

Ansys 2025/R2

POWERING INNOVATION THAT DRIVES HUMAN ADVANCEMENT

© 2025 ANSYS, Inc. or its affiliated companies
Unauthorized use, distribution, or duplication is prohibited.

Twin Builder® Components: Transfer



ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
<https://www.ansys.com>
(T) 724-746-3304
(F) 724-514-9494

Release 2025 R2
July 2025

ANSYS, Inc. and
ANSYS Europe,
Ltd. are UL
registered ISO
9001:2015 com-
panies.

Copyright and Trademark Information

© 1986-2025 ANSYS, Inc. Unauthorized use, distribution or duplication is prohibited.

ANSYS, Ansys Workbench, AUTODYN, CFX, FLUENT and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries located in the United States or other countries. ICM CFD is a trademark used by ANSYS, Inc. under license. All other brand, product, service and feature names or trademarks are the property of their respective owners. FLEXIm and FLEXnet are trademarks of Flexera Software LLC.

Disclaimer Notice

THIS ANSYS SOFTWARE PRODUCT AND PROGRAM DOCUMENTATION INCLUDE TRADE SECRETS AND ARE CONFIDENTIAL AND PROPRIETARY PRODUCTS OF ANSYS, INC., ITS SUBSIDIARIES, OR LICENSORS. The software products and documentation are furnished by ANSYS, Inc., its subsidiaries, or affiliates under a software license agreement that contains provisions concerning non-disclosure, copying, length and nature of use, compliance with exporting laws, warranties, disclaimers, limitations of liability, and remedies, and other provisions. The software products and documentation may be used, disclosed, transferred, or copied only in accordance with the terms and conditions of that software license agreement.

ANSYS, Inc. and ANSYS Europe, Ltd. are UL registered ISO 9001: 2015 companies.

U.S. Government Rights

For U.S. Government users, except as specifically granted by the ANSYS, Inc. software license agreement, the use, duplication, or disclosure by the United States Government is subject to restrictions stated in the ANSYS, Inc. software license agreement and FAR 12.212 (for non-DOD licenses).

Third-Party Software

See the legal information in the product help files for the complete Legal Notice for Ansys proprietary software and third-party software. If you are unable to access the Legal Notice, please contact ANSYS, Inc.

Table of Contents

Table of Contents	Contents-1
1 - Transfer Library	1-2
Linear Transfer Blocks	1-3
Allpass with First Order Time Delay	1-4
Derivative with First Order Time Delay	1-6
Derivative with Second Order Time Delay	1-8
Integrator with First Order Time Delay	1-10
Integrator with Second Order Time Delay	1-12
PD Controller	1-14
PD Element with First Order Time Delay	1-16
PD Element with Second Order Time Delay	1-18
PI Controller	1-20
PID Controller	1-22
PID Element with First Order Time Delay	1-25
PID Element with Second Order Time Delay	1-27
First Order Time Delay Element	1-29
Second Order Delay Element (T1,T2)	1-31
Second Order Delay Element (D,T1)	1-33
Variable Order Time Delay Element	1-35
Non-Linear Transfer Blocks	1-37
Dead Zone	1-38
Divider	1-41
Insensitivity Zone	1-43
Inverse Dead Zone	1-46
Variable Gain	1-49
Keyboard Controlled Ramp Generator	1-51
Pulse Frequency Modulation	1-53
Pulse Width Modulation	1-55

Pulse Width Modulation with Phase Triggering 1-57

Rise Rate Limiting 1-59

Rise Rate Limiting With Initial Value 1-61

Square Function 1-63

Square Root Function 1-65

Switch 11 1-67

Switch 12 1-69

Switch 21 1-71

Index Index-1

1 - Transfer Library

The Blocks in the Transfer model library are defined by an internal identifier, the name and a set of parameters; Blocks have no conservative nodes. A block is a linear or nonlinear transfer element with a defined static or dynamic response characteristic. The block output signal does not influence the block input quantity (non-reacting).

The blocks are subdivided into [Linear](#) and [Nonlinear](#) transfer blocks. The components are contained in the Tools Section.

The model library provides linear transmission components, partially also with nonlinear supplementary actions, which are quite frequently required in control engineering. The models are implemented as macro models and based on the block diagram components of the Basic section.

The block output signal and the specific block dependent parameters can be used as outputs. The block output signal is represented by the **VAL** entry; the other parameters by their corresponding name.

