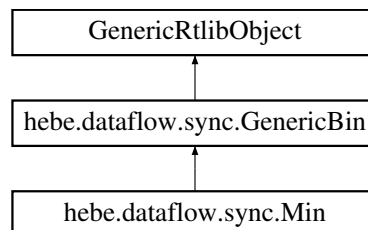
| | |
|------------|------------------------------|
| <i>arg</i> | an arbitrary object argument |
|------------|------------------------------|

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

- hebe/dataflow/sync/Merge.java

3.51 hebe.dataflow.sync.Min Class Reference

Inheritance diagram for hebe.dataflow.sync.Min:



Public Member Functions

- **Min** ()
- int **compute** (int data1, int data2)

3.51.1 Detailed Description

Min component for the UFV synchronous data flow simulator.

The component is responsible for passing the output to the lowest value input.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.51.2 Constructor & Destructor Documentation

3.51.2.1 hebe.dataflow.sync.Min.Min ()

Object Constructor.

3.51.3 Member Function Documentation

3.51.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

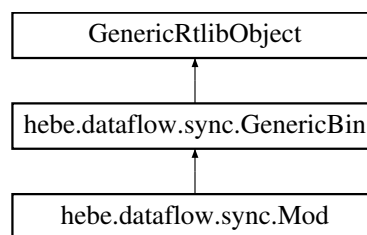
- Returns the result of the computation. In this case, the smallest of the parameters.

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

- `hebe/dataflow/sync/Min.java`

3.52 `hebe.dataflow.sync.Mod` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Mod`:



Public Member Functions

- `Mod ()`
- `int compute (int data1, int data2)`

3.52.1 Detailed Description

`Mod` component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.52.2 Constructor & Destructor Documentation

3.52.2.1 hebe.dataflow.sync.Mod.Mod ()

Object Constructor.

3.52.3 Member Function Documentation

3.52.3.1 int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

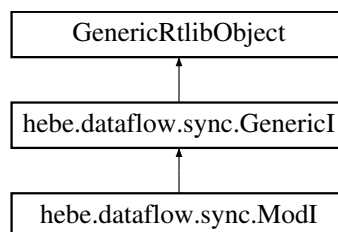
- Returns the result of the computation. In this case, it returns the rest of the division between the parameters.

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

- hebe/dataflow/sync/Mod.java

3.53 hebe.dataflow.sync.ModI Class Reference

Inheritance diagram for hebe.dataflow.sync.ModI:



Public Member Functions

- [ModI](#) ()
- int [compute](#) (int data)

3.53.1 Detailed Description

[ModI](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.53.2 Constructor & Destructor Documentation

3.53.2.1 `hebe.dataflow.sync.Modl.Modl ()`

Object Constructor.

3.53.3 Member Function Documentation

3.53.3.1 `int hebe.dataflow.sync.Modl.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

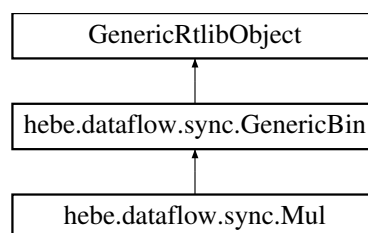
- Returns the result of the computation. In this case, it returns the rest of the division of the parameter by the id.

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

- `hebe/dataflow/sync/Modl.java`

3.54 `hebe.dataflow.sync.Mul` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Mul`:



Public Member Functions

- [Mul \(\)](#)
- `int compute (int data1, int data2)`

3.54.1 Detailed Description

[Mul](#) component for the UFV synchronous data flow simulator.

The component is responsible for multiplying the inputs.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicsa@gmail.com

Version

1.0

3.54.2 Constructor & Destructor Documentation**3.54.2.1 hebe.dataflow.sync.Mul.Mul ()**

Object Constructor.

3.54.3 Member Function Documentation**3.54.3.1 int hebe.dataflow.sync.Mul.compute (int *data1*, int *data2*)**

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

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

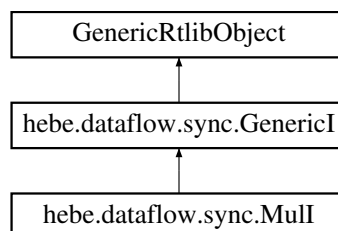
- Returns the result of the computation. In this case the value of the multiplication of the parameters.

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

- hebe/dataflow/sync/Mul.java

3.55 hebe.dataflow.sync.Mul Class Reference

Inheritance diagram for hebe.dataflow.sync.Mul:

**Public Member Functions**

- [Mul \(\)](#)
- int [compute](#) (int data)

3.55.1 Detailed Description

Mull component for the UFV synchronous data flow simulator.

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

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.55.2 Constructor & Destructor Documentation

3.55.2.1 `hebe.dataflow.sync.Mull.Mull ()`

Object Constructor.

3.55.3 Member Function Documentation

3.55.3.1 `int hebe.dataflow.sync.Mull.compute (int data)`

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

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

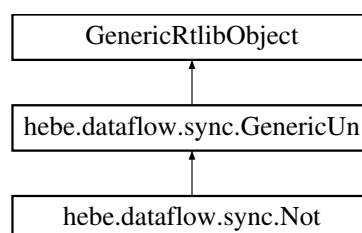
- Returns the result of the computation. In this case the value of the multiplication of the parameter by the id.

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

- `hebe/dataflow/sync/Mull.java`

3.56 `hebe.dataflow.sync.Not` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Not`:



Public Member Functions

- [Not](#) ()
- int [compute](#) (int data)

3.56.1 Detailed Description

[Not](#) component for the UFV synchronous data flow simulator.

The component is responsible for the bitwise inversion of the input.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.56.2 Constructor & Destructor Documentation

3.56.2.1 hebe.dataflow.sync.Not.Not ()

Object Constructor.

3.56.3 Member Function Documentation

3.56.3.1 int hebe.dataflow.sync.Not.compute (int data)

Method responsible for the component computation: in this case performs a bitwise inversion of the parameter.

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

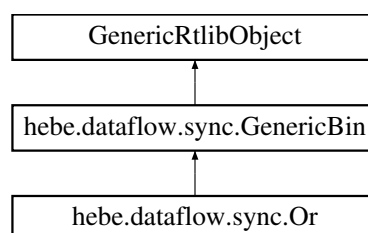
- Returns the result of the computation. In this case the value of the bitwise inversion of the parameter.

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

- hebe/dataflow/sync/Not.java

3.57 hebe.dataflow.sync.Or Class Reference

Inheritance diagram for hebe.dataflow.sync.Or:



Public Member Functions

- [Or](#) ()
- int [compute](#) (int data1, int data2)

3.57.1 Detailed Description

[Or](#) component for the UFV synchronous data flow simulator.

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

3.57.2 Constructor & Destructor Documentation

3.57.2.1 `hebe.dataflow.sync.Or.Or ()`

Object Constructor.

3.57.3 Member Function Documentation

3.57.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

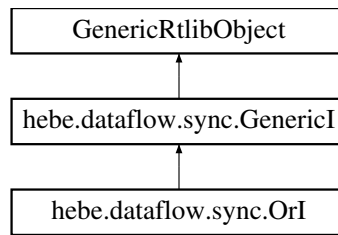
- Returns the result of the computation. In this case the result of the logical operation "Or" between the parameters.

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

- `hebe/dataflow/sync/Or.java`

3.58 `hebe.dataflow.sync.Or` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Or`:



Public Member Functions

- [Orl](#) ()
- int [compute](#) (int data)

3.58.1 Detailed Description

[Orl](#) component for the UFV synchronous data flow simulator.

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

3.58.2 Constructor & Destructor Documentation

3.58.2.1 `hebe.dataflow.sync.Orl.Orl ()`

Object Constructor.

3.58.3 Member Function Documentation

3.58.3.1 `int hebe.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

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

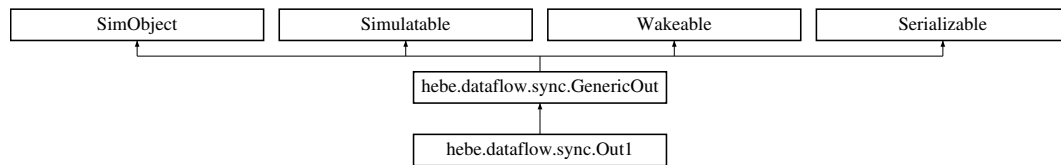
- Returns the result of the computation. In this case the result of the logical operation "OR" between the parameter and the id.

