

Benutzerhandbuch

www.simapp.com Version 2.6

SimApp

für Windows 2000/XP/Vista

Nutzungsbestimmungen

Die mit dieser Lizenz erworbene Software "SimApp" mit ihren Ausgaben "Light", "Workstation" und "Server" ist Eigentum der Firma Büsser Engineering (BE) oder derer Lizenzgeber und ist durch nationale Gesetze und internationale Verträge urheberrechtlich geschützt. Mit der Annahme der Lizenzbedingungen erhalten Sie das Recht zur Benutzung der Software. Sofern nicht durch eine mit dieser Lizenz ausgelieferte Zusatzvereinbarung andere Regelungen getroffen werden, unterliegt die Nutzung der Software folgenden Bestimmungen:

Lizenzbestimmungen

Die Nutzungsbefugnis an den SimApp-Ausgaben "Light" und "Workstation" ist beschränkt auf die Nutzung an einem einzigem Computerarbeitsplatz, unabhängig davon, ob es sich um einen PC-Arbeitsplatz innerhalb eines Netzwerkes oder eines sonstigen Mehrstations-Rechensystems oder um einen einzelnen, mit anderen Computerarbeitsplätzen nicht verbundenen PC-Arbeitsplatz handelt. Möchte der Kunde diese Ausgaben auf mehreren Computerarbeitsplätzen installieren, muss er entweder eine entsprechende Anzahl von Lizenzen in Form von Vollversionen (Datenträger und Handbuch) oder in Form von Zusatzlizenzen (Recht zur Vervielfältigung der Software ohne Erwerb zusätzlicher Datenträger bzw. Handbücher) erwerben, durch die ihm zusätzliche Rechte zur Vervielfältigung eingeräumt werden.

Bei der Nutzung der "Server" Ausgabe innerhalb eines Netzwerkes oder eines sonstigen Mehrstationen-Rechensystems umfasst die Nutzungsbefugnis das Recht, die erworbene Software einmal auf einem einzigem Netzwerkserver-Computer zu installieren und in den Arbeitsspeicher so vieler Arbeitsstationen zu laden, wie in der erworbenen Lizenz vereinbart wurde. Der Einsatz der Software innerhalb mehrerer Netzwerke ist unzulässig, sofern nicht die dazu notwendige Anzahl Lizenzen erworben worden ist.

Der Kunde ist berechtigt, pro erworbene Lizenz an einer Vollversion die Datenträger je einmal zu Sicherungszwecken zu vervielfältigen.

Wechselt der Kunde an einem Arbeitsplatz oder einem Netzwerkserver die Hardware, auf der er die lizenzierte Software einsetzt, muss er die Software vom Massenspeicher der bisher verwendeten Hardware löschen. Ein zeitgleiches Einspeichern, Vorrätighalten oder Benutzen der Software auf mehr als nur einem Computer ist unzulässig. Die Software darf an Dritte übertragen werden, wenn die Software vollständig (mit Originaldatenträger und Originalhandbuch) und ohne Zurückbehaltung von Kopien weitergegeben wird und der Empfänger sich mit den Lizenzbestimmungen einverstanden erklärt und sich bei BE als neuer Benutzer registriert.

Dem Kunden ist untersagt, die Software zu dekompileieren, zurückzuentwickeln, zu disassemblieren oder auf andere Weise zu versuchen, den Quellcode der Software zugänglich zu machen, die Software zu ändern zu übersetzen oder davon abgeleitete Produkte zu erstellen.

Es ist untersagt, diese Software anderen Personen zu vermieten, zu verleasen oder für kommerzielle Zwecke aufzubewahren.

Die Verwendung der Dokumentation und der verschiedenen Hilfequellen ist nur für interne und nicht kommerzielle Zwecke erlaubt. Die mit der Software erstellten Information wie Blockdiagramme und Simulationsergebnisse, dürfen in Dokumentationen und Veröffentlichungen verwendet werden, sofern in angemessener und sichtbarer Weise auf SimApp als Ursprung hingewiesen wird.

Garantie

Büsser Engineering gewährleistet für sechzig (60) Tage ab Empfangsdatum, dass der Datenträger auf dem die Software ausgeliefert wird, keine Material- und/oder Herstellungsmängel aufweist. Wenn die Software dieser Garantie nicht genügt, hat der Kunde den Anspruch auf Ersatz, falls er die fehlerhaften Erzeugnisse mit einem Kaufnachweis innerhalb der Garantiezeit zurückgibt.

Die Software selber unterliegt keiner Garantie. In keiner Weise kann BE garantieren, dass die Software fehlerfrei ist und dass der Kunde sie ohne jegliche Probleme oder Unterbrechungen einsetzen kann. Es kann nicht garantiert werden, dass die Software unverwundbar gegen Angriffe bössartiger Software innerhalb des verwendeten Computersystems ist.

Haftungsausschluss

Büsser Engineering ist in keinem Falle ersatzpflichtig für irgendwelche indirekten Folge- oder ähnlichen Schäden (eingeschlossen sind Schäden aus entgangenem Gewinn oder Verlust von Daten), die aufgrund der Benutzung der Software entstehen.

Benutzerhandbuch

SimApp

Version 2.6

für Windows 2000/XP/Vista

Dieses Handbuch und die Software SimApp sind urheberrechtlich geschützt. Alle Rechte bleiben vorbehalten.

Das Kopieren oder Vervielfältigen als Ganzes ist gestattet, sofern keine Änderungen vorgenommen werden.

Ansprüche gegenüber Büsser Engineering über die Garantie hinaus besteht nicht. Insbesondere übernimmt Büsser Engineering keine Haftung für die Richtigkeit des Inhalts dieses Handbuchs. Änderungen an der Software bleiben vorbehalten.

Alle Warenzeichen werden ausschliesslich zu Informationszwecken genannt.

Ausgabe März 2009

Copyright © 1998-2009 by Büsser Engineering

Inhaltsverzeichnis

1	Einführung	1
1.1	Was ist SimApp?	1
1.2	Für wen ist SimApp?	1
1.3	Die Hilfe verwenden	1
1.3.1	Hilfeprogramm aufrufen	1
1.3.2	Direkthilfe	2
1.4	Technische Unterstützung	2
2	Installation.....	3
2.1	Lizenzvereinbarung.....	3
2.2	Systemvoraussetzungen.....	3
2.3	Installation	3
2.3.1	Internet-Download	3
2.3.2	CD	3
3	Das SimApp-Hauptfenster	4
3.1	Übersicht	4
3.2	Menüs.....	4
3.2.1	Hauptmenuleiste	4
3.2.2	Kontextmenüs	4
3.3	Symbolleisten und Schaltflächen	4
3.3.1	Dateileiste.....	4
3.3.2	Objektpalette	4
3.3.3	Symbolleisten	5
3.3.4	Bibliotheksleisten	5
3.4	Statusleiste.....	5
3.5	Fehlerfenster	5
4	Einführungsbeispiel.....	6
4.1	SimApp starten.....	6
4.2	Ansichten und Seitenanordnung wählen	6
4.3	System modellieren.....	6
4.3.1	Vom realen System zum Blockdiagramm.....	6
4.3.2	Glieder verbinden	7
4.3.3	Übertragungsparameter ändern	7
4.4	System simulieren.....	8
4.4.1	Zeitsimulation	8
4.4.2	Frequenzsimulation	9
4.4.3	Liste der Simulationen	10
5	Zeichnungsfunktionen.....	12
5.1	Einführung.....	12
5.2	Zeichnungsobjekte	12
5.2.1	Linien.....	12
5.2.2	Rechtecke und Quadrate.....	12
5.2.3	Abgerundete Rechtecke	12
5.2.4	Ellipsen und Kreise.....	12
5.2.5	Linienzüge	12
5.2.6	Polygone	13
5.2.7	Signalleitungen.....	13
5.2.8	Text	14
5.2.9	Bilder	14
5.2.10	Pfeil	15
5.3	Objekte formatieren.....	15
5.3.1	Formatierungssymbolleiste.....	15
5.3.2	Formateigenschaften.....	16
5.3.3	Standardformatierung.....	17
5.4	Objekte ändern und anordnen	17
5.4.1	Objekte drehen und spiegeln.....	17
5.4.2	Objekte ordnen.....	17
5.4.3	Objekte auf Raster verschieben	18
5.4.4	Objekte gruppieren.....	18
5.5	Wichtige Hilfstasten.....	18