The response characteristic (i.e. the connection between the output signal y and the input signal x) is $y = G(s)*x$ (Laplace-domain, Laplace-operator specified as s).

Linear Transfer Blocks

- Allpass with First Order Time Delay (ALLP1)
- Derivative with First Order Time Delay (DT1)
- Derivative with Second Order Time Delay (DT2)
- Integrator with First Order Time Delay (IT1)
- Integrator with Second Order Time Delay (IT2)
- PD Controller (PD)
- PD Element with First Order Time Delay (PDT1)
- PD Element with Second Order Time Delay (PDT2)
- PI Controller (PI_)
- PID Controller (PID)
- PID Element with First Order Time Delay (PIDT1)
- PID Element with Second Order Time Delay (PIDT2)
- First Order Time Delay Element (PT1)
- Second Order Delay Element T1, T2 (PT2_AP)
- Second Order Delay Element D, T1 (PT2_OSC)
- Variable Order Delay Element (PTN)

Allpass with First Order Time Delay

Library: Transfer

Modeling Language: SML

Version Number: Twin Builder 2025.2



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with all-pass behavior. A typical characteristic is the undershoot in the unit step response, caused by a “negative” metered time constant. The macro consist of a cascade connection of a symmetrical pole-zero point pair and a PT1 element.

You can define the gain, the time constant 1, and time constant 2, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s(1 - T_1 \cdot s)}{1 + T_1 \cdot s} * \frac{1}{1 + T_2 \cdot s}$$

[Top](#)

Netlist Syntax

```
MODEL ALLP1 ?InstanceName(@InstanceName):(@Refbase@ID) ( INPUT:= @INPUT,
KS:= @KS, T1:= @T1, T2:= @T2, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
T1	Time Constant 1	real	1 [s]
T2	Time Constant 2	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Derivative with First Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

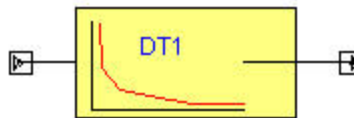


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with differential behavior and time delay of the 1st order.

You can define the derivative gain and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_D \cdot s}{1 + T_1 \cdot s}$$

[Top](#)

Netlist Syntax

MODEL DT1 ?InstanceName(@InstanceName):(@Refbase@)(ID) (INPUT:= @INPUT, KD:= @KD, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KD	Derivative Gain	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Derivative with Second Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with differential behavior and time delay of the 2nd order.

You can define the derivative gain, the damping, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_D \cdot s}{1 + 2D_s T_1 s + T_1^2 s^2}$$

[Top](#)

Netlist Syntax

MODEL DT2 ?InstanceName(@InstanceName):(@Refbase)@(ID)) (INPUT:= @INPUT, KD:= @KD, DS:= @DS, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KD	Derivative Gain	real	1
DS	Damping	real	1
T1	Time Constant 1	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Integrator with First Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with integrating behavior and time delay of the 1st order.

You can define the integral gain and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{1}{s} \cdot \frac{K_I}{1 + T_1 \cdot s}$$

[Top](#)

Netlist Syntax

MODEL IT1 ?InstanceName(@InstanceName):(@ (Refbase)@(ID)) (INPUT:= @INPUT, KI:= @KI, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KI	Integral Gain	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Integrator with Second Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with integrating behavior and time delay of the 2nd order.

You can define the integral gain, the damping, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{1}{s} \cdot \frac{K_I}{1 + 2D_s T_1 s + T_1^2 s^2}$$

[Top](#)

Netlist Syntax

```
MODEL IT2 ?InstanceName(@InstanceName):(@Refbase@)(ID) ( INPUT:= @INPUT, KI:=
@KI, DS:= @DS, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KI	Integral Gain	real	1
DS	Damping	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PD Controller

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

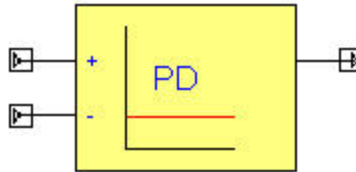


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a continuous PD controller with parameters for the P and D component.