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

- `hebe/dataflow/sync/Orl.java`

3.59 hebe.dataflow.sync.Out1 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out1:



Public Member Functions

- [Out1](#) ()

Additional Inherited Members

3.59.1 Detailed Description

[Out1](#) component for the UFV synchronous data flow simulator.

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

3.59.2 Constructor & Destructor Documentation

3.59.2.1 hebe.dataflow.sync.Out1.Out1 ()

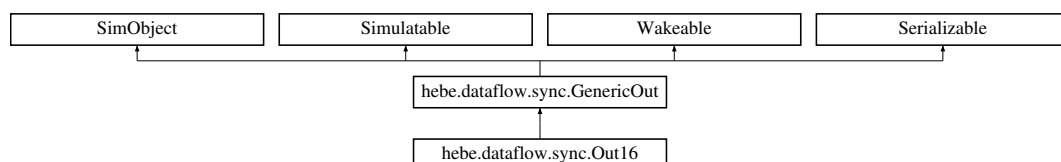
Object Constructor.

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

- hebe/dataflow/sync/Out1.java

3.60 hebe.dataflow.sync.Out16 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out16:



Public Member Functions

- [Out16](#) ()

Additional Inherited Members

3.60.1 Detailed Description

[Out16](#) component for the UFV synchronous data flow simulator.

O implements an input queue with 16 output.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.60.2 Constructor & Destructor Documentation

3.60.2.1 hebe.dataflow.sync.Out16.Out16 ()

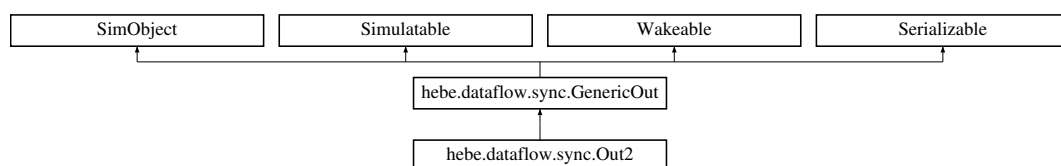
Object Constructor.

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

- hebe/dataflow/sync/Out16.java

3.61 hebe.dataflow.sync.Out2 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out2:



Public Member Functions

- [Out2](#) ()

Additional Inherited Members

3.61.1 Detailed Description

[Out2](#) component for the UFV synchronous data flow simulator.

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 - cacauvicsa@gmail.com

Version

1.0

3.61.2 Constructor & Destructor Documentation**3.61.2.1 hebe.dataflow.sync.Out2.Out2 ()**

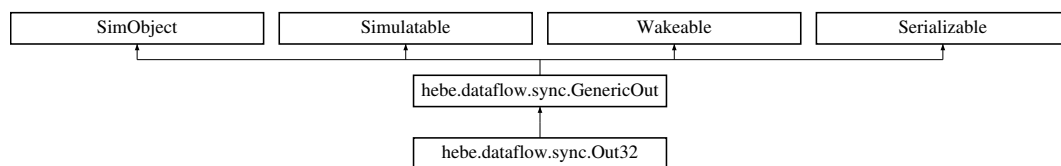
Object Constructor.

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

- hebe/dataflow/sync/Out2.java

3.62 hebe.dataflow.sync.Out32 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out32:

**Public Member Functions**

- [Out32 \(\)](#)

Additional Inherited Members**3.62.1 Detailed Description**

[Out32](#) component for the UFV synchronous data flow simulator.

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 - cacauvicsa@gmail.com

Version

1.0

3.62.2 Constructor & Destructor Documentation

3.62.2.1 hebe.dataflow.sync.Out32.Out32 ()

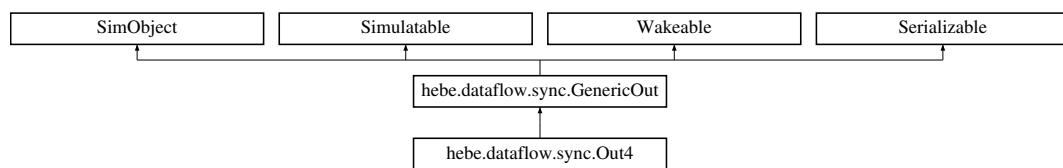
Object Constructor.

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

- hebe/dataflow/sync/Out32.java

3.63 hebe.dataflow.sync.Out4 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out4:



Public Member Functions

- [Out4](#) ()

Additional Inherited Members

3.63.1 Detailed Description

[Out4](#) component for the UfV synchronous data flow simulator.

The component implements an output queue with 4 input.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.63.2 Constructor & Destructor Documentation

3.63.2.1 hebe.dataflow.sync.Out4.Out4 ()

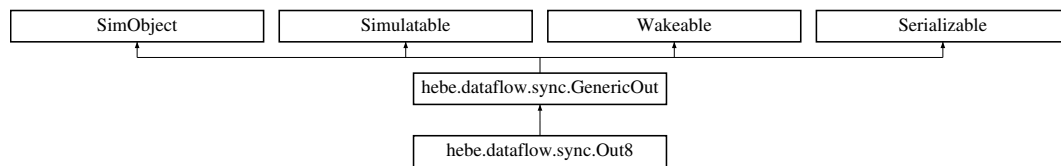
Object Constructor.

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

- hebe/dataflow/sync/Out4.java

3.64 hebe.dataflow.sync.Out8 Class Reference

Inheritance diagram for hebe.dataflow.sync.Out8:



Public Member Functions

- [Out8](#) ()

Additional Inherited Members

3.64.1 Detailed Description

[Out8](#) component for the UFV synchronous data flow simulator.

The component implements an output queue with 8 input.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.64.2 Constructor & Destructor Documentation

3.64.2.1 hebe.dataflow.sync.Out8.Out8 ()

Object Constructor.

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

- hebe/dataflow/sync/Out8.java

3.65 hebe.examples.dataflow_sync.PaethDataflowFPGA Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.65.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/PaethDataflowFPGA.java

3.66 hebe.examples.dataflow_sync.PaethDataflowSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.66.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/PaethDataflowSimulation.java

3.67 hebe.examples.dataflow_sync.Reduce32DataflowFpga Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.67.1 Detailed Description

[HistogramDataflowSimulation](#) example in FPGA Board.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

* 1.0

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

- hebe/examples/dataflow_sync/Reduce32DataflowFpga.java

3.68 hebe.examples.dataflow_sync.Reduce32DataflowSimulation Class Reference

Static Public Member Functions

- static void **main** (String argv[])

3.68.1 Detailed Description

[HistogramDataflowSimulation](#) example in simulator.

Universidade Federal de Viçosa - MG - Brasil.

Author

Jeronimo Costa Penha - jeronimopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

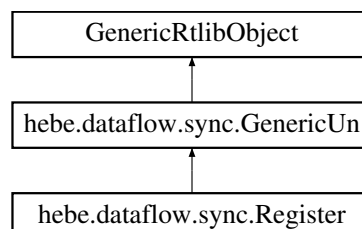
* 1.0

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

- hebe/examples/dataflow_sync/Reduce32DataflowSimulation.java

3.69 hebe.dataflow.sync.Register Class Reference

Inheritance diagram for hebe.dataflow.sync.Register:

**Public Member Functions**

- [Register](#) ()

3.69.1 Detailed Description

[Register](#) component for the UFV synchronous data flow simulator.

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

3.69.2 Constructor & Destructor Documentation

3.69.2.1 `hebe.dataflow.sync.Register.Register ()`

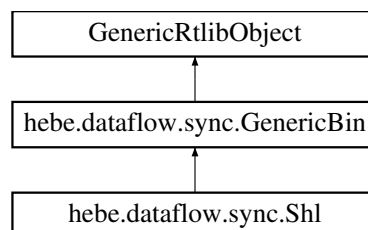
Object Constructor.

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

- `hebe/dataflow/sync/Register.java`

3.70 `hebe.dataflow.sync.Shl` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Shl`:



Public Member Functions

- [Shl](#) ()
- `int` [compute](#) (int data1, int data2)

3.70.1 Detailed Description

[Shl](#) component for the UFV synchronous data flow simulator.

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 - cacauvicsa@gmail.com

Version

1.0

3.70.2 Constructor & Destructor Documentation**3.70.2.1 `hebe.dataflow.sync.Shl.Shl ()`**

Object Constructor.