6	Simulationsobjekte	20
6.1	Beschreibung	20
6.2	Funktionsglieder verbinden	20
6.2.1	Additionsstelle	21
6.2.2	Verzweigungsstelle	21
6.3	Direkteinstellungen	22
6.3.1	Titel verschieben und editieren	22
6.3.2	Parameter ändern	22
6.4	Simulationseigenschaften	22
6.4.1	Parametereigenschaften	23
6.4.2	Einheiten	23
6.4.3	Optionen	23
6.4.4	Glieder und Signale bezeichnen	24
7	Frequenzsimulation	25
7.1	Blockdiagramm erstellen, Modellieren	25
7.2	Frequenzsonden	25
7.3	Simulationsparameter	26
7.4	Frequenzsimulation starten	27
7.5	Resultate der Frequenzsimulation	27
7.5.1	Diagramme	27
7.5.2	Besondere Effekte	28
7.5.3	Eigenwerte	28
7.5.4	Datentabellen	29
7.5.5	Rapport	29
8	Zeitsimulation	31
8.1	Blockdiagramm	31
8.2	Quellen und Zeitsonden	31
8.3	Quellengruppierung	32
8.4	Weitere Bemerkungen zur Zeitsimulation	33
8.5	XY-Diagramme	33
8.6	Simulationsparameter	34
8.7	Zeitsimulation starten	35
8.8	Resultate der Zeitsimulation	35
8.8.1	Zeitdiagramm	35
8.8.2	Datentabellen	36
8.8.3	Rapport	36
8.9	Beispiele zur Zeitsimulation	37
8.9.1	Numerische Lösung von Differentialgleichungen	37
9	Parametervariation	38
9.1	Einstellungen zur Parametervariation	38
9.1.1	Simulationseigenschaften	38
9.1.2	Eingabefenster für die Parametersätze	39
9.1.3	Simulationseinstellungen	39
9.2	Parametervariation starten	39
10	Anwenderblöcke	41
10.1	Einfache Blöcke durch Auswählen erzeugen	41
10.2	Blöcke in der Blockmappe erzeugen	43
10.2.1	Einführungsbeispiel	43
10.2.2	Beziehung zwischen Symbol- und Strukturblatt	46
10.2.3	Die Parametertabelle	46
10.2.4	Innere Struktur entwerfen	46
10.2.5	Formeln für die inneren Parameter festlegen	46
10.2.6	Inneres System austesten	47
10.2.7	Knotenobjekte und Blockknoten	47
10.2.8	Blocksymbol entwerfen	48
10.2.9	Struktur und Symbol zusammenführen (exportieren)	48
10.2.10	Blöcke verwenden und abspeichern	48
10.2.11	Blöcke überarbeiten	49
10.2.12	Weitere Bemerkungen zum Entwurf von Anwenderblöcken	49
11	Palette bearbeiten	50

11.1	Seiten erzeugen, löschen und umbenennen.....	50
11.2	Reiter und Schaltflächen verschieben.....	50
11.3	Objekte in Palette speichern	50
11.3.1	Objekte aus Zeichnungen einfügen.....	50
11.3.2	Objekte aus Bibliotheken einfügen	50
11.4	Schaltflächen bearbeiten.....	51
11.4.1	Eigenschaften von Schaltflächen.....	51
11.4.2	Mit Microsoft Paint eigene Schaltsymbole erzeugen	51
11.5	Palette laden, speichern, wieder herstellen	52
12	Bibliotheken	53
12.1	Beispiel.....	53
13	Katalog der Standardglieder	55
13.1	Quellen	55
13.1.1	Konstante	55
13.1.2	Rampe.....	56
13.1.3	Schritt.....	56
13.1.4	Oszillator (Sinus, Dreieck, Rechteck).....	57
13.1.5	Pulsgenerator	57
13.1.6	Pulsweitenmodulator (PWM).....	58
13.1.7	Timer	58
13.1.8	Taktquelle.....	59
13.1.9	Fahrkurve	60
13.1.10	Rauschquelle, Zufallszahlengenerator	60
13.1.11	Programmierbare Quelle	61
13.1.12	Dateiquelle	63
13.2	Lineare Glieder.....	65
13.2.1	Summierglied (Addierer).....	65
13.2.2	Proportionalglied, Verstärkungsglied (P-Glied).....	66
13.2.3	Integrator (I-Glied).....	67
13.2.4	Differenzierer (D-Glied)	68
13.2.5	Vorhaltglied (DT ₁ Glied).....	69
13.2.6	Verzögerungsglied 1.Ordnung (PT ₁ -Glied).....	70
13.2.7	Verzögerungsglied 2.Ordnung (PT ₂ -Glied).....	71
13.2.8	Nicht schwingfähiges Verzögerungsglied 2. Ordnung (PT ₁ T ₂ -Glied).....	72
13.2.9	Verzögerungsglied n-ter Ordnung (PT _n -Glied).....	73
13.2.10	Lead/Lag-Glied.....	74
13.2.11	Rationales Übertragungsglied (G(s)-Glied).....	74
13.2.12	Totzeitglied (PTt-Glied).....	75
13.2.13	Allpassglied 1. Ordnung (PTa1-Glied).....	77
13.2.14	Allpassglied 2. Ordnung (PTa2-Glied).....	78
13.2.15	Lineares Differentialgleichungssystem	79
13.3	Nichtlineare Glieder.....	80
13.3.1	Quadrierglied.....	81
13.3.2	Radizierglied (Wurzel)	81
13.3.3	Inverter	81
13.3.4	Multiplizierglied (Produkt)	81
13.3.5	Dividierglied.....	82
13.3.6	Verknüpfungsglied.....	82
13.3.7	Funktionsglied mit einem Eingang.....	82
13.3.8	Funktionsglied mit zwei Eingängen	83
13.3.9	Kennlinienglied (KL-Glied).....	83
13.3.10	Sättigungsglied	83
13.3.11	Ansprechschwelle (Tote Zone).....	84
13.3.12	Vorlast (Offset)	84
13.3.13	Hysterese	84
13.3.14	Minimum/Maximum (MinMax).....	85
13.3.15	Extremwert	85
13.3.16	Haftreibung.....	86
13.4	Stellglieder.....	88
13.4.1	Anstiegsbegrenzung.....	88
13.4.2	Anstiegskonstante	88
13.5	Regler	89
13.5.1	Zweipunkt.....	89
13.5.2	Dreipunkt.....	89
13.5.3	Idealer PI-Regler (PI).....	90
13.5.4	Modifizierter PI-Regler (PI-m).....	91

13.5.5	Idealer PD-Regler (PD-Regler).....	92
13.5.6	Realer PD-Regler (PD-r)	93
13.5.7	Idealer PID-Regler, Typ I (PID-I)	94
13.5.8	Adaptiver PID-Regler.....	95
13.5.9	Industrie PID-Regler.....	96
13.5.10	Idealer PID-Regler, Typ II (PID-II)	98
13.5.11	Realer PID-Regler (PID-r)	99
13.5.12	Modifizierter PID-Regler (PIDm-Regler)	100
13.5.13	Verallgemeinerter PID-Regler (PID-a).....	101
13.5.14	Lead/Lag-Regler.....	101
13.6	Zeitdiskrete Übertragungsglieder	102
13.6.1	Einführung	102
13.6.2	Abtastglied.....	104
13.6.3	Halteglied	105
13.6.4	Abtast-/Halteglied	106
13.6.5	Zeitdiskreter Integrator (Iz)	107
13.6.6	Zeitdiskreter Differenzierer (Dz)	108
13.6.7	Einheitsverzögerung.....	109
13.6.8	Zeitdiskreter PID-Regler (PIDz).....	110
13.6.9	z-Übertragungsfunktion (G(z)-Glieder)	111
13.6.10	Zeitdiskretes Filter (z-Filter).....	111
13.6.11	Lineares Differenzgleichungssystem	112
13.7	Wandler	113
13.7.1	Analog/Digital Umsetzer (ADU, ADC).....	113
13.7.2	Digital/Analog Umsetzer (DAU, DAC).....	114
13.7.3	Analog/Binär Umsetzer (ABU, ABC).....	115
13.7.4	Binär/Analog Umsetzer (BAU, BAC).....	115
13.7.5	Quantisierer.....	116
13.8	Logik	117
13.8.1	GND (Ground, Logisch 0, Masse)	117
13.8.2	V+ (Logisch 1)	117
13.8.3	UND (AND).....	117
13.8.4	ODER (OR).....	118
13.8.5	Exklusiv-ODER (XOR, Antivalenz)	118
13.8.6	Inverter (NOT, Negation).....	118
13.8.7	SR Flip-Flop	119
13.8.8	JK Flip-Flop	119
13.8.9	D Flip-Flop.....	120
13.8.10	Monoflop.....	120
13.8.11	Verzögerter Monoflop.....	121
13.9	Verschiedenes	122
13.9.1	1:2 Schalter	122
13.9.2	2:1 Schalter	122
13.9.3	1:n Ausgangsschalter (Demultiplexer).....	123
13.9.4	n:1 Eingangsschalter (Multiplexer)	123
13.9.5	Steuerbares Abtast-/Halteglied.....	123
13.9.6	Steuerbares Verzögerungsglied	124
13.9.7	Vergleicher	124
13.9.8	Fensterkomparator	125
13.9.9	Nulldurchgangsdetektor.....	125
13.9.10	Schrittrampe	126
13.10	Spezialglieder	127
13.10.1	Sende- und Empfangsglied	127
14	Literaturhinweise.....	128

Über dieses Handbuch

In diesem Handbuch finden Sie eine Einführung in die Benutzung von SimApp und eine Beschreibung aller Standardglieder. Wenn Sie SimApp noch nicht oder nur oberflächlich kennen, raten wir Ihnen, es sorgfältig durchzulesen. Simulationsprogramme sind keine so weitgehend standardisierten Programme wie z.B. Text- oder Zeichnungsprogramme. Viele wichtige Details erfahren Sie nur, wenn Sie dieses Handbuch lesen. Hauptanliegen dieses Handbuches ist es, die in dieser Software verwendeten Konzepte und Werkzeuge vorzustellen und Ihnen das Rüstzeug für erfolgreiche Simulationen zu geben.