You can define the set point, the initial derivative value, the limitations of the internal D component, and the total controllers output signals, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PD ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( IN:= @IN, SET:= @SET,
KP:= @KP, KD:= @KD, UL:= @UL, LL:= @LL, TS:= @TS, D0:= @D0, UL_D:= @UL_D, LL_D:=
@LL_D) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KP	Gain	real	0.5
KD	Derivative Gain	real	0.1
UL	Upper Limit	real	1e12
LL	Lower Limit	real	-1e12
UL_D	Upper Derivative Value	real	0
LL_D	Lower Derivative Value	real	0
TS	Sample Time	real	0 [s]
D0	Initial Derivative Value	real	0
IN	Input Signal	real	0
SET	Reference Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PD Element with First Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

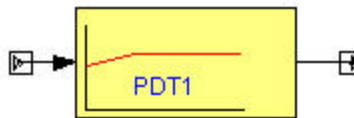


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with both a pole as well as a zero point.

You can define the gain, the derivative action time, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s \cdot (1 + T_D \cdot s)}{1 + T_1 \cdot s}$$

[Top](#)

Netlist Syntax

MODEL PDT1 ?InstanceName(@InstanceName):(@Refbase)(@ID) (INPUT:= @INPUT, KS:= @KS, TD:= @TD, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
T1	Time Constant	real	1 [s]
TD	Derivative Action Time	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PD Element with Second Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous and oscillating transmission element with 2 poles and one zero point.

You can define the gain, the derivative action time, the damping and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s \cdot (1 + T_D \cdot s)}{(1 + 2D_s T_1 s) \cdot (T_1^2 s^2)}$$

[Top](#)

Netlist Syntax

```
MODEL PDT2 ?InstanceName(@InstanceName):(@Refbase@ID) ( INPUT:= @INPUT,
KS:= @KS, DS:= @DS, TD:= @TD, T1:= @T1, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
DS	Damping	real	1
TD	Derivative Action Time	real	1 [s]
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PI Controller

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

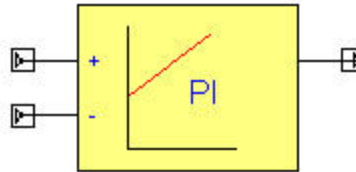


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a continuous PI controller with parameters for the P and I component.

You can define the set point, the initial integral value, the limitations of the internal I component, and the total controllers output signals, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PI_?InstanceName(@InstanceName):(@Refbase)@(ID)) ( IN:= @IN, SET:= @SET,
KP:= @KP, KI:= @KI, UL:= @UL, LL:= @LL, TS:= @TS, IO:= @IO, UL_I:= @UL_I, LL_I:= @LL_
I) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KP	Gain	real	0.5
KI	Integral Gain	real	0.1
UL	Upper Limit	real	1e12
LL	Lower Limit	real	-1e12
TS	Sample Time	real	0 [s]
IO	Initial Integral Value	real	0
UL_I	Upper Integral Limit	real	0
LL_I	Lower Integral Limit	real	0
IN	Input Signal	real	0
SET	Reference Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PID Controller

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

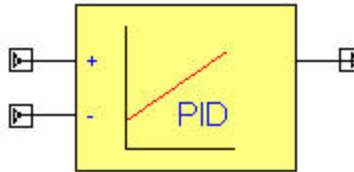


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a continuous PID controller with parameters for the P, D, and I component.

You can define the set point, the initial integral value, the initial derivative value, the limitations of the internal I component, the limitations of the internal D component and the total controllers output signals, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PID ?InstanceName(@InstanceName):(@ (Refbase)@(ID)) ( IN:= @IN, SET:= @SET,
KP:= @KP, KI:= @KI, KD:= @KD, UL:= @UL, LL:= @LL, TS:= @TS, IO:= @IO, UL_I:= @UL_I,
LL_I:= @LL_I, D0:= @D0, UL_D:= @UL_D, LL_D:= @LL_D) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
KP	Gain	real	0.5
KI	Integral Gain	real	0.1
KD	Derivative Gain	real	0.1
UL	Upper Limit	real	1e12
LL	Lower Limit	real	-1e12
TS	Sample Time	real	0 [s]
IO	Initial Integral Value	real	0
UL_I	Upper Integral Limit	real	0
LL_I	Lower Integral Limit	real	0
DO	Initial Derivative Value	real	0
UL_D	Upper Derivative Limit	real	0
LL_D	Lower Derivative Limit	real	0
IN	Input Signal	real	0
SET	Reference Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PID Element with First Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element.

You can define the gain, the integral action time, the derivative action time, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = K_s \cdot \left(\frac{1 + \frac{1}{T_N \cdot s} + T_V \cdot s}{1 + T_1 \cdot s} \right) = K_s \cdot \frac{1 + T_N \cdot s + T_N \cdot T_V \cdot s^2}{T_N \cdot s + T_N \cdot T_1 \cdot s^2}$$

[Top](#)

Netlist Syntax