3.70.3 Member Function Documentation**3.70.3.1 `int hebe.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

| | |
|--------------|-------------------------------------|
| <i>data1</i> | - Value 1 to be used for computing. |
| <i>data2</i> | - Value 2 to be used for computing. |

Returns

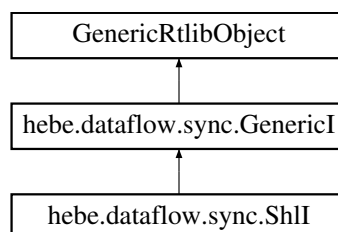
- Returns the result of the computation. In this case, it moves all the bits of the first parameter to the left N times, where N is equal to the value of second parameter.

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

- `hebe/dataflow/sync/Shl.java`

3.71 `hebe.dataflow.sync.ShlI` Class Reference

Inheritance diagram for `hebe.dataflow.sync.ShlI`:

**Public Member Functions**

- [ShlI \(\)](#)
- `int` [compute](#) (`int data`)

3.71.1 Detailed Description

[Shll](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.71.2 Constructor & Destructor Documentation

3.71.2.1 hebe.dataflow.sync.Shll.Shll ()

Object Constructor.

3.71.3 Member Function Documentation

3.71.3.1 int hebe.dataflow.sync.Shll.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
```

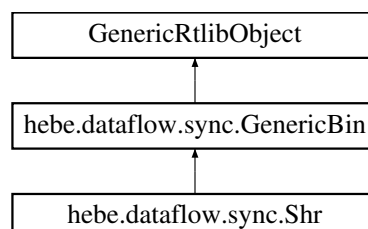
all the bits of the parameter to the left N times, where N is equal to the value of a (immediate) id.

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

- hebe/dataflow/sync/Shll.java

3.72 hebe.dataflow.sync.Shr Class Reference

Inheritance diagram for hebe.dataflow.sync.Shr:



Public Member Functions

- [Shr](#) ()
- int [compute](#) (int data1, int data2)

3.72.1 Detailed Description

Shr component for the UFV synchronous data flow simulator.

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

3.72.2 Constructor & Destructor Documentation

3.72.2.1 `hebe.dataflow.sync.Shr.Shr ()`

Object Constructor.

3.72.3 Member Function Documentation

3.72.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

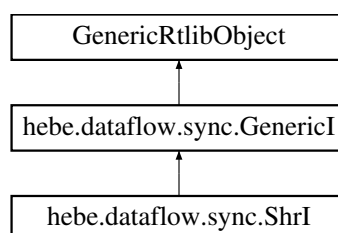
- Returns the result of the computation. In this case, it moves all the bits of the first parameter to the right N times, where N is equal to the value of the second parameter.

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

- `hebe/dataflow/sync/Shr.java`

3.73 `hebe.dataflow.sync.Shrl` Class Reference

Inheritance diagram for `hebe.dataflow.sync.Shrl`:



Public Member Functions

- [Shrl](#) ()
- `int compute` (int data)

3.73.1 Detailed Description

[Shrl](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com

Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.73.2 Constructor & Destructor Documentation

3.73.2.1 `hebe.dataflow.sync.Shrl.Shrl` ()

Object Constructor.

3.73.3 Member Function Documentation

3.73.3.1 `int hebe.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
```

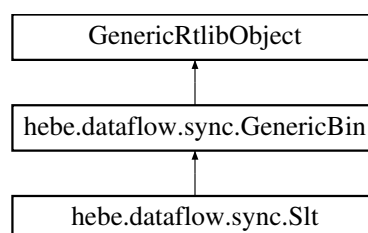
all the bits of the parameter to the right N times, where N is equal to the value of a (immediate) id.

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

- `hebe/dataflow/sync/Shrl.java`

3.74 hebe.dataflow.sync.Slt Class Reference

Inheritance diagram for `hebe.dataflow.sync.Slt`:



Public Member Functions

- [Slt](#) ()
- int [compute](#) (int data1, int data2)

3.74.1 Detailed Description

[Slt](#) component for the UFV synchronous data flow simulator.

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

3.74.2 Constructor & Destructor Documentation

3.74.2.1 `hebe.dataflow.sync.Slt.Slt ()`

Object Constructor.

3.74.3 Member Function Documentation

3.74.3.1 `int hebe.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

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

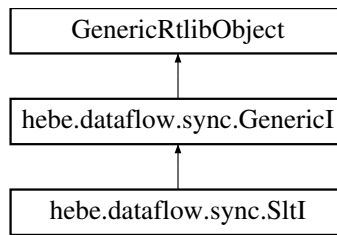
- Returns the result of the computation. In this case 1 or 0 depending on the comparison between the parameters.

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

- `hebe/dataflow/sync/Slt.java`

3.75 `hebe.dataflow.sync.SltI` Class Reference

Inheritance diagram for `hebe.dataflow.sync.SltI`:



Public Member Functions

- [Sltl](#) ()
- int [compute](#) (int data)

3.75.1 Detailed Description

[Sltl](#) component for the UFV synchronous data flow simulator.

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 - jeronomopenha@gmail.com
 Ricardo Santos Ferreira - cacauvicosa@gmail.com

Version

1.0

3.75.2 Constructor & Destructor Documentation

3.75.2.1 `hebe.dataflow.sync.Sltl.Sltl ()`

Object Constructor.

3.75.3 Member Function Documentation

3.75.3.1 `int hebe.dataflow.sync.Sltl.compute (int data)`

Method responsible for the component computation: in this case performs a comparison if parameter is less than the (immediate) id.

Parameters

| | |
|-------------|-----------------------------------|
| <i>data</i> | - Value to be used for computing. |
|-------------|-----------------------------------|

Returns

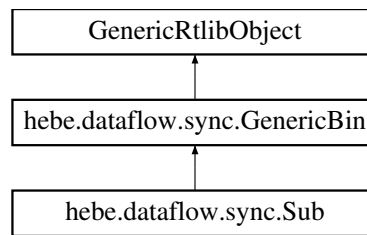
- Returns the result of the computation. In this case 1 or 0 depending on the comparison between the parameter and the id.

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

- `hebe/dataflow/sync/Sltl.java`

3.76 hebe.dataflow.sync.Sub Class Reference

Inheritance diagram for hebe.dataflow.sync.Sub:



Public Member Functions

- [Sub](#) ()
- int [compute](#) (int data1, int data2)

3.76.1 Detailed Description

[Sub](#) component for the UFV synchronous data flow simulator.

The component is responsible for subtracting the inputs.

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.76.2 Constructor & Destructor Documentation

3.76.2.1 hebe.dataflow.sync.Sub.Sub ()

Object Constructor.

3.76.3 Member Function Documentation

3.76.3.1 int hebe.dataflow.sync.Sub.compute (int *data1*, int *data2*)

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

Parameters

| | |
|--------------|--|
| <i>data1</i> | - Value to be used for the computation related to input 1. |
| <i>data2</i> | - Value to be used for the computation related to input 2. |

Returns

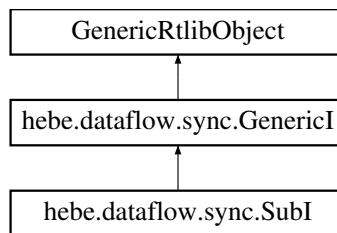
- Returns the result of the computation. In this case the value of the subtraction of the parameters.

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

- hebe/dataflow/sync/Sub.java

3.77 hebe.dataflow.sync.SubI Class Reference

Inheritance diagram for hebe.dataflow.sync.SubI:

**Public Member Functions**

- [SubI](#) ()
- int [compute](#) (int data)

3.77.1 Detailed Description

[SubI](#) component for the UFV synchronous data flow simulator.

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

Universidade Federal de Viçosa - MG - Brasil.

Author

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

Version

1.0

3.77.2 Constructor & Destructor Documentation

3.77.2.1 hebe.dataflow.sync.SubI.SubI ()

Object Constructor.