Referenzinformationen über die Benutzung bestimmter Werkzeuge und Befehle finden Sie dagegen ausschliesslich in der Online-Hilfe. Benutzen Sie die Direkthilfe , wenn Sie Informationen zu Eingabe-, Schalt- und Auswahlflächen benötigen. Sollten Sie mit einem bestimmten Werkzeug Schwierigkeiten haben, drücken Sie die Funktionstaste F1, die zuständige Hilfetaste oder klicken auf *Inhalt und Index* im Menü *Hilfe (?)*.

In diesem Handbuch wird vorausgesetzt, dass Sie mit den grundlegenden Konzepten von Windows vertraut sind. Im Weiteren ist es kein Lehrbuch über Regelungstechnik und Modellbildung. Bitte beachten Sie dazu das Literaturverzeichnis am Schluss.

Danksagungen

Ich möchte mich herzlich bei folgenden Persönlichkeiten bedanken, die massgebend zum Erfolg von SimApp beigetragen haben:

Prof. Dr. Ing. Ivan Vaclavik

Haute Ecole d'Ingénierie et de Gestion (HEIG-VD),
Institut d'Automatisation industrielle (iAi), <http://www.iai.heig-vd.ch/>
Route de Cheseaux 1
CH-1401 Yverdon-les-Bains, Schweiz

für seine langjährige fachliche, kommerzielle und nicht zuletzt moralische Unterstützung des Projektes.

Prof. Michel Etique, ing. dipl. EPFL

Haute Ecole d'Ingénierie et de Gestion (HEIG-VD) <http://www.heig-vd.ch>
Département des Technologies Industrielles (TIN)
Route de Cheseaux 1
CH-1401 Yverdon-les-Bains, Schweiz

für die Übersetzung der Programmtexte ins Französische, die ab Version 2 eine französische Programmoberfläche ermöglichte.

Prof. Michel Huguet agrégé de mécanique

Lycée Jacques Amyot d'Auxerre, <http://lyc89-amyot.ac-dijon.fr/>
Classes Préparatoires aux Grandes Ecoles (PCSI-PSI)
3 rue de l'étang St Vigile
F-89015 Auxerre, Frankreich

für die Übersetzung der Benutzeranleitung ins Französische und Redigierung sämtlicher französischer Texte ab Version 2.5. Dank der grossen Arbeit von Herrn Huguet ist nun SimApp vollständig in französischer Sprache erhältlich.

Peter Way, SimApp Marketing und Verkauf

VentiMar, LLC, Suite 104
Fort Collins, CO 80525, USA

für seine tatkräftige Unterstützung in allen Bereichen des Projekts und die Bearbeitung der englischen Texte.

Frühling 2009, Bruno Büsser

1 Einführung

1.1 Was ist SimApp?

SimApp ist ein Programm zur Untersuchung und Optimierung von dynamischen Systemen mittels Blockdiagrammen (auch Blockschaltbilder, Wirkungspläne, Wirkschaltpläne, Strukturbilder, Signalfusspläne genannt), wobei keine vorgegebenen Strukturen vorausgesetzt werden. Es können beliebige Blockdiagramme mit den vorhandenen Funktionsgliedern entworfen und simuliert werden. SimApp eignet sich vor allem für den Einsatz in der Steuer- und Regelungstechnik.

Lineare und nichtlineare, zeitinvariante und zeitvariante Systeme können im Zeitbereich simuliert und die Ergebnisse mittels Zeit- und XY-Diagrammen dargestellt werden. Für lineare und zeitinvariante Systeme besteht zusätzlich die Möglichkeit verschiedener Analysemethoden im Frequenzbereich. Es lassen sich Frequenzgänge (Bodediagramme) und Ortskurven (Nyquistdiagramme) sowie Black-Nicholsdiagramme erstellen und die Eigenwerte des Systems oder der untersuchten Teilsysteme berechnen. Die resultierenden Datenwerte können in tabellarischer und grafischer Form ausgedruckt und zur Weiterverarbeitung in andere Programme exportiert werden.

Die Modellierung der Systeme als Blockdiagramme erfolgt vollständig visuell durch die graphische Verknüpfung von Funktionsgliedern. Die Eingabe der wichtigsten Zahlenwerte wie Verstärkungen, Zeitkonstanten, Verzögerungszeiten etc. kann direkt in der Zeichnung ohne Öffnen von Dialogboxen vorgenommen werden. Zusätzlich zu den Funktionsgliedern können auch verschiedene grafische Formen (Kreise, Linien, Linienzüge, etc.) und Text in die Zeichnung eingefügt werden, die es gestatten die Modelle gestalterisch zu ergänzen und zu dokumentieren.

Die Objektpalette von SimApp enthält über 80 Funktionsglieder. Oft verwendete Teilsysteme können in Gruppen zusammengefasst und in der Palette oder in Bibliotheken abgespeichert werden. Es können aber auch eigene wieder verwendbar Blöcke, so genannte Anwenderblöcke, entworfen werden, die aus einem System von Grundgliedern und anderen Anwenderblöcken bestehen können. Anwenderblöcke unterscheiden sich in keiner Weise von den Grundgliedern. Sie enthalten ein Blocksymbol, Ein- und Ausgangsknoten und eine Parameterliste. Mittels Anwenderblöcken kann SimApp beliebigen Anwendungsgebieten angepasst werden.

Das Zeit- und Frequenzverhalten kann an jedem beliebigen Schaltungsknoten erfasst und analysiert werden. Für erweiterte Anwendungen stehen spezielle Messsonden zur Verfügung. Z.B. kann der Frequenzgang für den offenen und geschlossenen Regelkreis in einem Durchgang erfasst und miteinander verglichen werden. Eine spezielle Sonde gestattet die zweidimensionale Darstellung im Zeitbereich.

SimApp ist eine MDI-Anwendung, d.h. es erlaubt, wie bei einem Textverarbeitungsprogramm, mit mehreren Dokumenten gleichzeitig zu arbeiten und Simulationen durchzuführen.

1.2 Für wen ist SimApp?

SimApp eignet sich gleichermaßen für Lehrlinge, Studenten, Ingenieure, Techniker, Mathematiker und Naturwissenschaftler, die das dynamische Verhalten von Systemen untersuchen wollen und Wert auf einfache, intuitive Bedienung legen.

SimApp ist sehr einfach zu bedienen und ermöglicht dem Anfänger, auch ohne diese Anleitung zu lesen, brauchbare Resultate in wenigen Minuten zu erzielen.

1.3 Die Hilfe verwenden

In SimApp werden Sie durch verschiedene Hilfe-Mechanismen unterstützt. Sie haben die Möglichkeit, ein spezielles Hilfeprogramm aufzurufen, das Ihnen die Informationen in strukturierter Form anzeigt. Mit der Direkthilfe können Sie im Weiteren spezifische Informationen zu den einzelnen Eingabe-, Schalt- und Auswahlflächen in Dialogboxen aufrufen. Ausserdem ist auch dieses Handbuch in den Hilfetexten enthalten.

1.3.1 Hilfeprogramm aufrufen

Das Hilfeprogramm starten Sie über das *Menu ? + Inhalt* und *Index*. Sie haben nun die Möglichkeit, Ihre Informationen über das Inhaltsverzeichnis zu erhalten. Diese Hilfe eignet sich, wenn Sie sich für bestimmte Themenbereiche interessieren. Sie eignet sich aber nicht, wenn Sie möglichst schnell spezifische Informationen über eine bestimmte Eingabe- oder Schaltflächen in Dialogboxen erhalten möchten. Benutzen Sie dazu die Direkthilfe.

1.3.2 Direkthilfe

Die Direkthilfe ist überall dort verfügbar, wo Sie in der Titelzeile eines Fensters das ?-Zeichen oder innerhalb des Fensters die Schaltfläche  entdecken. Klicken Sie auf das ?-Zeichen oder die Schaltfläche und danach in ein Eingabe- oder Schaltfeld. Dadurch öffnet sich ein der Textgrösse angepasstes Fenster, das die wichtigsten Information zu diesem Feld enthält. Sie können auch, wenn die Maus innerhalb eines Eingabefeldes oder auf einem Schaltfeld ist, die F1-Taste drücken und erhalten so dasselbe Resultat. Es existieren Felder, die sehr viel Informationen aber auch solche die keine Informationen enthalten. Der Umfang hängt vom jeweiligen Bedarf an zusätzlichen Erläuterungen ab.

1.4 Technische Unterstützung

Sollten Probleme oder Fragen beim Arbeiten mit SimApp auftreten, so schlagen Sie zunächst im Handbuch nach oder benutzen Sie die Online-Hilfe.

Sie können uns natürlich auch jederzeit per **Email** erreichen:

support@simapp.com

Wenn Sie Fragen, Anregungen oder Kritik haben, zögern Sie bitte nicht, uns zu kontaktieren. Wir freuen uns über jede Zuschrift und werden sie umgehend und kompetent beantworten.