```
MODEL PIDT1 ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT,
KS:= @KS, TN:= @TN, TV:= @TV, T1:= @T1, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
TN	Integral Action Time	real	10 [s]
TV	Derivative Action Time	real	10 [s]
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

PID Element with Second Order Time Delay

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear, continuous, and oscillating transmission element.

You can define the gain, the integral action time, the derivative action time, the damping and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = K_s \cdot \left(\frac{1 + \frac{1}{T_N \cdot s} + T_V \cdot s}{1 + 2T_1 D_s s + T_1^2 s^2} \right) = K_s \cdot \frac{1 + T_N \cdot s + T_N \cdot T_V \cdot s^2}{T_N \cdot s + 2 \cdot D_s \cdot T_N \cdot T_1 \cdot s^2 + T_N \cdot T_1^2 \cdot s^3}$$

[Top](#)

Netlist Syntax

```
MODEL PIDT2 ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT,
KS:= @KS, TN:= @TN, TV:= @TV, DS:= @DS, T1:= @T1, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
TN	Integral Action Time	real	10 [s]
TV	Derivative Action Time	real	10 [s]
DS	Damping	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

First Order Time Delay Element

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with time delay of the 1st order. You can define the gain and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s}{1 + T_1 \cdot s}$$

[Top](#)

Netlist Syntax

```
MODEL PT1 ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT, KS:=  
@KS, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Second Order Delay Element (T1,T2)

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

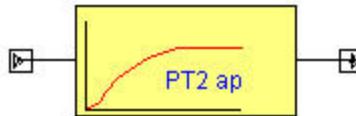


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with time delay of the 2nd order (two time constants, aperiodic).

You can define the gain, the time constant 1, and the time constant 2, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s}{(1 + T_1 \cdot s) \cdot (1 + T_2 \cdot s)}$$

[Top](#)

Netlist Syntax

```
MODEL PT2_AP ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT,
KS:= @KS, T1:= @T1, T2:= @T2, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
T1	Time Constant 1	real	1 [s]
T2	Time Constant 2	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Second Order Delay Element (D,T1)

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear, continuous, and oscillating transmission element with time delay of the 2nd order.

You can define the gain, the damping, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s}{(1 + 2D_s T_1 \cdot s - T_1^2 s^2)}$$

[Top](#)

Netlist Syntax

```
MODEL PT2_OSC ?InstanceName(@InstanceName):(@@ (Refbase)@(ID)) ( INPUT:= @INPUT,
KS:= @KS, DS:= @DS, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
DS	Damping	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Variable Order Time Delay Element

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

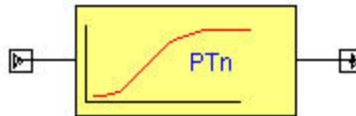


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a linear continuous transmission element with time delay of n-th order.

You can define the gain, the order, and the time constant, as well as the sample time and input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$G(s) = \frac{K_s}{(1 + T_1 \cdot s)^N} \quad 0 < N < 10$$

[Top](#)

Netlist Syntax

```
MODEL PTN ?InstanceName(@InstanceName):(@(Refbase)@(ID)) ( INPUT:= @INPUT, N:=  
@N, KS:= @KS, T1:= @T1, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
Ks	Gain	real	1
N	Order	real	1
T1	Time Constant	real	1 [s]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Non-Linear Transfer Blocks

- Dead Zone (DEADBAND)
- Divider (DIV)
- Insensitivity Zone (INSTTY)
- Inverse Dead Zone (INVDEAD)
- Variable Gain (KPVAR)
- Keyboard Controlled Ramp Generator (KRMP)
- Pulse Frequency Modulation (PFM)
- Pulse Width Modulation (PWM)
- Pulse Width Modulation with Phase Triggering (PWM_phase)
- Rise Rate Limiting (RISEMAX)
- Rise Rate Limiting with Initial Value(RISEMAX_INIT)
- Square Function (SQR)
- Square Root Function (SQRT)
- Switch 11 (SW10)
- Switch 12 (SW12)
- Switch 21 (SW21)

Dead Zone

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a dead zone element. All input signals within the interval $[a,b]$ will be ignored. The signals left of the dead zone are multiplied by $K1$, the signals right of the dead zone by $K3$.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x)$$

$$x \leq a \quad y = K1 \cdot x$$

$$a < x \leq b \quad y = 0$$

$$x > b \quad y = K3 \cdot x$$

[Top](#)

Netlist Syntax