3.77.3 Member Function Documentation

3.77.3.1 int hebe.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:

- [hebe/dataflow/sync/Subl.java](#)

Index

Abs
 hebe::dataflow::sync::Abs, 5

AccAdd
 hebe::dataflow::sync::AccAdd, 7

AccMax
 hebe::dataflow::sync::AccMax, 7

AccMin
 hebe::dataflow::sync::AccMin, 9

AccMul
 hebe::dataflow::sync::AccMul, 10

accumulate
 hebe::dataflow::sync::AccMax, 8
 hebe::dataflow::sync::AccMin, 9
 hebe::dataflow::sync::AccMul, 10
 hebe::dataflow::sync::GenericAcc, 31

Add
 hebe::dataflow::sync::Add, 11

Addl
 hebe::dataflow::sync::Addl, 13

And
 hebe::dataflow::sync::And, 15

Andl
 hebe::dataflow::sync::Andl, 16

Beq
 hebe::dataflow::sync::Beq, 17

Beql
 hebe::dataflow::sync::Beql, 18

Bne
 hebe::dataflow::sync::Bne, 19

Bnel
 hebe::dataflow::sync::Bnel, 21

compute
 hebe::dataflow::sync::Abs, 6
 hebe::dataflow::sync::Add, 11
 hebe::dataflow::sync::Addl, 13
 hebe::dataflow::sync::And, 15
 hebe::dataflow::sync::Andl, 16
 hebe::dataflow::sync::Beq, 17
 hebe::dataflow::sync::Beql, 18
 hebe::dataflow::sync::Bne, 20
 hebe::dataflow::sync::Bnel, 21
 hebe::dataflow::sync::Div, 26
 hebe::dataflow::sync::Divl, 27
 hebe::dataflow::sync::GenericBin, 34
 hebe::dataflow::sync::GenericBranch, 40
 hebe::dataflow::sync::GenericBranchl, 47
 hebe::dataflow::sync::Genericl, 56
 hebe::dataflow::sync::GenericUn, 86
 hebe::dataflow::sync::Histogram, 92
 hebe::dataflow::sync::Max, 100
 hebe::dataflow::sync::Min, 104
 hebe::dataflow::sync::Mod, 105
 hebe::dataflow::sync::Modl, 106
 hebe::dataflow::sync::Mul, 107
 hebe::dataflow::sync::Null, 108
 hebe::dataflow::sync::Not, 109
 hebe::dataflow::sync::Or, 110
 hebe::dataflow::sync::Orl, 111
 hebe::dataflow::sync::Shl, 120
 hebe::dataflow::sync::Shll, 121
 hebe::dataflow::sync::Shr, 122
 hebe::dataflow::sync::Shrl, 123
 hebe::dataflow::sync::Slt, 124
 hebe::dataflow::sync::Slll, 125
 hebe::dataflow::sync::Sub, 126
 hebe::dataflow::sync::Subl, 127

constructDynamicSymbol
 hebe::dataflow::sync::GenericBin, 34
 hebe::dataflow::sync::GenericBranch, 40
 hebe::dataflow::sync::GenericBranchl, 47
 hebe::dataflow::sync::Genericl, 57
 hebe::dataflow::sync::GenericIn, 66
 hebe::dataflow::sync::GenericOut, 76
 hebe::dataflow::sync::GenericUn, 86

constructPorts
 hebe::dataflow::sync::GenericBin, 34
 hebe::dataflow::sync::GenericBranch, 40
 hebe::dataflow::sync::GenericBranchl, 47
 hebe::dataflow::sync::Genericl, 57
 hebe::dataflow::sync::GenericIn, 66
 hebe::dataflow::sync::GenericOut, 76
 hebe::dataflow::sync::GenericUn, 86

constructStandardValues
 hebe::dataflow::sync::GenericIn, 66
 hebe::dataflow::sync::GenericOut, 76

Div
 hebe::dataflow::sync::Div, 26

Divl
 hebe::dataflow::sync::Divl, 27

evaluate
 hebe::dataflow::sync::GenericAcc, 31
 hebe::dataflow::sync::GenericBin, 34
 hebe::dataflow::sync::GenericBranch, 41
 hebe::dataflow::sync::GenericBranchl, 47
 hebe::dataflow::sync::Genericl, 57
 hebe::dataflow::sync::GenericIn, 66

- hebe::dataflow::sync::GenericOut, [76](#)
 - hebe::dataflow::sync::GenericUn, [86](#)
 - hebe::dataflow::sync::Histogram, [93](#)
 - hebe::dataflow::sync::Merge, [103](#)
- execFpga
 - hebe::dataflow::DataflowSyncSimulBase, [23](#)
- execHades
 - hebe::dataflow::DataflowSyncSimulBase, [23](#)
- GenericAcc
 - hebe::dataflow::sync::GenericAcc, [31](#)
- GenericBin
 - hebe::dataflow::sync::GenericBin, [34](#)
- GenericBranch
 - hebe::dataflow::sync::GenericBranch, [40](#)
- GenericBranchI
 - hebe::dataflow::sync::GenericBranchI, [47](#)
- GenericI
 - hebe::dataflow::sync::GenericI, [56](#)
- GenericIn
 - hebe::dataflow::sync::GenericIn, [66](#)
- GenericOut
 - hebe::dataflow::sync::GenericOut, [76](#)
- GenericUn
 - hebe::dataflow::sync::GenericUn, [86](#)
- getAcc
 - hebe::dataflow::sync::GenericAcc, [31](#)
- getBackground
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [57](#)
 - hebe::dataflow::sync::GenericUn, [86](#)
- GetComponentId
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [57](#)
- GetComponentImmediate
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [57](#)
- GetComponentType
 - hebe::dataflow::sync::GenericBin, [34](#)
 - hebe::dataflow::sync::GenericBranch, [41](#)
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [57](#)
 - hebe::dataflow::sync::GenericIn, [66](#)
 - hebe::dataflow::sync::GenericOut, [76](#)
 - hebe::dataflow::sync::GenericUn, [86](#)
- getCounter
 - hebe::dataflow::sync::GenericAcc, [31](#)
 - hebe::dataflow::sync::Histogram, [93](#)
- getDecr
 - hebe::dataflow::sync::Histogram, [93](#)
- getDefaultdelay
 - hebe::dataflow::sync::GenericIn, [66](#)
 - hebe::dataflow::sync::GenericOut, [76](#)
- getDelay
 - hebe::dataflow::sync::GenericIn, [66](#)
 - hebe::dataflow::sync::GenericOut, [76](#)
- getDoneSignal
 - hebe::dataflow::sync::GenericOut, [77](#)
- getHistogram
 - hebe::dataflow::sync::Histogram, [93](#)
- getId
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [57](#)
- getIdxDin
 - hebe::dataflow::sync::GenericIn, [67](#)
- getIdxDout
 - hebe::dataflow::sync::GenericOut, [77](#)
- getImmediate
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [58](#)
- getLabel_nome
 - hebe::dataflow::sync::GenericBranch, [41](#)
- getLabelFormatter
 - hebe::dataflow::sync::GenericIn, [67](#)
 - hebe::dataflow::sync::GenericOut, [77](#)
- getLabelNome
 - hebe::dataflow::sync::GenericBin, [34](#)
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [58](#)
 - hebe::dataflow::sync::GenericUn, [86](#)
- getN_bits
 - hebe::dataflow::sync::GenericIn, [67](#)
 - hebe::dataflow::sync::GenericOut, [77](#)
- getNUMBITS
 - hebe::dataflow::sync::Histogram, [93](#)
- getPortClk
 - hebe::dataflow::sync::GenericBin, [35](#)
 - hebe::dataflow::sync::GenericBranch, [41](#)
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [58](#)
 - hebe::dataflow::sync::GenericIn, [67](#)
 - hebe::dataflow::sync::GenericOut, [77](#)
 - hebe::dataflow::sync::GenericUn, [87](#)
- getPortDconf
 - hebe::dataflow::sync::GenericBranchI, [48](#)
 - hebe::dataflow::sync::GenericI, [58](#)
 - hebe::dataflow::sync::GenericIn, [67](#)
- getPortDin
 - hebe::dataflow::sync::GenericBranchI, [49](#)
 - hebe::dataflow::sync::GenericI, [58](#)
 - hebe::dataflow::sync::GenericOut, [77](#)
 - hebe::dataflow::sync::GenericUn, [87](#)
- getPortDin1
 - hebe::dataflow::sync::GenericBin, [35](#)
 - hebe::dataflow::sync::GenericBranch, [41](#)
- getPortDin2
 - hebe::dataflow::sync::GenericBin, [35](#)
 - hebe::dataflow::sync::GenericBranch, [41](#)
- getPortDout
 - hebe::dataflow::sync::GenericBin, [35](#)
 - hebe::dataflow::sync::GenericI, [58](#)
 - hebe::dataflow::sync::GenericIn, [67](#)
 - hebe::dataflow::sync::GenericUn, [87](#)
- getPortElse
 - hebe::dataflow::sync::GenericBranch, [41](#)
 - hebe::dataflow::sync::GenericBranchI, [49](#)
- getPortEn