Beachten Sie bitte auch unsere **Website**

<http://www.simapp.com>

Auf unserer Homepage haben Sie die Möglichkeit, den jeweiligen neusten Release Ihrer registrierten Version gratis zu downloaden. Dazu benötigen Sie den Benutzernamen und das Passwort, die Sie beim Kauf von SimApp erhalten haben.

Unsere Anschrift lautet:

Büsser Engineering
Wacht 28
CH-8630 Rüti ZH
Schweiz
Tel. +41 (0)55 260 12 88
Fax +41 (0)55 260 12 89

2 Installation

2.1 Lizenzvereinbarung

Bevor Sie SimApp auf Ihrem Computer installieren, sollten Sie die Lizenzvereinbarung am Anfang dieses Handbuches aufmerksam durchlesen und prüfen, ob Sie mit diesen Bedingungen einverstanden sind.

Bitte beachten Sie, dass technische Programme nur einen begrenzten Anwenderkreis besitzen und deren Erstellung wegen des benötigten Fachwissens aufwändig und teuer sind. Dank Ihrer Unterstützung und Ehrlichkeit ermöglichen Sie uns, das Produkt ständig zu verbessern und Ihren steigenden Anforderungen anzupassen. Bitte helfen Sie uns dabei!

2.2 Systemvoraussetzungen

SimApp läuft unter Windows 2000, XP und Vista, ansonsten benötigt es keine speziellen Systemvoraussetzungen. Vorteilhaft ist es, wenn Sie einen neueren Computer einsetzen und für längere und hochauflösende Simulationen genügend Arbeitsspeicher besitzen.

2.3 Installation

Bevor Sie SimApp ausführen können, müssen Sie die Software zuerst durch Ausführung des Setup-Programms installieren. Sie haben zwei Möglichkeiten, diese Datei zu erhalten:

- Laden Sie das Programm vom Internet herunter
- Bestellen Sie die optionale CD

2.3.1 Internet-Download

Sie können SimApp jederzeit von unserer Homepage www.simapp.com downloaden. Bitte benutzen Sie den Benutzernamen und das Passwort, die Sie beim Kauf erhalten haben.

Die Datei, die Sie auf Ihren PC herunterladen, ein Installationsprogramm. Sie müssen es ausführen, um SimApp permanent auf Ihrem System zu installieren. Es ist nicht möglich und auch nicht zulässig nur die Exe-Datei von einem anderen Computer zu kopieren.

Am einfachsten ist es, wenn Sie die Datei auf Ihren Desktop herunterladen. Dadurch wird ein Symbol auf Ihrem Windows-Desktop erstellt. Sie können dann das Installationsprogramm ausführen, indem Sie auf das Desktop-Symbol doppelklicken. Sie können das Installationsprogramm auch mit dem Befehl *Ausführen* im Windows Startmenu starten.

2.3.2 CD

Wenn Sie SimApp auf einer CD erworben haben, müssen Sie nur die CD in Ihr CD-Laufwerk einlegen. Das Programm **setup.exe** startet automatisch. Sollte es nicht starten, müssen Sie setup.exe manuell ausführen.

Zur manuellen Ausführung des Setup-Programms benutzen Sie den Befehl *Ausführen*, den Sie über die Windows-Schaltfläche *Start* finden. Durch *Ausführen* erscheint ein Dialogfeld, in dem Sie den Namen des Programms spezifizieren müssen, das Sie ausführen wollen. Wenn Sie die SimApp-CD z.B. in Laufwerk D eingelegt haben, geben Sie ein:

D:\Setup

Folgen Sie danach den Anweisungen des Setup-Programms.

Das Setup-Programm können sie natürlich auch durch Doppelklick innerhalb des Explorers starten.

3 Das SimApp-Hauptfenster

3.1 Übersicht

Nach dem Programmstart öffnet sich das Hauptfenster von SimApp. Im Dialogfeld *Extras+Optionen* können Sie wählen, ob die geöffneten Dateien der letzten Arbeitssitzung automatisch wieder geöffnet werden sollen, ansonsten wird ein leeres Zeichnungsformular erstellt.

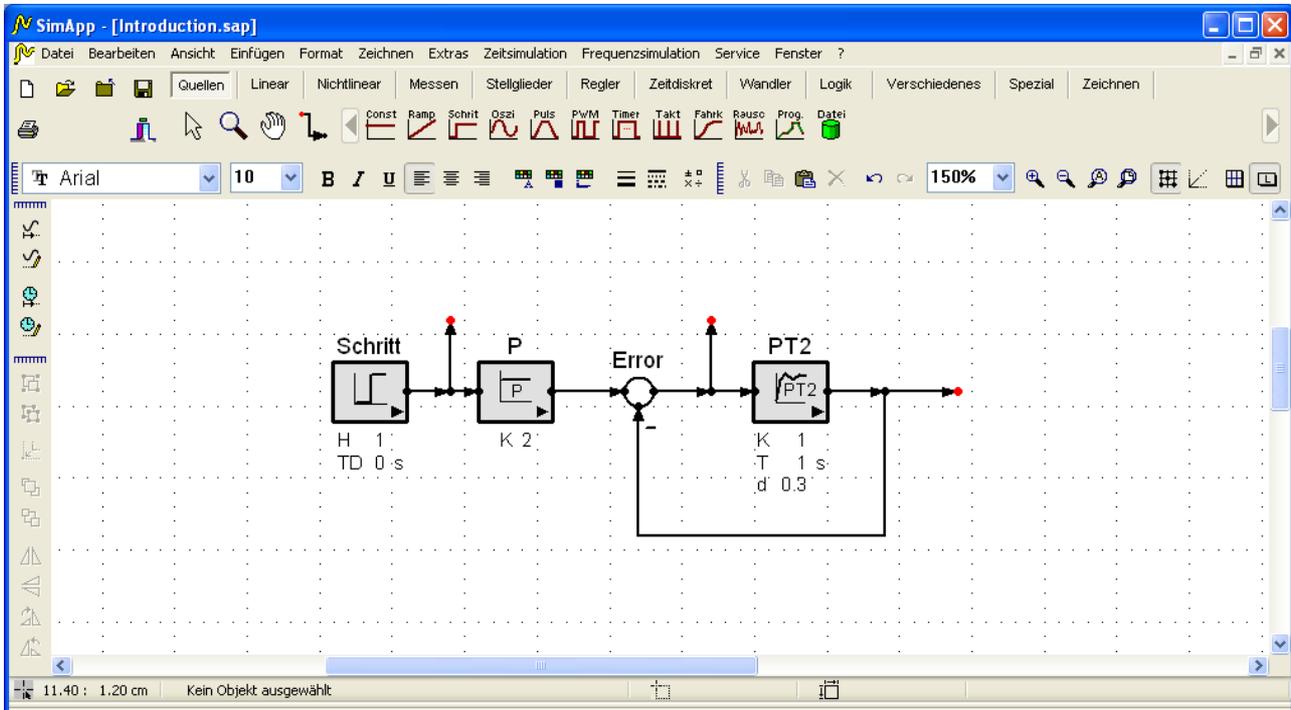


Abbildung 1: SimApp-Hauptfenster

3.2 Menüs

3.2.1 Hauptmenuleiste

In der Hauptmenuleiste am oberen Fensterrand finden Sie die meisten zur Verfügung stehenden Befehle. Einige wenige objekt- oder kontextbezogene Befehle befinden sich aber nur in den Kontextmenüs.

3.2.2 Kontextmenüs

Um das Kontextmenu zu öffnen, klicken Sie die rechte Maustaste. Sie erhalten daraufhin eine Liste mit den am häufigsten benötigten Befehlen. Abhängig von der Position des Cursors auf dem Bildschirm, stehen verschiedene Kontextmenüs zur Auswahl. Wenn sich Ihr Cursor z.B. über einem Zeichnungsobjekt befindet, enthält das Kontextmenu spezifische Bearbeitungsbefehle für dieses Objekt. Es existieren aber auch spezielle Kontextmenüs für die Objektpalette, die Symbolleisten und die Schaltflächen.

3.3 Symbolleisten und Schaltflächen

3.3.1 Dateileiste

Die Symbolleiste in der linken oberen Ecke des Hauptfensters enthält Schaltflächen für die wichtigsten Dateifunktionen (*Speichern*, *Öffnen*, *Drucken*) und das Schliessen von SimApp. Diese Leiste kann nicht verschoben sondern nur mit der Palette zusammen ein- und ausgeblendet werden.

3.3.2 Objektpalette

Die Objektpalette stellt alle Standardobjekte zur Verfügung, die Sie in Ihre Zeichnungen einfügen können. Sie besteht aus mehreren Registerseiten, die Sie mit den Reitern einzeln auswählen können.



Abbildung 2: Objektpalette

Es stehen grundsätzlich zwei Objekttypen zur Verfügung:

1. Funktionsglieder
Diese Objekte benötigen Sie, um Ihre Simulationsmodelle (Blockdiagramme) zu erstellen.
2. Zeichnungsobjekte (Register *Zeichnen*)
Mit SimApp lassen sich einfache Formen wie Linien, Rechtecke, Text, etc. erstellen, die sie zum Illustrieren und Dokumentieren der Blockdiagramm einsetzen können. Sie werden sie ausserdem brauchen, um die Symbole für Ihre eigenen Blöcke zu entwerfen.