```
MODEL DEADBAND ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:=
@INPUT, K1:= @K1, K3:= @K3, A:= @A, B:= @B, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
K1	Left Gain	real	1
K3	Right Gain	real	1
A	Left Limit	real	-1
B	Right Limit	real	1
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Divider

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

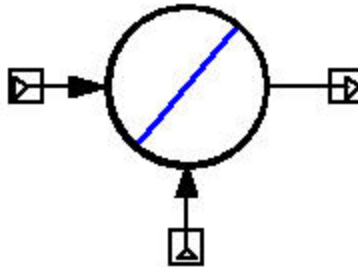


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents the division of two signals.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = \frac{x_1}{x_2} = \frac{I_N}{I_D} \quad x_2 \neq 0$$

[Top](#)

Netlist Syntax

MODEL DIV ?InstanceName(@InstanceName):(@(Refbase)@(ID)) (I_N:= @I_N, I_D:= @I_D, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
I_N	Dividend	real	0
I_D	Divisor	real	1
TS	Sample Time	real	0 [s]

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Insensitivity Zone

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------



Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a insensitivity zone element. All input signals within the interval $[a,b]$ are multiplied by $K2$. The signals left of the insensitivity zone are multiplied by $K1$, the signals right of the insensitivity zone by $K3$. If $K2 > K1, K3$, an oversensitive zone is defined.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x)$$

$$x \leq a \quad y = K1 \cdot x$$

$$a < x \leq b \quad y = K2 \cdot x$$

$$x > b \quad y = K3 \cdot x$$

[Top](#)

Netlist Syntax

```
MODEL INSTTY ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT,
K1:= @K1, K2:= @K2, K3:= @K3, A:= @A, B:= @B, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
K1	Left Gain	real	1
K2	Central Gain	real	0.1
K3	Right Gain	real	1
A	Left Limit	real	-1
B	Right Limit	real	1
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Inverse Dead Zone

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

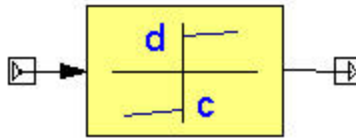


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a inverse dead zone element. All input signals less than a trigger point are multiplied by K1 and added to an offset CC. All input signals greater than a trigger point are multiplied by K2 and added to an offset DD. So a preload characteristic can be modeled (e.g. compensation the dead zone of a real actuator).

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x)$$

$$x < TRIGG \quad y = K1 \cdot x + CC$$

$$x \geq TRIGG \quad y = K2 \cdot x + DD$$

[Top](#)

Netlist Syntax

MODEL INVDEAD ?InstanceName(@InstanceName):(@Refbase)@(ID)) (INPUT:= @INPUT, K1:= @K1, K2:= @K2, TRIGG:= @TRIGG, CC:= @CC, DD:= @DD, TS:= @TS) SRC: DB(Lib:- :=@ModelLibraryName) ;

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
K1	Left Gain	real	1
K2	Right Gain	real	1
TRIGG	Threshold Value	real	0
CC	Lower Offset	real	-1
DD	Upper Offset	real	1
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Variable Gain

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

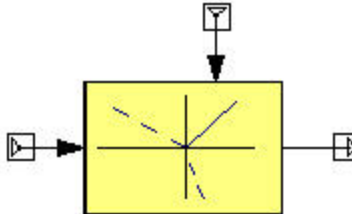


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a variable, over a control signal switched factor. With the block a direction coupled transfer characteristic, or an amplification in only one direction (diode function) can be modeled.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x, CTRL)$$

$$y = K2 \cdot x, \quad \text{if } CTRL \geq TRIGG$$

$$y = K1 \cdot x, \quad \text{otherwise}$$

[Top](#)

Netlist Syntax

```
MODEL KPVAR ?InstanceName(@InstanceName):(@ (Refbase)@(ID)) ( INPUT:= @INPUT,
CTRL:= @CTRL, K1:= @K1, K2:= @K2, TRIGG:= @TRIGG, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
K1	Left Gain	real	-1
K2	Right Gain	real	1
TRIGG	Threshold Value	real	0
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0
CTRL	Control Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Keyboard Controlled Ramp Generator

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

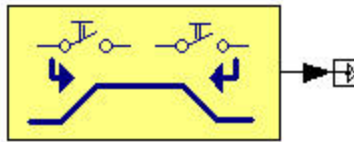


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a ramp generator, which is controlled with user-defined key actions.