- hebe::dataflow::sync::GenericBin, 35
- hebe::dataflow::sync::GenericBranch, 41
- hebe::dataflow::sync::GenericBranchI, 49
- hebe::dataflow::sync::GenericI, 58
- hebe::dataflow::sync::GenericOut, 77
- hebe::dataflow::sync::GenericUn, 87
- getPortEnOut
 - hebe::dataflow::sync::GenericIn, 67
- getPortIf
 - hebe::dataflow::sync::GenericBranch, 42
 - hebe::dataflow::sync::GenericBranchI, 49
- getPortRdy
 - hebe::dataflow::sync::GenericIn, 67
 - hebe::dataflow::sync::GenericOut, 77
- getPortRin
 - hebe::dataflow::sync::GenericBranchI, 49
 - hebe::dataflow::sync::GenericI, 58
 - hebe::dataflow::sync::GenericOut, 78
 - hebe::dataflow::sync::GenericUn, 87
- getPortRin1
 - hebe::dataflow::sync::GenericBin, 35
 - hebe::dataflow::sync::GenericBranch, 42
- getPortRin2
 - hebe::dataflow::sync::GenericBin, 35
 - hebe::dataflow::sync::GenericBranch, 42
- getPortRout
 - hebe::dataflow::sync::GenericBin, 35
 - hebe::dataflow::sync::GenericI, 58
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericUn, 87
- getPortRst
 - hebe::dataflow::sync::GenericBin, 35
 - hebe::dataflow::sync::GenericBranch, 42
 - hebe::dataflow::sync::GenericBranchI, 49
 - hebe::dataflow::sync::GenericI, 59
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
 - hebe::dataflow::sync::GenericUn, 87
- getQTDE_PORTS
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
- getQtdeSave
 - hebe::dataflow::sync::GenericOut, 78
- getS
 - hebe::dataflow::sync::GenericBin, 36
 - hebe::dataflow::sync::GenericBranch, 42
 - hebe::dataflow::sync::GenericUn, 87
- getStringLabel
 - hebe::dataflow::sync::GenericBin, 36
 - hebe::dataflow::sync::GenericBranch, 42
 - hebe::dataflow::sync::GenericUn, 87
- getStringLabelId
 - hebe::dataflow::sync::GenericBranchI, 49
 - hebe::dataflow::sync::GenericI, 59
- getStringLabelImmediate
 - hebe::dataflow::sync::GenericBranchI, 49
 - hebe::dataflow::sync::GenericI, 59
- getTOT_PORTS
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
- getTamVectorOut
 - hebe::dataflow::sync::GenericOut, 78
- getValueLabel
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
- getVector
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
- getVector_000
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 78
- getVector_111
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 79
- getVector_UUU
 - hebe::dataflow::sync::GenericIn, 68
 - hebe::dataflow::sync::GenericOut, 79
- getVector_XXX
 - hebe::dataflow::sync::GenericIn, 69
 - hebe::dataflow::sync::GenericOut, 79
- getVector_ZZZ
 - hebe::dataflow::sync::GenericIn, 69
 - hebe::dataflow::sync::GenericOut, 79
- getVectorIn
 - hebe::dataflow::sync::GenericIn, 69
- getVectorOut
 - hebe::dataflow::sync::GenericOut, 79
- getVectorOutputPort
 - hebe::dataflow::sync::GenericIn, 69
 - hebe::dataflow::sync::GenericOut, 79
- hebe.dataflow.AFU, 14
- hebe.dataflow.DataflowSyncSimulBase, 23
- hebe.dataflow.sync.Abs, 5
- hebe.dataflow.sync.AccAdd, 6
- hebe.dataflow.sync.AccMax, 7
- hebe.dataflow.sync.AccMin, 8
- hebe.dataflow.sync.AccMul, 9
- hebe.dataflow.sync.Add, 10
- hebe.dataflow.sync.AddI, 12
- hebe.dataflow.sync.And, 14
- hebe.dataflow.sync.AndI, 15
- hebe.dataflow.sync.Beq, 16
- hebe.dataflow.sync.BeqI, 18
- hebe.dataflow.sync.Bne, 19
- hebe.dataflow.sync.BneI, 20
- hebe.dataflow.sync.Div, 25
- hebe.dataflow.sync.DivI, 26
- hebe.dataflow.sync.GenericAcc, 30
- hebe.dataflow.sync.GenericBin, 32
- hebe.dataflow.sync.GenericBranch, 39
- hebe.dataflow.sync.GenericBranchI, 45
- hebe.dataflow.sync.GenericI, 54
- hebe.dataflow.sync.GenericIn, 64
- hebe.dataflow.sync.GenericOut, 74
- hebe.dataflow.sync.GenericUn, 84
- hebe.dataflow.sync.Histogram, 92

- hebe.dataflow.sync.In1, [95](#)
- hebe.dataflow.sync.In16, [96](#)
- hebe.dataflow.sync.In2, [97](#)
- hebe.dataflow.sync.In32, [97](#)
- hebe.dataflow.sync.In4, [98](#)
- hebe.dataflow.sync.In8, [99](#)
- hebe.dataflow.sync.Max, [100](#)
- hebe.dataflow.sync.Merge, [102](#)
- hebe.dataflow.sync.Min, [103](#)
- hebe.dataflow.sync.Mod, [104](#)
- hebe.dataflow.sync.Mod1, [105](#)
- hebe.dataflow.sync.Mul, [106](#)
- hebe.dataflow.sync.Mull, [107](#)
- hebe.dataflow.sync.Not, [108](#)
- hebe.dataflow.sync.Or, [109](#)
- hebe.dataflow.sync.Ord, [110](#)
- hebe.dataflow.sync.Out1, [112](#)
- hebe.dataflow.sync.Out16, [112](#)
- hebe.dataflow.sync.Out2, [113](#)
- hebe.dataflow.sync.Out32, [114](#)
- hebe.dataflow.sync.Out4, [115](#)
- hebe.dataflow.sync.Out8, [116](#)
- hebe.dataflow.sync.Register, [118](#)
- hebe.dataflow.sync.Shl, [119](#)
- hebe.dataflow.sync.Shll, [120](#)
- hebe.dataflow.sync.Shr, [121](#)
- hebe.dataflow.sync.Shrl, [122](#)
- hebe.dataflow.sync.Slt, [123](#)
- hebe.dataflow.sync.Sltl, [124](#)
- hebe.dataflow.sync.Sub, [126](#)
- hebe.dataflow.sync.Subl, [127](#)
- hebe.examples.dataflow_sync.AddAbHelloWorld-
DataflowHadesSimulation, [11](#)
- hebe.examples.dataflow_sync.AddAbHelloWorld-
DataflowHadesSimulationWithGeneratedHds,
[12](#)
- hebe.examples.dataflow_sync.AddiDataflowFpga-
Simulation, [13](#)
- hebe.examples.dataflow_sync.BranchTestDataflow-
HadesSimulation, [21](#)
- hebe.examples.dataflow_sync.BranchTestDataflow-
HadesSimulationWithGeneratedHds, [22](#)
- hebe.examples.dataflow_sync.Fir16DataflowFPGA, [27](#)
- hebe.examples.dataflow_sync.Fir16DataflowSimulation,
[28](#)
- hebe.examples.dataflow_sync.Fir4DataflowHades-
Simulation, [28](#)
- hebe.examples.dataflow_sync.Fir4DataflowHades-
SimulationWithGeneratedHds, [29](#)
- hebe.examples.dataflow_sync.Fir8DataflowFPGA, [29](#)
- hebe.examples.dataflow_sync.Fir8DataflowSimulation,
[30](#)
- hebe.examples.dataflow_sync.GourandDataflowFPGA,
[91](#)
- hebe.examples.dataflow_sync.GourandDataflow-
Simulation, [91](#)
- hebe.examples.dataflow_sync.HistogramDataflowFpga-
Simulation, [94](#)
- hebe.examples.dataflow_sync.HistogramDataflow-
Simulation, [94](#)
- hebe.examples.dataflow_sync.PaethDataflowFPGA,
[116](#)
- hebe.examples.dataflow_sync.PaethDataflowSimulation,
[117](#)
- hebe.examples.dataflow_sync.Reduce32DataflowFpga,
[117](#)
- hebe.examples.dataflow_sync.Reduce32Dataflow-
Simulation, [118](#)
- hebe.util.ConfReader, [22](#)
- hebe::dataflow::DataflowSyncSimulBase
 - execFpga, [23](#)
 - execHades, [23](#)
 - startFpgaJtag, [24](#)
 - startSimulation, [24](#), [25](#)
- hebe::dataflow::sync::Abs
 - Abs, [5](#)
 - compute, [6](#)
- hebe::dataflow::sync::AccAdd
 - AccAdd, [7](#)
- hebe::dataflow::sync::AccMax
 - AccMax, [7](#)
 - accumulate, [8](#)
 - reset, [8](#)
- hebe::dataflow::sync::AccMin
 - AccMin, [9](#)
 - accumulate, [9](#)
 - reset, [9](#)
- hebe::dataflow::sync::AccMul
 - AccMul, [10](#)
 - accumulate, [10](#)
 - reset, [10](#)
- hebe::dataflow::sync::Add
 - Add, [11](#)
 - compute, [11](#)
- hebe::dataflow::sync::Addl
 - Addl, [13](#)
 - compute, [13](#)
- hebe::dataflow::sync::And
 - And, [15](#)
 - compute, [15](#)
- hebe::dataflow::sync::Andl
 - Andl, [16](#)
 - compute, [16](#)
- hebe::dataflow::sync::Beq
 - Beq, [17](#)
 - compute, [17](#)
- hebe::dataflow::sync::Beql
 - Beql, [18](#)
 - compute, [18](#)
- hebe::dataflow::sync::Bne
 - Bne, [19](#)
 - compute, [20](#)
- hebe::dataflow::sync::Bnel
 - Bnel, [21](#)
 - compute, [21](#)
- hebe::dataflow::sync::Div