Auf der linken Seite in der Palette immer sichtbar sind die Werkzeuge für Auswählen, Zoomen und Verschieben sowie die häufig benutzte Signallinie.

Die Palette lässt sich durch beliebige weitere Objekte ergänzen (siehe Kapitel [Palette bearbeiten](#)).

3.3.3 Symboleisten

In den diversen Symboleisten werden die wichtigsten Programmbefehle als Schnellasten zur Verfügung gestellt. Jede Symboleiste enthält eine bestimmte Kategorie von Befehlen. Sie können beliebig auf dem Bildschirm positioniert oder an den Rändern des Innenbereichs des Hauptfensters angedockt werden und sind einzeln ein- und ausblendbar (siehe Menü *Ansicht Symboleisten*). Wenn Sie auf eine Symboleiste doppelklicken, können Sie diese schnell zwischen schwebendem und angedocktem Zustand umschalten.

3.3.4 Bibliotheksleisten

In SimApp lassen sich eigene Objektbibliotheken erzeugen (siehe Kapitel [Bibliotheken](#)). Jede erzeugte und geöffnete Bibliothek besitzt eine eigene Symboleiste, deren Schaltflächen zum Auswahl der Objekte dienen. Bibliotheksleisten verhalten sich wie normale Symboleisten, können aber zusätzlich geöffnet (Menü *Extras + Bibliothek öffnen*) und geschlossen werden.

3.4 Statusleiste

Die Statusleiste zeigt die aktuelle Cursorposition sowie Informationen über die ausgewählten Objekte.

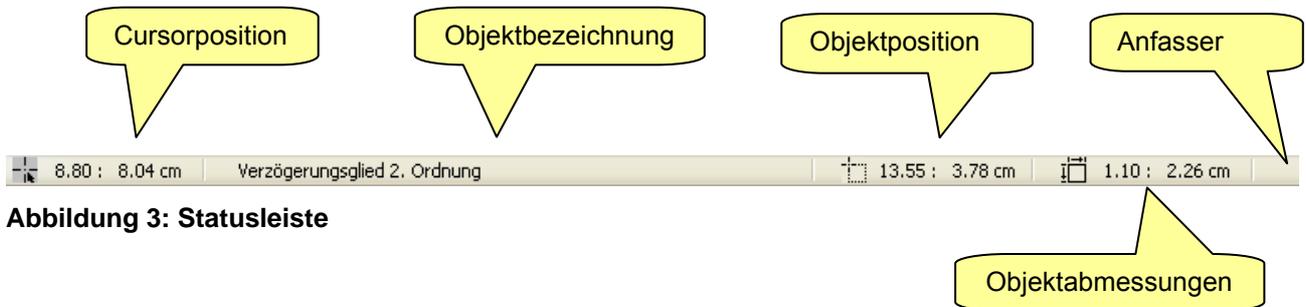


Abbildung 3: Statusleiste

3.5 Fehlerfenster

Wenn SimApp während der Simulation auf Fehler im Simulationsmodell oder bei den Einstellungen stösst, öffnet sich zwischen Arbeitsbereich und Statusleiste ein Fehlerfenster, das die Fehler kurz beschreibt. Durch Klicken mit der Maus auf eine Fehlerbeschreibung wird der entsprechende Block im Simulationsmodell markiert, der mit dem Fehler in Zusammenhang steht.

4 Einführungsbeispiel

In diesem Kapitel werden Sie Schritt um Schritt durch ein kleineres Projekt geführt und lernen dabei die Grundlagen zur Durchführung einfacher Simulationen. Die Beschreibung von Frequenz- und Zeitsonden, XY-Diagrammen und der Parametervariation folgt in den darauf folgenden Kapiteln.

4.1 SimApp starten

Sie können SimApp über die Taskleiste und das Startmenu aufrufen. Sie finden SimApp standardmässig unter *Start + Programme + SimApp*. Rufen Sie SimApp durch Anklicken auf. Es öffnet sich das SimApp-Hauptfenster mit einem leeren Zeichnungsdokument. Das Zeichnungsdokument ist maximiert und füllt den ganzen Innenbereich aus. SimApp kann jedoch auch mehrere Dokumente gleichzeitig enthalten, die unterschiedlich angeordnet werden können. Die Dokumente sind völlig unabhängig voneinander. In jedem Dokumentfenster können Sie ein Blockdiagramm entwerfen und simulieren. Ein zusammenhängendes System kann nicht über mehrere Dokumente verteilt werden. Ein Dokument kann jedoch aus mehreren zusammenhängenden Seiten im Hoch- oder Querformat bestehen. Neue Zeichnungen erscheinen im Querformat. Die Seitenränder sind als blau gestrichelte Linien erkennbar.

4.2 Ansichten und Seitenanordnung wählen

Mit verschiedenen Befehlen können Sie die Grösse und Lage Ihrer Zeichnungen Ihren Wünschen anpassen. Die Befehle zum Anpassen der Grösse und Lage der Zeichnungen sind als Schaltflächen in der Standard-Symbolleiste erreichbar.



Abbildung 4: Ansicht einstellen

Folgende drei Buttons bedürfen einer näheren Betrachtung:



Einrasten sollte immer eingeschaltet sein, damit das Ausrichten und Verbinden von Objekten leichter ist. Drücken Sie die Alt-Taste, um Objekte unabhängig vom Raster zu verschieben.



Linien ausrichten ermöglicht beim Zeichnen das automatische Ausrichten von Linien in 45 Grad Schritten.



Seitenanordnung ermöglicht grosse Blockdiagramme auf mehr als nur einer einzigen Seite darzustellen. Jede Seite entspricht einer Druckseite.

4.3 System modellieren

Das reale System wird in SimApp als Blockdiagramm (auch Blockschaltbild, Wirkungsplan, Wirkschaltplan, Strukturbild, Signalflussplan oder Modell genannt) abgebildet. Das Blockdiagramm besteht im Wesentlichen aus Blöcken (Funktionsgliedern) und Signalleitungen. Die Blöcke charakterisieren die Übertragungsglieder, die die Grössen des Systems verändern oder auch neue Signale erzeugen können. Sie werden mittels Signalleitungen miteinander verbunden und beschreiben auf diese Weise den Signalfluss innerhalb des Systems. Jede Signalleitung charakterisiert eine Systemgrösse, deren Wirkrichtung durch einen Pfeil am Ende angedeutet wird.

4.3.1 Vom realen System zum Blockdiagramm

Wenn Ihr System bereits als mathematische Beschreibung oder gar als Blockdiagramm vorliegt, haben Sie Glück, denn Sie können gleich mit dem Entwurf beginnen, andernfalls müssen Sie noch einiges an Vorarbeit leisten. Sie müssen Ihr reales System zuerst analysieren, gegen aussen abgrenzen, in Teilsysteme zerlegen und geeignete mathematische Zusammenhänge (Differentialgleichungen) finden. Wenn die mathematische Beschreibung des Systems vorliegt, können Sie dieses mit den vorhandenen Funktionsgliedern aus der SimApp-Objektpalette in einer grafischen Form - als Blockdiagramm eben - darstellen und simulieren. Die Schritte vom komplexen Gesamtsystem zum geeigneten Blockdiagramm sind jedoch nicht Gegenstand dieses Handbuchs. Wir verweisen dazu auf die umfangreiche Literatur über System- und Regelungstechnik. Einige grundlegende Werke finden Sie im [Literaturverzeichnis](#) am Schluss dieses Handbuchs.

Alle Funktionsglieder befinden sich nach verschiedenen Kategorien sortiert in der Objektpalette. Als Übungsbeispiel werden wir nun das folgende einfache Regelungssystem erfassen:

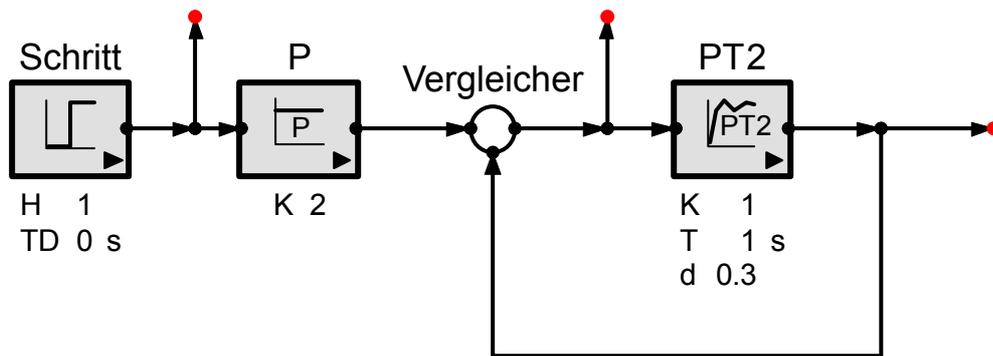


Abbildung 5: Regelkreis mit Schrittanregung

Überprüfen Sie vor dem Zeichnen unbedingt, ob das automatische Einrasten an den Gitterpunkten eingeschaltet ist (#-Schaltfläche). Holen Sie ein Objekt aus der Objektpalette, indem Sie zuerst die entsprechende Kategorie durch Klicken auf ihren Reiter in den Vordergrund stellen und danach die Schaltfläche für das gewünschte Objekt anklicken (Maustaste drücken und loslassen). Positionieren Sie die Maus in den Zeichnungsbereich und drücken Sie wiederum die linke Maustaste. Sie können nun das Objekt durch Ziehen korrekt positionieren und die linke Maustaste an der gewünschten Stelle wieder loslassen.