You can define the rate of rise, the initial value, the limitations of the output signal, and the keys controlling the ramp curve, as well as the sample time.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL KRMP ?InstanceName(@InstanceName):(@(Rebase)@(ID)) ( SR:= @SR, SR0:=  
@SR0, UL:= @UL, LL:= @LL, TEP:= @TEP, TAP:= @TAP, TEM:= @TEM, TAM:= @TAM,  
TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
SR	Slope Rate	real	0
SR0	Initial Slope Rate	real	0
UL	Upper Limit	real	0
LL	Lower Limit	real	0
TS	Sample Time	real	0 [s]

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Pulse Frequency Modulation

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

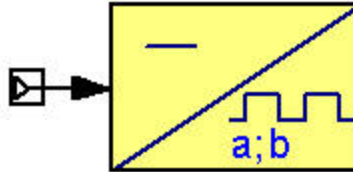


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a pulse rate modulation for symmetrical input signals (signal amplitudes between maximum values). The output A provides a pulse with a constant ON period and a variable OFF period. The output B provides a pulse with a variable ON period and a constant OFF period. Positive input signal amplitudes result in high pulse rate, negative input signal amplitudes result in lower pulse rate.

You can define the ON period, the maximum period, and the maximum input amplitude, as well as the sample time and the input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PFM ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:= @INPUT,  
TON:= @TON, TMAX:= @TMAX, INMAX:= @INMAX, TS:= @TS, A:= @A, B:= @B) SRC: DB  
(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
T _{ON}	ON Period	real	0.0005 [s]
T _{MAX}	OFF Period	real	0.001 [s]
INMAX	Maximum Input Amplitude	real	1
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
A	Output Signal	Input/Output	real
B	Negated Output Signal	Input/Output	real

[Top](#)

Example

[Top](#)

References

Pulse Width Modulation

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

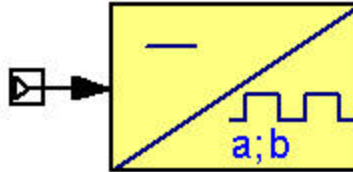


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a pulse-width modulation. The intern trigger signal (saw-tooth rising) should be adjusted in frequency, amplitude, offset to the input signal. Example: At an input signal between 0V and +10V a trigger signal with an amplitude of 5V and an offset of 5V should be used. A maximum input signal result to maximum pulse width, a minimum input signal result to minimum pulse width (pulse width close to 0)..

You can define amplitude, offset, and frequency of the trigger signal, as well as the sample time ant the input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PWM ?InstanceName(@InstanceName):(@Refbase)@(ID) ( INPUT:= @INPUT,
ATR:= @ATR, FTR:= @FTR, OTR:= @OTR, TS:= @TS, A:= @A, B:= @B) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
ATR	Trigger Signal Amplitude	real	1
FTR	Trigger Signal Frequency	real	1
OTR	Trigger Signal Offset	real	0
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
A	Output Signal	Input/Output	real
B	Negated Output Signal	Input/Output	real

[Top](#)

Example

[Top](#)

References

Pulse Width Modulation with Phase Triggering

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

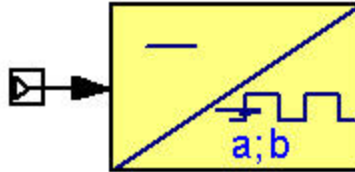


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a pulse-width modulation with threshold value. If the input signal \geq threshold value, the output A is '1'. If the input signal $<$ threshold value, the output A is '0'.