- compute, 26
- Div, 26
- hebe::dataflow::sync::Divl
 - compute, 27
 - Divl, 27
- hebe::dataflow::sync::GenericAcc
 - accumulate, 31
 - evaluate, 31
 - GenericAcc, 31
 - getAcc, 31
 - getCounter, 31
 - reset, 31
 - setAcc, 32
 - setCounter, 32
- hebe::dataflow::sync::GenericBin
 - compute, 34
 - constructDynamicSymbol, 34
 - constructPorts, 34
 - evaluate, 34
 - GenericBin, 34
 - getComponentType, 34
 - getLabelNome, 34
 - getPortClk, 35
 - getPortDin1, 35
 - getPortDin2, 35
 - getPortDout, 35
 - getPortEn, 35
 - getPortRin1, 35
 - getPortRin2, 35
 - getPortRout, 35
 - getPortRst, 35
 - getS, 36
 - getStringLabel, 36
 - initialize, 36
 - needsDynamicSymbol, 36
 - notCompute, 36
 - reseted, 36
 - setCompName, 36
 - setComponentType, 36
 - setLabelNome, 37
 - setPortClk, 37
 - setPortDin1, 37
 - setPortDin2, 37
 - setPortDout, 37
 - setPortEn, 37
 - setPortRin1, 37
 - setPortRin2, 37
 - setPortRout, 38
 - setPortRst, 38
 - setS, 38
 - setString, 38
 - setLabel, 38
 - setSymbol, 38
 - tickDown, 38
 - tickUp, 38
 - write, 38
- hebe::dataflow::sync::GenericBranch
 - compute, 40
- constructDynamicSymbol, 40
- constructPorts, 40
- evaluate, 41
- GenericBranch, 40
- getComponentType, 41
- getLabel_nome, 41
- getPortClk, 41
- getPortDin1, 41
- getPortDin2, 41
- getPortElse, 41
- getPortEn, 41
- getPortIf, 42
- getPortRin1, 42
- getPortRin2, 42
- getPortRst, 42
- getS, 42
- getStringLabel, 42
- initialize, 42
- needsDynamicSymbol, 42
- reseted, 43
- setCompName, 43
- setComponentType, 43
- setLabel_nome, 43
- setPortClk, 43
- setPortDin1, 43
- setPortDin2, 43
- setPortElse, 44
- setPortEn, 44
- setPortIf, 44
- setPortRin1, 44
- setPortRin2, 44
- setPortRst, 44
- setS, 44
- setString, 44
- setLabel, 45
- setSymbol, 45
- tickDown, 45
- tickUp, 45
- write, 45
- hebe::dataflow::sync::GenericBranchl
 - compute, 47
 - constructDynamicSymbol, 47
 - constructPorts, 47
 - evaluate, 47
 - GenericBranchl, 47
 - getBackground, 48
 - getComponentId, 48
 - getComponentImmediate, 48
 - getComponentType, 48
 - getId, 48
 - getImmediate, 48
 - getLabelNome, 48
 - getPortClk, 48
 - getPortDconf, 48
 - getPortDin, 49
 - getPortElse, 49
 - getPortEn, 49
 - getPortIf, 49

- getPortRin, [49](#)
- getPortRst, [49](#)
- getStringLabelId, [49](#)
- getStringLabelImmediate, [49](#)
- initialize, [49](#)
- mousePressed, [51](#)
- needsDynamicSymbol, [51](#)
- reseted, [51](#)
- setBackground, [51](#)
- setCompName, [51](#)
- setComponentId, [51](#)
- setComponentImmediate, [52](#)
- setComponentType, [52](#)
- setId, [52](#)
- setImmediate, [52](#)
- setLabelNome, [52](#)
- setPortClk, [52](#)
- setPortDconf, [52](#)
- setPortDin, [52](#)
- setPortElse, [53](#)
- setPortEn, [53](#)
- setPortIf, [53](#)
- setPortRin, [53](#)
- setPortRst, [53](#)
- setString, [53](#)
- getStringLabelId, [53](#)
- getStringLabelImmediate, [53](#)
- setSymbol, [54](#)
- tickDown, [54](#)
- tickUp, [54](#)
- write, [54](#)
- hebe::dataflow::sync::GenericI
 - compute, [56](#)
 - constructDynamicSymbol, [57](#)
 - constructPorts, [57](#)
 - evaluate, [57](#)
 - GenericI, [56](#)
 - getBackground, [57](#)
 - getComponentId, [57](#)
 - getComponentImmediate, [57](#)
 - getComponentType, [57](#)
 - getId, [57](#)
 - getImmediate, [58](#)
 - setLabelNome, [58](#)
 - setPortClk, [58](#)
 - setPortDconf, [58](#)
 - setPortDin, [58](#)
 - setPortDout, [58](#)
 - setPortEn, [58](#)
 - setPortRin, [58](#)
 - setPortRout, [58](#)
 - setPortRst, [59](#)
 - getStringLabelId, [59](#)
 - getStringLabelImmediate, [59](#)
 - initialize, [59](#)
 - mousePressed, [59](#)
 - needsDynamicSymbol, [59](#)
 - notCompute, [59](#)
 - reset, [60](#)
 - setBackground, [60](#)
 - setCompName, [60](#)
 - setComponentId, [60](#)
 - setComponentImmediate, [60](#)
 - setComponentType, [60](#)
 - setId, [60](#)
 - setImmediate, [60](#)
 - setLabelNome, [61](#)
 - setPortClk, [61](#)
 - setPortDconf, [61](#)
 - setPortDin, [61](#)
 - setPortDout, [61](#)
 - setPortEn, [61](#)
 - setPortRin, [61](#)
 - setPortRout, [61](#)
 - setPortRst, [62](#)
 - setString, [62](#)
 - getStringLabelId, [62](#)
 - getStringLabelImmediate, [62](#)
 - setSymbol, [62](#)
 - tickDown, [62](#)
 - tickUp, [62](#)
 - write, [62](#)
- hebe::dataflow::sync::GenericIn
 - constructDynamicSymbol, [66](#)
 - constructPorts, [66](#)
 - constructStandardValues, [66](#)
 - evaluate, [66](#)
 - GenericIn, [66](#)
 - getComponentType, [66](#)
 - getDefaultDelay, [66](#)
 - getDelay, [66](#)
 - getIdxDin, [67](#)
 - getLabelFormatter, [67](#)
 - getN_bits, [67](#)
 - setPortClk, [67](#)
 - setPortDconf, [67](#)
 - setPortDout, [67](#)
 - setPortEnOut, [67](#)
 - setPortRdy, [67](#)
 - setPortRout, [68](#)
 - setPortRst, [68](#)
 - getQTDE_PORTS, [68](#)
 - getTOT_PORTS, [68](#)
 - getValueLabel, [68](#)
 - getVector, [68](#)
 - getVector_000, [68](#)
 - getVector_111, [68](#)
 - getVector_UUU, [68](#)
 - getVector_XXX, [69](#)
 - getVector_ZZZ, [69](#)
 - getVectorIn, [69](#)
 - getVectorOutputPort, [69](#)
 - initialize, [69](#)
 - isEnabledAnimationFlag, [69](#)
 - isStart, [69](#)
 - needsDynamicSymbol, [69](#)