4.3.2 Glieder verbinden

Die Schaltfläche für die Signalleitungen oder Verbindungslinien finden Sie im linken, immer sichtbaren Teil der Objektpalette. Die Benutzung dieser Schaltfläche ist jedoch nur in den wenigsten Fällen notwendig, da sich das Werkzeug zum Ziehen der Signalleitungen automatisch aktiviert, sobald Sie mit dem Cursor über einen Knoten eines Funktionsgliedes fahren und das Glied nicht selektiert ist. Wenn das Glied selektiert ist, können sie zuerst neben das Glied klicken, so dass die Markierung verschwindet.

Ziehen Sie die Signalleitung zu einem anderen Knoten und lassen Sie die Maustaste los. Das Ende der Signalleitung wird dann automatisch mit dem Zielknoten verbunden. Um einen geordneten Eindruck zu erzeugen oder wenn die Signalleitung gar quer durch ein Funktionsglied führt, sollten Sie nur waagrechte und senkrechte Segmente ziehen, indem Sie Ecken einfügen. Ecken erhalten Sie, wenn Sie beim Ziehen kurz die Maustaste loslassen und danach gleich wieder drücken. Es ist auch möglich, noch nachträglich an beliebigen Stellen neue Ecken einzufügen oder bestehende zu löschen (siehe Kapitel [Funktionsglieder verbinden](#)).

Wenn der Zielort kein Knoten ist, wird die Signalleitung durch Loslassen der Maustaste nicht automatisch abgeschlossen. In diesem Fall müssen Sie mit der Maus doppelklicken.

Beachte: Alle Glieder haben eine bestimmte Wirkrichtung, die mit dem Pfeil im Symbol des Gliedes angedeutet wird. Bevor sie die Glieder miteinander verbinden, sollten Sie sie demnach korrekt ausrichten. Dies macht das Verbinden wesentlich einfacher. Die Wirkrichtung der Signalleitungen ist ebenfalls mit einem Pfeil gekennzeichnet. Die Wirkrichtungen von Funktionsglied und Signallinien müssen immer übereinstimmen, ansonsten lassen sich die Signallinien nicht mit dem Glied verbinden. Die Wirkrichtung der Signallinien können Sie über das Kontextmenu bei Bedarf umkehren.

4.3.3 Übertragungsparameter ändern

Nachdem Sie nun das Blockdiagramm gezeichnet haben, ändern Sie nun die Dämpfung d des PT2-Gliedes auf den Wert 0.3. Sie haben zwei Möglichkeiten, um dies auszuführen:

Ändern von d direkt in der Zeichnung

Klicken Sie mit der Maus direkt auf den numerischen Wert von d (nicht auf das Formelzeichen). Am Ort des Wertes erscheint ein Editerrahmen und der bisherige Zahlenwert wird markiert. Geben Sie den neuen Wert ein und drücken Sie die Eingabe(Enter)-Taste.

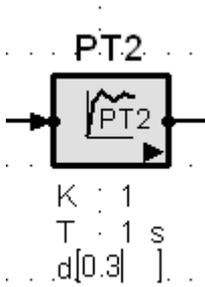


Abbildung 6: Parameter ändern

Ändern von d im Eigenschaftendialog

Doppelklicken Sie mit der rechten Maustaste auf das PT2-Glied oder wählen Sie im Kontextmenu *Simulationseigenschaften*. Im Eingabefeld für die Dämpfung geben Sie nun den neuen Wert ein und drücken die Eingabe-Taste. Beachten Sie, dass Sie im Dialog noch mehr Parameter verändern können, als in der Zeichnung sichtbar sind.

Verzögerungsglied 2. Ordnung (PT2)

Parameter | Optionen

$T^2\ddot{y} + 2dT\dot{y} + y = Ku$

$h(t)$ $G(s) = \frac{K}{1 + 2dT_s + T^2s^2}$

Anzahl Parametersätze: 0

Allgemein

Verstärkung	K	1	<input type="checkbox"/> multipel
Zeitkonstante	T	1 s	<input type="checkbox"/> multipel
Dämpfung	d	0.3	<input type="checkbox"/> multipel
Anfangswert $y(t=0)$	Y0	0	<input type="checkbox"/> multipel
Anfangssteigung	Y0'	0	<input type="checkbox"/> multipel

Zeitdiskret

Zeitdiskrete Simulation:

Abtastintervall T_s : 0.1 s multipel

... in alle zeitdiskreten Glieder übernehmen:

Integrationsregel: Vorwärts Rechteck

Buttons: OK, Abbrechen, Handbuch

Abbildung 7: Simulationseigenschaften des PT2-Gliedes

4.4 System simulieren

Nachdem Sie das Übungssystem korrekt erfasst haben, können Sie bereits die ersten Simulationen durchführen.

4.4.1 Zeitsimulation

Wir untersuchen zuerst, wie das System auf eine Schrittanregung reagiert. Dazu drücken wir die Starttaste für die Zeitsimulation. Es erscheint sofort ein weiteres Fenster, das aus mehreren Tabs besteht. Auf dem obersten Tab ist ein Diagramm, das den zeitlichen Verlauf der Signale an den Ausgangsknoten des Systems darstellt. Standardmässig sind die Kurven nach dem Block benannt, der das Signal generiert hat.

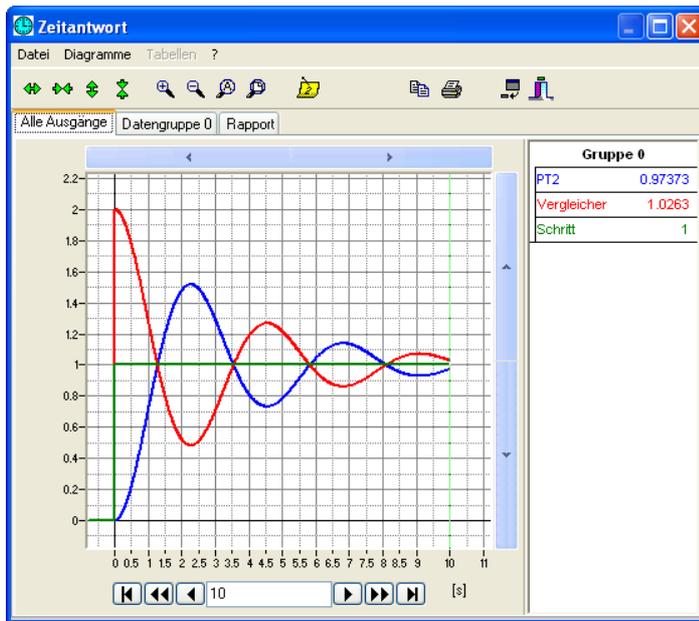


Abbildung 8: Zeitdiagramm

Auf der zweiten Ebene werden die numerischen Simulationsergebnisse in Tabellenform dargestellt und auf der Dritten finden Sie eine kurze Zusammenfassung der wichtigsten Eckdaten der Simulation. Jede Ebene hat verschiedene Bearbeitungs- und Darstellungsmöglichkeiten. Alle Inhalte können Sie auch ausdrucken und in die Windows-Zwischenablage kopieren.

Im Diagramm lässt sich die senkrechte Messlinie mit der Maus oder mit den Pfeilschaltflächen verschieben. Den aktuellen Zeitwert sehen Sie in der Box unterhalb des Diagramms und die zugehörigen Signalwerte in der Legende. Einen bestimmten Ausschnitt des Diagramms können Sie vergrößern, indem Sie mit der Maus einen rechteckigen Markierungsrahmen ziehen. Kurven bearbeiten Sie, indem Sie mit der rechten Maustaste direkt darauf oder in den zugehörigen Legendeneintrag klicken.

Die Werte in der Datentabelle lassen sich markieren und über die Zwischenablage in andere Programme kopieren. Die Spaltenbreiten lassen sich mit der Maus verändern.

4.4.2 Frequenzsimulation

Das vorliegende Regelungssystem besteht ausschliesslich aus linearen Gliedern. Sie können es also ohne Änderungen im Frequenzbereich simulieren. Die Ausgangsleitung zwischen Schritt und P-Glied wird nun zum Eingang, da für die Frequenzsimulation die Quellen nicht benutzt werden. Falls dieses Leitungsstück fehlt, können Sie auch die Quelle löschen, so dass der Knoten am Quellenausgang zum Eingang wird. Das Fenster der vorausgehenden Zeitsimulation sollten Sie übungshalber noch nicht schliessen.

Drücken Sie nun die Starttaste für die Frequenzsimulation . Es erscheint sogleich ein neues Fenster, das dem Vorausgehenden sehr ähnlich ist.

Das Fenster enthält, wieder auf verschiedene Tabs verteilt, ein Bodediagramm, ein Nicholsdiagramm, ein Nyquistdiagramm, eine Tabelle mit den Eigenwerten der beiden Übertragungstrecken (Eingang bis Vergleicherausgang sowie Eingang bis PT2-Ausgang), eine Tabelle mit allen Simulationsergebnissen und am Schluss wieder die Zusammenfassung.

