

Block Component modeling

Spoken Tutorial Project
<http://spoken-tutorial.org>

National Mission on Education through ICT
<http://sakshat.ac.in>

Bhargava Nemmaru
FOSSEE, IIT Bombay

22 June 2015



Learning Objectives

In this tutorial, we are going to learn how to:

Learning Objectives

In this tutorial, we are going to learn how to:

- ▶ **define a block**

Learning Objectives

In this tutorial, we are going to learn how to:

- ▶ **define a block**
- ▶ **connect blocks**

Learning Objectives

In this tutorial, we are going to learn how to:

- ▶ **define a block**
- ▶ **connect blocks**
- ▶ **use blocks from Modelica Library**

System Requirements

- ▶ **OpenModelica 1.9.2**

System Requirements

- ▶ **OpenModelica 1.9.2**
- ▶ **Any OS: Linux, Windows, Mac OS X or FOSSEE OS on ARM**

Prerequisites

- ▶ **Knowledge of component oriented modeling in Modelica**

Prerequisites

- ▶ Knowledge of component oriented modeling in Modelica
- ▶ Prerequisite tutorials are mentioned on www.spoken-tutorial.org

- ▶ **block** is a specialized class

block

- ▶ **block** is a specialized class
- ▶ It is useful in control applications

block

- ▶ **block** is a specialized class
- ▶ It is useful in control applications
- ▶ Eg: Modelica Library has blocks for PI and PID Controllers

block

- ▶ **block** is a specialized class
- ▶ It is useful in control applications
- ▶ Eg: Modelica Library has blocks for PI and PID Controllers
- ▶ Variables of a **block** class must have fixed causality: either **input** or **output**



block

- ▶ **block** is a specialized class
- ▶ It is useful in control applications
- ▶ Eg: Modelica Library has blocks for PI and PID Controllers
- ▶ Variables of a **block** class must have fixed causality: either **input** or **output**
- ▶ Blocks can be connected using **connect** statements

block

- ▶ **block** is a specialized class
- ▶ It is useful in control applications
- ▶ Eg: Modelica Library has blocks for PI and PID Controllers
- ▶ Variables of a **block** class must have fixed causality: either **input** or **output**
- ▶ Blocks can be connected using **connect** statements

- ▶ **Connectors act as interface between blocks**

block connectors

- ▶ **Connectors act as interface between blocks**
- ▶ **They are used to model input and output signals**

block connectors

- ▶ Connectors act as interface between blocks
- ▶ They are used to model input and output signals
- ▶ eg: **connector** RealInput = **input** Real;

Problem Statement

Write a class using blocks to perform the following operations:

- ▶ **Take two time varying signals as input and output their sum**

Problem Statement

Write a class using blocks to perform the following operations:

- ▶ **Take two time varying signals as input and output their sum**
- ▶ **Take a time varying signal as input and amplify it by a constant**

Sum of signals



- ▶ **Signal 1 - t**
- ▶ **Signal 2 - $2*t^2$**

Amplification of signal



- ▶ **Signal 1 - t**
- ▶ **Signal 2 - k(=5)**



► MISO - Multiple Input Single Output

MISO



- ▶ **MISO - Multiple Input Single Output**
- ▶ **Available in Modelica.Interfaces.Block package**

MISO



- ▶ **MISO - Multiple Input Single Output**
- ▶ **Available in Modelica.Interfaces.Block package**
- ▶ **$u[nin]$ - Input vector with nin dimensions**

MISO



- ▶ **MISO - Multiple Input Single Output**
- ▶ **Available in Modelica.Interfaces.Block package**
- ▶ **$u[nin]$ - Input vector with nin dimensions**



y - Output



Solution Methodology

- ▶ Extend MISO to create a block named **'Sum'**

Solution Methodology

- ▶ Extend MISO to create a block named **'Sum'**
- ▶ Extend MISO to create a block named **'Product'**

Solution Methodology

- ▶ Extend MISO to create a block named **'Sum'**
- ▶ Extend MISO to create a block named **'Product'**
- ▶ Create a class named **'main'**

Solution Methodology

- ▶ Extend MISO to create a block named **'Sum'**
- ▶ Extend MISO to create a block named **'Product'**
- ▶ Create a class named **'main'**
- ▶ Create instances of **'Sum'** and **'Product'**

Solution Methodology

- ▶ Extend MISO to create a block named **'Sum'**
- ▶ Extend MISO to create a block named **'Product'**
- ▶ Create a class named **'main'**
- ▶ Create instances of **'Sum'** and **'Product'**
- ▶ Program necessary equations related to input and output variables



Assignment

Look into the codes for RealInput, RealOutput, SI, SO and MIMO blocks. You can find them in Modelica.Blocks.Interfaces.

About the Spoken Tutorial Project

- ▶ Watch the video available at http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
- ▶ It summarises the Spoken Tutorial project

About the Spoken Tutorial Project

- ▶ Watch the video available at http://spoken-tutorial.org/What_is_a_Spoken_Tutorial
- ▶ It summarises the Spoken Tutorial project
- ▶ If you do not have good bandwidth, you can download and watch it

Spoken Tutorial Workshops

The Spoken Tutorial Project Team

- ▶ Conducts workshops using spoken tutorials
- ▶ Gives certificates to those who pass an online test
- ▶ For more details, please write to contact@spoken-tutorial.org

Forum to answer questions

- ▶ Do you have questions in **THIS Spoken Tutorial?**
- ▶ Choose the minute and second where you have the question.
- ▶ Explain your question briefly.
- ▶ Someone from the **FOSSEE** team will answer them.

Please visit <http://forums.spoken-tutorial.org/>



Textbook Companion Project

- ▶ **The FOSSEE team coordinates coding of solved examples of popular books**
- ▶ **We give honorarium and certificate to those who do this**

For more details, please visit this site:

<http://OM.fossee.in/Textbook-Companion-Project>



Lab Migration Project

- ▶ **The FOSSEE team helps migrate commercial simulator labs to OpenModelica**
- ▶ **We give honorarium and certificates to those who do this**

For more details, please visit this site:

<http://OM.fossee.in/lab-migration-project>



Acknowledgements

- ▶ Spoken Tutorial Project is a part of the Talk to a Teacher project
- ▶ It is supported by the National Mission on Education through ICT, MHRD, Government of India
- ▶ More information on this Mission is available at <http://spoken-tutorial.org/NMEICT-Intro>

Thanks!

<http://openmodelica.org>