- setCompName, [70](#)
- setComponentType, [70](#)
- setDefaultdelay, [70](#)
- setDelay, [70](#)
- setEnableAnimationFlag, [70](#)
- setIdxDin, [71](#)
- setLabelFormatter, [71](#)
- setN_bits, [71](#)
- setPortClk, [71](#)
- setPortDconf, [71](#)
- setPortDout, [71](#)
- setPortEnOut, [71](#)
- setPortRdy, [71](#)
- setPortRout, [72](#)
- setPortRst, [72](#)
- setStart, [72](#)
- setValueLabel, [72](#)
- setVector, [72](#)
- setVector_000, [72](#)
- setVector_111, [72](#)
- setVector_UUU, [72](#)
- setVector_XXX, [73](#)
- setVector_ZZZ, [73](#)
- setVectorIn, [73](#)
- setVectorOutputPort, [73](#)
- updateSymbol, [73](#)
- wakeup, [73](#)
- write, [73](#)
- hebe::dataflow::sync::GenericOut
 - constructDynamicSymbol, [76](#)
 - constructPorts, [76](#)
 - constructStandardValues, [76](#)
 - evaluate, [76](#)
 - GenericOut, [76](#)
 - getComponentType, [76](#)
 - getDefaultdelay, [76](#)
 - getDelay, [76](#)
 - getDoneSignal, [77](#)
 - getIdxDout, [77](#)
 - setLabelFormatter, [77](#)
 - getN_bits, [77](#)
 - getPortClk, [77](#)
 - getPortDin, [77](#)
 - getPortEn, [77](#)
 - getPortRdy, [77](#)
 - getPortRin, [78](#)
 - getPortRst, [78](#)
 - getQTDE_PORTS, [78](#)
 - getQtdeSave, [78](#)
 - getTOT_PORTS, [78](#)
 - getTamVectorOut, [78](#)
 - setValueLabel, [78](#)
 - getVector, [78](#)
 - getVector_000, [78](#)
 - getVector_111, [79](#)
 - getVector_UUU, [79](#)
 - getVector_XXX, [79](#)
 - getVector_ZZZ, [79](#)
 - getVectorOut, [79](#)
 - getVectorOutputPort, [79](#)
 - initialize, [79](#)
 - isDone, [80](#)
 - isEnableAnimationFlag, [80](#)
 - needsDynamicSymbol, [80](#)
 - setCompName, [80](#)
 - setComponentType, [80](#)
 - setDefaultdelay, [80](#)
 - setDelay, [80](#), [81](#)
 - setDone, [81](#)
 - setEnableAnimationFlag, [81](#)
 - setIdxDout, [81](#)
 - setLabelFormatter, [81](#)
 - setN_bits, [81](#)
 - setPortClk, [81](#)
 - setPortDin, [82](#)
 - setPortEn, [82](#)
 - setPortRdy, [82](#)
 - setPortRin, [82](#)
 - setPortRst, [82](#)
 - setQtdeSave, [82](#)
 - setTamVectorOut, [82](#)
 - setValueLabel, [82](#)
 - setVector, [83](#)
 - setVector_000, [83](#)
 - setVector_111, [83](#)
 - setVector_UUU, [83](#)
 - setVector_XXX, [83](#)
 - setVector_ZZZ, [83](#)
 - setVectorOut, [83](#)
 - setVectorOutputPort, [84](#)
 - updateSymbol, [84](#)
 - wakeup, [84](#)
 - write, [84](#)
- hebe::dataflow::sync::GenericUn
 - compute, [86](#)
 - constructDynamicSymbol, [86](#)
 - constructPorts, [86](#)
 - evaluate, [86](#)
 - GenericUn, [86](#)
 - getBackground, [86](#)
 - getComponentType, [86](#)
 - getLabelNome, [86](#)
 - getPortClk, [87](#)
 - getPortDin, [87](#)
 - getPortDout, [87](#)
 - getPortEn, [87](#)
 - getPortRin, [87](#)
 - getPortRout, [87](#)
 - getPortRst, [87](#)
 - getS, [87](#)
 - getStringLabel, [87](#)
 - initialize, [88](#)
 - needsDynamicSymbol, [88](#)
 - notCompute, [88](#)
 - reseted, [88](#)
 - setBackground, [88](#)

- setCompName, [88](#)
- setComponentType, [88](#)
- setLabelNome, [89](#)
- setPortClk, [89](#)
- setPortDin, [89](#)
- setPortDout, [89](#)
- setPortEn, [89](#)
- setPortRin, [89](#)
- setPortRout, [89](#)
- setPortRst, [89](#)
- setS, [90](#)
- setString, [90](#)
- setStringLabel, [90](#)
- setSymbol, [90](#)
- tickDown, [90](#)
- tickUp, [90](#)
- write, [90](#)
- hebe::dataflow::sync::Histogram
 - compute, [92](#)
 - evaluate, [93](#)
 - getCounter, [93](#)
 - getDecr, [93](#)
 - getHistogram, [93](#)
 - getNumBITS, [93](#)
 - Histogram, [92](#)
 - reset, [93](#)
 - setCounter, [93](#)
 - setDecr, [94](#)
 - setHistogram, [94](#)
- hebe::dataflow::sync::In1
 - In1, [96](#)
- hebe::dataflow::sync::In16
 - In16, [96](#)
- hebe::dataflow::sync::In2
 - In2, [97](#)
- hebe::dataflow::sync::In32
 - In32, [98](#)
- hebe::dataflow::sync::In4
 - In4, [99](#)
- hebe::dataflow::sync::In8
 - In8, [100](#)
- hebe::dataflow::sync::Max
 - compute, [100](#)
 - Max, [100](#)
- hebe::dataflow::sync::Merge
 - evaluate, [103](#)
 - Merge, [102](#)
- hebe::dataflow::sync::Min
 - compute, [104](#)
 - Min, [103](#)
- hebe::dataflow::sync::Mod
 - compute, [105](#)
 - Mod, [105](#)
- hebe::dataflow::sync::Modl
 - compute, [106](#)
 - Modl, [106](#)
- hebe::dataflow::sync::Mul
 - compute, [107](#)
 - Mul, [107](#)
- hebe::dataflow::sync::Mull
 - compute, [108](#)
 - Mull, [108](#)
- hebe::dataflow::sync::Not
 - compute, [109](#)
 - Not, [109](#)
- hebe::dataflow::sync::Or
 - compute, [110](#)
 - Or, [110](#)
- hebe::dataflow::sync::Orl
 - compute, [111](#)
 - Orl, [111](#)
- hebe::dataflow::sync::Out1
 - Out1, [112](#)
- hebe::dataflow::sync::Out16
 - Out16, [113](#)
- hebe::dataflow::sync::Out2
 - Out2, [114](#)
- hebe::dataflow::sync::Out32
 - Out32, [115](#)
- hebe::dataflow::sync::Out4
 - Out4, [115](#)
- hebe::dataflow::sync::Out8
 - Out8, [116](#)
- hebe::dataflow::sync::Register
 - Register, [119](#)
- hebe::dataflow::sync::Shl
 - compute, [120](#)
 - Shl, [120](#)
- hebe::dataflow::sync::Shll
 - compute, [121](#)
 - Shll, [121](#)
- hebe::dataflow::sync::Shr
 - compute, [122](#)
 - Shr, [122](#)
- hebe::dataflow::sync::Shrl
 - compute, [123](#)
 - Shrl, [123](#)
- hebe::dataflow::sync::Slt
 - compute, [124](#)
 - Slt, [124](#)
- hebe::dataflow::sync::Sltl
 - compute, [125](#)
 - Sltl, [125](#)
- hebe::dataflow::sync::Sub
 - compute, [126](#)
 - Sub, [126](#)
- hebe::dataflow::sync::Subl
 - compute, [127](#)
 - Subl, [127](#)
- hebe::util::ConfReader
 - ReadConfig, [22](#)
- Histogram
 - hebe::dataflow::sync::Histogram, [92](#)
- In1
 - hebe::dataflow::sync::In1, [96](#)
- In16