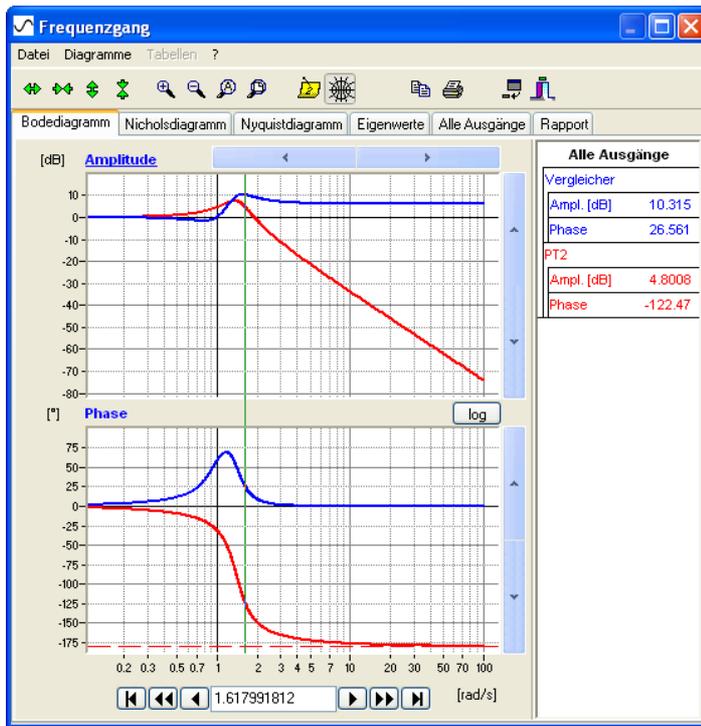


Abbildung 9: Bodediagramm

	Frequenz [rad/s]	Bode Vergleicher Ampl. [dB]	Bode Vergleicher Phase	Nyquist Vergleicher Realteil	Nyquist Vergleicher Imag. teil	Bode PT2 Ampl. [dB]	Bode PT2 Phase	Nyquist PT2 Realteil	Nyquist PT2 Imag. teil
0	0.1	-0.031781	1.7412	0.99589	0.030275	0.039592	-1.727	1.0041	-0.030275
1	0.10233	-0.033283	1.7829	0.99569	0.030993	0.041462	-1.7676	1.0043	-0.030993
2	0.10471	-0.034855	1.8256	0.99549	0.031729	0.043419	-1.8092	1.0045	-0.031729
3	0.10715	-0.036503	1.8694	0.99528	0.032484	0.04547	-1.8518	1.0047	-0.032484
4	0.10965	-0.038228	1.9142	0.99505	0.033257	0.047618	-1.8954	1.0049	-0.033257
5	0.1122	-0.040035	1.9603	0.99482	0.034049	0.049867	-1.9401	1.0052	-0.034049
6	0.11482	-0.041928	2.0075	0.99457	0.034861	0.052223	-1.9858	1.0054	-0.034861
7	0.11749	-0.04391	2.0559	0.99432	0.035694	0.05469	-2.0327	1.0057	-0.035694
8	0.12023	-0.045987	2.1056	0.99405	0.036547	0.057274	-2.0807	1.006	-0.036547
9	0.12303	-0.048162	2.1565	0.99377	0.037421	0.059981	-2.1298	1.0062	-0.037421
10	0.12589	-0.05044	2.2088	0.99347	0.038318	0.062816	-2.1802	1.0065	-0.038318
11	0.12882	-0.052827	2.2624	0.99316	0.039237	0.065785	-2.2317	1.0068	-0.039237
12	0.13183	-0.055327	2.3175	0.99284	0.04018	0.068895	-2.2846	1.0072	-0.04018
13	0.1349	-0.057946	2.374	0.9925	0.041147	0.072153	-2.3387	1.0075	-0.041147
14	0.13804	-0.060689	2.432	0.99214	0.042138	0.075565	-2.3941	1.0079	-0.042138
15	0.14125	-0.063562	2.4916	0.99177	0.043155	0.079139	-2.4509	1.0082	-0.043155
16	0.14454	-0.066573	2.5527	0.99138	0.044199	0.082883	-2.5092	1.0086	-0.044199
17	0.14791	-0.069726	2.6155	0.99097	0.045269	0.086804	-2.5688	1.009	-0.045269
18	0.15136	-0.07303	2.6801	0.99054	0.046367	0.090912	-2.6299	1.0095	-0.046367

Abbildung 10: Datentabelle

4.4.3 Liste der Simulationen

Für jede Simulation wird ein Diagrammfenster wie oben beschrieben erzeugt. Sie können diese Fenster zu Vergleichszwecken stehen lassen oder gleich wieder löschen. Beim Löschen gehen jedoch auch alle Simulationsdaten verloren. Sie können die Diagrammfenster aber auch vorübergehend verbergen, so dass sie das Hauptfenster nicht überdecken. Der Fenstertitel lässt sich ebenfalls verändern.

Eine Übersicht über die offenen Diagrammfenster zeigt ein kleines verschiebbares Fenster in der linken oberen Ecke des Zeichnungsbereichs, das in Form einer Liste den Simulationstyp (Zeit oder Frequenz) und die Fenstertitel anzeigt. Klicken Sie auf die entsprechende Zeile in der Liste, wenn Sie ein bestimmtes Ergebnisfenster in den Vordergrund stellen oder wieder sichtbar machen möchten.



Abbildung 11: Simulationsliste

5 Zeichnungsfunktionen

5.1 Einführung

SimApp weist grosse Ähnlichkeiten zu einem Zeichnungsprogramm auf, wobei aber der Funktionsumfang auf die spezifischen Anforderungen eines Simulationsprogramms beschränkt ist. Zusätzlich zu den Funktionsgliedern und Signalleitungen können einfache Formen, wie Rechtecke, Linien, etc. und Text eingefügt werden. Diese Objekte dienen v. a. zur Illustration und Beschreibung der Blockdiagramme und zum Entwerfen der Symbole der Anwenderblöcke.

5.2 Zeichnungsobjekte

Mit dem Reiter *Zeichnen* in der Palette bringen Sie die Zeichnungswerkzeuge zum Vorschein.



Abbildung 12: Zeichnungsobjekte in der Palette

Die meisten Objekte, wie Rechtecke und Linien, werden durch Klicken und Ziehen mit der Maus erzeugt. Die Zeichnungsaktion wird automatisch abgeschlossen, sobald Sie die linke Maustaste loslassen. Danach werden blaue Bearbeitungsgriffe gezeigt, die es Ihnen erlauben, die Grösse des Objektes zu verändern. Durch Klicken in die Mitte können Sie das Objekt auch verschieben. Linienzüge und Polygone verhalten sich ein wenig anders, wie weiter unten beschrieben wird.

Jedes Objekt besitzt ein Kontextmenu, das Sie durch Klicken mit der rechten Maustaste öffnen können.

5.2.1 Linien

Linien werden durch Klicken und Ziehen gezeichnet.

5.2.2 Rechtecke und Quadrate

Rechtecke und Quadrate werden durch Klicken und Ziehen gezeichnet. Exakte Quadrate lassen sich durch gleichzeitiges Drücken der Umschalt-Taste erreichen.

5.2.3 Abgerundete Rechtecke

Abgerundete Rechtecke werden wie normale Rechtecke gezeichnet. Sie können jedoch auch ein normales Rechteck in ein abgerundetes Rechteck verwandeln und umgekehrt, indem Sie mit der rechten Maustaste das Rechteck anklicken und im Kontextmenü *abgerundete Ecken* auswählen.

5.2.4 Ellipsen und Kreise

Ellipsen und Kreise werden durch Klicken und Ziehen gezeichnet. Exakte Kreise lassen sich durch gleichzeitiges Drücken der Umschalt-Taste erreichen.

5.2.5 Linienzüge

Linienzüge bestehen aus mehreren geraden Teillinien (Segmente), die über Knoten oder Ecken miteinander verbunden sind. Einen Linienzug können Sie jederzeit zu einem Polygon schliessen.

Starten Sie einen Linienzug durch Drücken der linken Maustaste und ziehen Sie das erste Segment. Am Ende des Segments, wo Sie eine Ecke einfügen möchten, lassen Sie die Maustaste kurzzeitig los und ziehen bei gedrückter Maustaste das zweite Segment. Ein Linienzug wird abgeschlossen, indem Sie mit der Maus am Endpunkt doppelklicken. Um aus einem Linienzug ein Polygon zu erzeugen, wählen Sie im Kontextmenü den Menüpunkt *Geschlossen verbinden*. Auf die gleiche Weise können Sie die Verbindung auch wieder auftrennen.

Nach dem Abschliessen eines Linienzuges haben Sie die Wahl zwischen zwei Bearbeitungsmodi:

1. Objektbearbeitung:

Das Objekt wird als Gesamtes markiert. Sie können es in der Grösse verändern und verschieben. Ecken und Endpunkte können Sie aber nicht einzeln verschieben. Dieser Modus ist automatisch eingestellt, wenn Sie das Ziehen des Objektes abgeschlossen haben.

2. Punktbearbeitung:

Alle Ecken und Endpunkte erhalten einen Markierungskreis und können einzeln verschoben und gelöscht werden. Es lassen sich auch neue Ecken einfügen.