You can define the threshold value, as well as the sample time and the input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL PWM_phase ?InstanceName(@InstanceName):(@Refbase)@(ID)) ( INPUT:=
@INPUT, ATR:= @ATR, FTR:= @FTR, OTR:= @OTR, PHTR:= @PHTR, TS:= @TS, A:= @A,
B:= @B) SRC: DB(Lib:=@ModelLibraryName) ;
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
ATR	Trigger Signal Amplitude	real	1
FTR	Trigger Signal Frequency	real	1 [Hz]
OTR	Trigger Signal Offset	real	0
PHTR	Trigger Signal Phase	real	0 [deg]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
TRIGGER	Trigger Signal	Output	real
A	Output Signal	Input/Output	real
B	Negated Output Signal	Input/Output	real

[Top](#)

Example

[Top](#)

References

Rise Rate Limiting

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

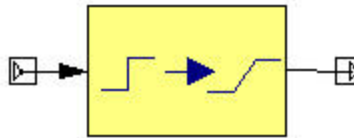


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a limited, maximum possible rate of signal rise. A real actuator can only react to input signal changes with a limited rate, i.e., if an input signal changes more quickly than the given limit, the rate of change is bound to this limit..

You can define the maximum rise for the positive and negative direction separately, as well as the sample time and the input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

MODEL RISEMAX ?InstanceName(@InstanceName):(@ (Refbase)@(ID)) (INPUT:= @INPUT, APOS:= @APOS, ANEG:= @ANEG, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
APOS	Positive Rise Rate	real	1
ANEG	Negative Rise Rate	real	-1
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Rise Rate Limiting With Initial Value

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

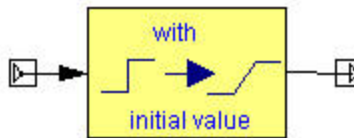


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a limited, maximum possible rate of signal rise with a defined initial value. A real actuator can only react to input signal changes with a limited rate, i.e., if an input signal changes more quickly than the given limit, the rate of change is bound to this limit.

You can define the maximum rise for the positive and negative direction separately and the initial value, as well as the sample time and the input signal.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

[Top](#)

Netlist Syntax

```
MODEL RISEMAX_INIT ?InstanceName(@InstanceName):(@ (Refbase)@ (ID)) ( INPUT:=  
@INPUT, APOS:= @APOS, ANEG:= @ANEG, RR0:= @RR0) SRC: DB(Lib:-  
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
APOS	Positive Rise Rate	real	1
ANEG	Negative Rise Rate	real	-1
RRO	Initial Rise Rate	real	0
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Square Function

Library: Transfer

Modeling Language: SML

Version Number: Twin Builder 2025.2

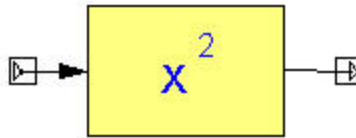


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a square function.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = x^2$$

[Top](#)

Netlist Syntax

```
MODEL SQR ?InstanceName(@InstanceName):(@Refbase@)(ID) ( INPUT:= @INPUT, TS:=  
@TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Square Root Function

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

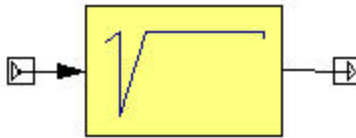


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a square root function. For negative input signals, the output signal is zero.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = \sqrt{x} \quad x \geq 0$$

$$y = 0 \quad \textit{in all other cases}$$

[Top](#)

Netlist Syntax

```
MODEL SQRT ?InstanceName(@InstanceName):(@(Refbase)@(ID)) ( INPUT:= @INPUT,  
TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Switch 11

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

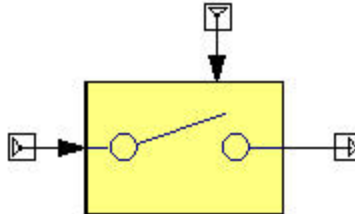


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a controlled switch. Dependent on the control signal, the input signal is connected to the output, or zero is set at the output.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x, CTRL)$$

$$y = x \quad \text{if } CTRL \geq TRIGG$$

$$y = 0 \quad \text{otherwise}$$

[Top](#)

Netlist Syntax

```
MODEL SW10 ?InstanceName(@InstanceName):(@Refbase@ID) ( INPUT:= @INPUT,  
CTRL:= @CTRL, TRIGG:= @TRIGG, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName) ;
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
CTRL	Control Signal	real	0
TRIGG	Trigger Signal	real	0
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Switch 12