- hebe::dataflow::sync::In16, [96](#)
- In2
 - hebe::dataflow::sync::In2, [97](#)
- In32
 - hebe::dataflow::sync::In32, [98](#)
- In4
 - hebe::dataflow::sync::In4, [99](#)
- In8
 - hebe::dataflow::sync::In8, [100](#)
- initialize
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericBranch, [42](#)
 - hebe::dataflow::sync::GenericBranchI, [49](#)
 - hebe::dataflow::sync::GenericI, [59](#)
 - hebe::dataflow::sync::GenericIn, [69](#)
 - hebe::dataflow::sync::GenericOut, [79](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- isDone
 - hebe::dataflow::sync::GenericOut, [80](#)
- isEnabledAnimationFlag
 - hebe::dataflow::sync::GenericIn, [69](#)
 - hebe::dataflow::sync::GenericOut, [80](#)
- isStart
 - hebe::dataflow::sync::GenericIn, [69](#)
- Max
 - hebe::dataflow::sync::Max, [100](#)
- Merge
 - hebe::dataflow::sync::Merge, [102](#)
- Min
 - hebe::dataflow::sync::Min, [103](#)
- Mod
 - hebe::dataflow::sync::Mod, [105](#)
- ModI
 - hebe::dataflow::sync::ModI, [106](#)
- mousePressed
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericI, [59](#)
- Mul
 - hebe::dataflow::sync::Mul, [107](#)
- Mull
 - hebe::dataflow::sync::Mull, [108](#)
- needsDynamicSymbol
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericBranch, [42](#)
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericI, [59](#)
 - hebe::dataflow::sync::GenericIn, [69](#)
 - hebe::dataflow::sync::GenericOut, [80](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- Not
 - hebe::dataflow::sync::Not, [109](#)
- notCompute
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericI, [59](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- Or
 - hebe::dataflow::sync::Or, [110](#)
- OrI
 - hebe::dataflow::sync::OrI, [111](#)
- Out1
 - hebe::dataflow::sync::Out1, [112](#)
- Out16
 - hebe::dataflow::sync::Out16, [113](#)
- Out2
 - hebe::dataflow::sync::Out2, [114](#)
- Out32
 - hebe::dataflow::sync::Out32, [115](#)
- Out4
 - hebe::dataflow::sync::Out4, [115](#)
- Out8
 - hebe::dataflow::sync::Out8, [116](#)
- ReadConfig
 - hebe::util::ConfReader, [22](#)
- Register
 - hebe::dataflow::sync::Register, [119](#)
- reset
 - hebe::dataflow::sync::AccMax, [8](#)
 - hebe::dataflow::sync::AccMin, [9](#)
 - hebe::dataflow::sync::AccMul, [10](#)
 - hebe::dataflow::sync::GenericAcc, [31](#)
 - hebe::dataflow::sync::GenericI, [60](#)
 - hebe::dataflow::sync::Histogram, [93](#)
- reseted
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- setAcc
 - hebe::dataflow::sync::GenericAcc, [32](#)
- setBackground
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericI, [60](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- setCompName
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericI, [60](#)
 - hebe::dataflow::sync::GenericIn, [70](#)
 - hebe::dataflow::sync::GenericOut, [80](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- setComponentId
 - hebe::dataflow::sync::GenericBranchI, [51](#)
 - hebe::dataflow::sync::GenericI, [60](#)
- setComponentImmediate
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [60](#)
- setComponentType
 - hebe::dataflow::sync::GenericBin, [36](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [60](#)
 - hebe::dataflow::sync::GenericIn, [70](#)

- hebe::dataflow::sync::GenericOut, [80](#)
 - hebe::dataflow::sync::GenericUn, [88](#)
- setCounter
 - hebe::dataflow::sync::GenericAcc, [32](#)
 - hebe::dataflow::sync::Histogram, [93](#)
- setDecr
 - hebe::dataflow::sync::Histogram, [94](#)
- setDefaultdelay
 - hebe::dataflow::sync::GenericIn, [70](#)
 - hebe::dataflow::sync::GenericOut, [80](#)
- setDelay
 - hebe::dataflow::sync::GenericIn, [70](#)
 - hebe::dataflow::sync::GenericOut, [80](#), [81](#)
- setDone
 - hebe::dataflow::sync::GenericOut, [81](#)
- setEnableAnimationFlag
 - hebe::dataflow::sync::GenericIn, [70](#)
 - hebe::dataflow::sync::GenericOut, [81](#)
- setHistogram
 - hebe::dataflow::sync::Histogram, [94](#)
- setId
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [60](#)
- setIdxDin
 - hebe::dataflow::sync::GenericIn, [71](#)
- setIdxDout
 - hebe::dataflow::sync::GenericOut, [81](#)
- setImmediate
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [60](#)
- setLabel_nome
 - hebe::dataflow::sync::GenericBranch, [43](#)
- setLabelFormatter
 - hebe::dataflow::sync::GenericIn, [71](#)
 - hebe::dataflow::sync::GenericOut, [81](#)
- setLabelNome
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setN_bits
 - hebe::dataflow::sync::GenericIn, [71](#)
 - hebe::dataflow::sync::GenericOut, [81](#)
- setPortClk
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericIn, [71](#)
 - hebe::dataflow::sync::GenericOut, [81](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortDconf
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericIn, [71](#)
- setPortDin
 - hebe::dataflow::sync::GenericBranchI, [52](#)
 - hebe::dataflow::sync::GenericI, [61](#)
- hebe::dataflow::sync::GenericOut, [82](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortDin1
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
- setPortDin2
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [43](#)
- setPortDout
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericIn, [71](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortElse
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericBranchI, [53](#)
- setPortEn
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericOut, [82](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortEnOut
 - hebe::dataflow::sync::GenericIn, [71](#)
- setPortIf
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericBranchI, [53](#)
- setPortRdy
 - hebe::dataflow::sync::GenericIn, [71](#)
 - hebe::dataflow::sync::GenericOut, [82](#)
- setPortRin
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericOut, [82](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortRin1
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
- setPortRin2
 - hebe::dataflow::sync::GenericBin, [37](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
- setPortRout
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericI, [61](#)
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setPortRst
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [82](#)
 - hebe::dataflow::sync::GenericUn, [89](#)
- setQtdeSave
 - hebe::dataflow::sync::GenericOut, [82](#)
- setS

- hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- setStart
 - hebe::dataflow::sync::GenericIn, [72](#)
- setString
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [44](#)
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- setStringLabel
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [45](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- setStringLabelId
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [62](#)
- setStringLabelImmediate
 - hebe::dataflow::sync::GenericBranchI, [53](#)
 - hebe::dataflow::sync::GenericI, [62](#)
- setSymbol
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [45](#)
 - hebe::dataflow::sync::GenericBranchI, [54](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- setTamVectorOut
 - hebe::dataflow::sync::GenericOut, [82](#)
- setValueLabel
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [82](#)
- setVector
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVector_000
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVector_111
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVector_UUU
 - hebe::dataflow::sync::GenericIn, [72](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVector_XXX
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVector_ZZZ
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVectorIn
 - hebe::dataflow::sync::GenericIn, [73](#)
- setVectorOut
 - hebe::dataflow::sync::GenericOut, [83](#)
- setVectorOutputPort
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [84](#)
- ShI
 - hebe::dataflow::sync::ShI, [120](#)
- ShII
 - hebe::dataflow::sync::ShII, [121](#)
- Shr
 - hebe::dataflow::sync::Shr, [122](#)
- ShrI
 - hebe::dataflow::sync::ShrI, [123](#)
- Slt
 - hebe::dataflow::sync::Slt, [124](#)
- SltI
 - hebe::dataflow::sync::SltI, [125](#)
- startFpgaJtag
 - hebe::dataflow::DataflowSyncSimulBase, [24](#)
- startSimulation
 - hebe::dataflow::DataflowSyncSimulBase, [24](#), [25](#)
- Sub
 - hebe::dataflow::sync::Sub, [126](#)
- SubI
 - hebe::dataflow::sync::SubI, [127](#)
- tickDown
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [45](#)
 - hebe::dataflow::sync::GenericBranchI, [54](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- tickUp
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [45](#)
 - hebe::dataflow::sync::GenericBranchI, [54](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericUn, [90](#)
- updateSymbol
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [84](#)
- wakeup
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [84](#)
- write
 - hebe::dataflow::sync::GenericBin, [38](#)
 - hebe::dataflow::sync::GenericBranch, [45](#)
 - hebe::dataflow::sync::GenericBranchI, [54](#)
 - hebe::dataflow::sync::GenericI, [62](#)
 - hebe::dataflow::sync::GenericIn, [73](#)
 - hebe::dataflow::sync::GenericOut, [84](#)
 - hebe::dataflow::sync::GenericUn, [90](#)