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

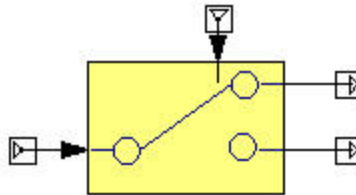


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a controlled switch. Dependent on the control signal, the input signal is connected to the output 1 or the output 2.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y1, y2 = f(x, CTRL)$$

$$y1 = x \quad \text{if } CTRL \geq TRIGG$$

$$y2 = x \quad \text{otherwise}$$

[Top](#)

Netlist Syntax

```
MODEL SW12 ?InstanceName(@InstanceName):(@ (Refbase)@(ID)) ( INPUT:= @INPUT,
CTRL:= @CTRL, TRIGG:= @TRIGG, TS:= @TS) SRC: DB(Lib:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
CTRL	Control Signal	real	0
TRIGG	Trigger Signal	real	0
TS	Sample Time	real	0 [s]
INPUT	Input Signal	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL_1	Output Signal 1	Output	real
VAL_2	Output Signal 2	Output	real

[Top](#)

Example

[Top](#)

References

Switch 21

Library: Transfer	Modeling Language: SML	Version Number: Twin Builder 2025.2
-------------------	------------------------	-------------------------------------

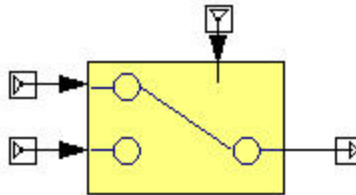


Figure 1. Component symbol

- [Description](#)
- [Assumptions and Limitations](#)
- [Mathematical Description](#)
- [Netlist Syntax](#)
- [Parameters](#)
- [Input/Output Quantities](#)
- [Example](#)
- [References](#)

Description

The Block represents a controlled switch. Dependent on the control signal, input signal 1 or input signal 2 is connected to the output.

[Top](#)

Assumptions and Limitations

[Top](#)

Mathematical Description

$$y = f(x1, x2, CTRL)$$

$$y = x1 \quad \text{if } CTRL \geq TRIGG$$

$$y = x2 \quad \text{otherwise}$$

[Top](#)

Netlist Syntax

```
MODEL SW21 ?InstanceName(@InstanceName):(@Refbase@ID) ( INPUT_1:= @INPUT_
1, INPUT_2:= @INPUT_2, CTRL:= @CTRL, TRIGG:= @TRIGG, TS:= @TS) SRC: DB(Lib:-
:=@ModelLibraryName);
```

[Top](#)

Parameters

Table 1

Name	Description	Data Type	Default Value [Unit]
CTRL	Control Signal	real	0
TRIGG	Trigger Signal	real	0
TS	Sample Time	real	0 [s]
INPUT_1	Input Signal 1	real	0
INPUT_2	Input Signal 2	real	0

[Top](#)

Input/Output Quantities

Table 2

Name	Description [Unit]	Direction	Data Type
VAL	Output Signal	Output	real

[Top](#)

Example

[Top](#)

References

Index

A

Allpass with First Order Time Delay
1-4

D

Dead Zone 1-38

Derivative with First Order Time
Delay 1-6

Derivative with Second Order Time
Delay 1-8

Divider 1-41

F

First Order Time Delay Element 1-
29

I

Insensitivity Zone 1-43

Integrator with First Order Time
Delay 1-10

Integrator with Second Order Time
Delay 1-12

Inverse Dead Zone 1-46

K

Keyboard Controlled Ramp Gen-
erator 1-51

L

Linear Transfer Blocks 1-3

N

Non-Linear Transfer Blocks 1-37

P

PD Controller 1-14

PD Element with First Order Time
Delay 1-16

PD Element with Second Order Time
Delay 1-18

PI Controller 1-20

PID Controller 1-22

PID Element with First Order Time
Delay 1-25

PID Element with Second Order Time
Delay 1-27

Pulse Frequency Modulation 1-53

Pulse Width Modulation 1-55

Pulse Width Modulation with Phase
Triggering 1-57

R

Rise Rate Limiting 1-59

Rise Rate Limiting With Initial Value
1-61

S

Second Order Delay Element (D,T1)
1-33

Second Order Delay Element
(T1,T2) 1-31

Square Function 1-63

Square Root Function 1-65

Switch 11 1-67

Switch 12 1-69

Switch 21 1-71

T

Transfer Blocks 1-2

V

Variable Gain 1-49

Variable Order Time Delay Element
1-35