Schalten Sie zwischen diesen beiden Modi um, indem Sie im Kontextmenü *Punkte bearbeiten* aktivieren oder deaktivieren.

Die folgenden Operationen beziehen sich auf die Punktbearbeitung:

Zum Verschieben einzelner Endpunkte oder Ecken, positionieren Sie den Mauszeiger auf die Ecke oder den Endpunkt, drücken die Maustaste und ziehen die Maus an eine andere Stelle.

Zum Verschieben einzelner Segmente, markieren Sie zuerst den Linienzug, positionieren danach den Mauszeiger auf das gewünschte Segment, drücken die Strg-Taste und verschieben das Segment an eine andere Stelle. Beachten Sie, dass der Linienzug verdoppelt wird, wenn Sie die Strg-Taste erst nach der Maustaste drücken.

Zum Einfügen einer neuen Ecke positionieren Sie den Mauszeiger bei markiertem Linienzug auf das Segment, in der die neue Ecke entstehen soll und drücken die Umschalt-Taste. An der Stelle, wo die neue Ecke entstehen soll drücken Sie die Maustaste und verschieben sie an die gewünschte Position.

Ecken löschen Sie, indem Sie die Umschalttaste drücken und mit der Maus auf die Ecke klicken.

Schief verlaufende Segmente können Sie durch je ein waagrechtes und senkrecht Segment mit einer zusätzlichen Ecke ersetzen, indem Sie die Umschalttaste drücken und mit der rechten Maustaste auf das Segment klicken.

5.2.6 Polygone

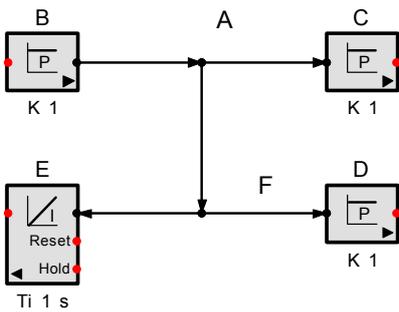
Für Polygone gelten die gleichen Zeichnungsregeln wie für Linienzüge. Das Segment zum Schliessen der Form wird aber automatisch erzeugt. Das Umwandeln von Polygonen in Linienzüge und zurück können Sie über das Kontextmenu erreichen.

5.2.7 Signalleitungen

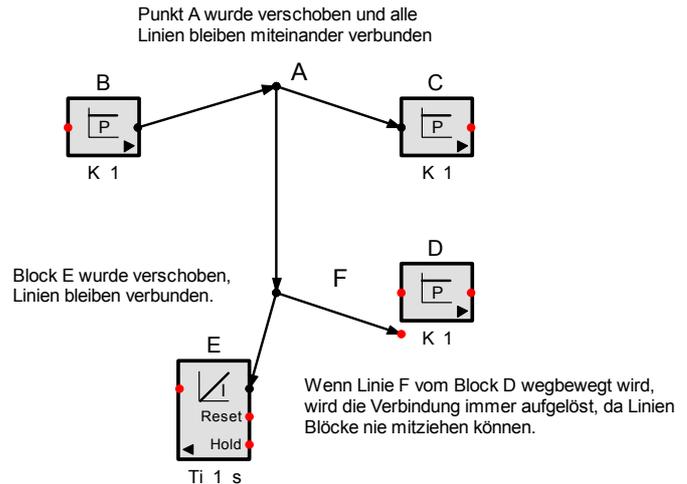
Mittels Signalleitungen werden die Daten von Block zu Block geführt. Sie werden ähnlich wie Linienzüge erstellt, mit dem Unterschied, dass die Endknoten an anderen Knoten haften bleiben. Blöcke ziehen Signalleitungen immer mit, aber Signalleitungen ziehen die Blöcke nie mit. Wenn eine Signalleitung von einem Block entfernt wird, wird die Verbindung sofort aufgelöst. Standardmässig ziehen Signalleitung auch andere Signalleitungen mit. Dieses Verhalten kann jedoch in den Programmooptionen ausgeschaltet werden. Wenn sie eine Signalleitung verschieben, und die anderen Signalleitungen nicht mitziehen möchten, drücken während der Aktion die Shifttaste. Wenn Sie die Option so gewählt haben, dass sich Signalleitung gegenseitig trennen hat die Shifttaste die gegenteilige Auswirkung. Die Verbindungen zwischen der Signalleitungen werden somit nicht aufgetrennt.

Diesen Sachverhalt erklärt die folgende Grafik etwas genauer.

Beispielschema



Standardverhalten



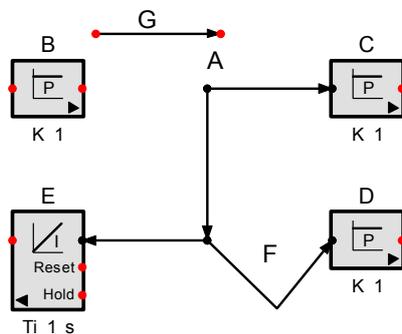
Shift-Taste

Bei einer selektierten Signallinie erscheinen kleine Kreise an Ecken und Enden. Wenn die Maus über die Liniensegmente bewegt wird, erscheinen folgende Icons:

-  Innerhalb eines Segments eine neue Ecke erstellen.
-  Ein vollständiges Segment verschieben

Mit der Shift-Taste können Sie zwischen den beiden Icons wechseln.
 Wenn das untere Icon erscheint können Sie das Segment verschieben, und die Enden der anderen Linien werden automatisch mitgezogen.
 Wenn Sie das Ende einer Linie anklicken und die Shift-Taste drücken, werden die anderen Linien nicht mitgezogen.
 Das erwartete Resultat wird mittels gestrichelten "Gummibändern" angedeutet. Wenn sie erscheinen, werden die anderen Linien mitgezogen, sonst werden die Verbindungen aufgelöst.

Sie können das Standardverhalten beim Verschieben von Linien umschalten, so dass sich die Signallinien beim Verschieben voneinander trennen. Die Shift-Taste hat dann die gegenteilige Auswirkung. Extra > Optionen > Signallinien> Verknüpfung auftrennen, wenn eine einzelne Signallinie verschoben wird.



Die beiden Enden von G wurden mit der Shift-Taste verschoben.

Linie F wurde selektiert und die Shift-Taste gedrückt. Beim Ziehen entsteht nun eine neue Ecke.

5.2.8 Text

Mit dem Textwerkzeug können Sie Text mit Zeilenumbrüchen erzeugen. Klicken Sie mit der Maus auf die gewünschte Position und geben Sie den Text mit der Tastatur ein. Beenden Sie die Eingabe durch Drücken der Eingabetaste. Einen Zeilenumbruch erreichen Sie mit Strg + Eingabetaste (Ctrl+ENTER). Drücken Sie die Escape-Taste, um einen noch nicht abgeschlossenen Text zu verwerfen. Einen bestehenden Text verändern Sie, indem Sie ihn markieren und dann das Textwerkzeug in der Palette anklicken oder den Text einfach doppelklicken.

Beachte: Für die Titelzeile von Funktionsgliedern genügt ein Einfachklick.

5.2.9 Bilder

Mit Bildern lassen sich Bitmaps (*.bmp), Metadatei-Objekte (*.wmf; .emf) oder Icons (.ico) in die Zeichnung einfügen. Diese Objekte können von einer Datei oder von der Windows-Zwischenablage stammen. Mit Bildern können Sie in SimApp Grafiken realisieren, die Sie mit dem vorhandenen Zeichnungsinstrumentarium nicht erzeugen könnten. Entwerfen Sie die Grafik in einem leistungsfähigen Zeichnungs- oder Malprogramm und fügen Sie diese als Bild in Ihre SimApp-Zeichnung ein. Die Bilder lassen sich beliebig vergrößern und verkleinern (ausser Icons) aber ihr Inhalt kann nicht mehr verändert werden. Metadateien lassen sich auch in SimApp-Objekte umwandeln.

5.2.9.1 Bilder über die Zwischenablage einfügen

Erstellen Sie eine Zeichnung in einem Mal oder Zeichnungsprogramm und kopieren Sie einen Ausschnitt davon in die Windows-Zwischenablage. In SimApp fügen Sie danach das Bild ein (Menü *Bearbeiten + Einfügen* oder mit Einfügeschaltfläche )

Wenn Sie Daten von einem Programm in die Zwischenablage kopieren, werden diese im Allgemeinen in mehreren Formaten zwischengespeichert. Der Einfügebefehl des Zielprogramms wählt jeweils dasjenige Format aus, das für die Applikation am vorteilhaftesten ist. Vorrang vor allen anderen Formaten hat immer das programmeigene, proprietäre Format, danach folgt das am nächsten verwandte Format. Ein Zeichnungsprogramm wird also immer Metadateien vor Bitmaps und ein Malprogramm immer Bitmaps vor Metadateien auswählen. Da SimApp die Eigenschaften eines Zeichnungsprogramms hat, bevorzugt es Metadateien.

Wenn Sie nicht das bevorzugte Format einfügen möchten, wählen Sie nicht den Befehl *Einfügen* sondern *Inhalte einfügen...* (besitzt keine eigene Schaltfläche). Sie erhalten darauf eine Auswahl von Formaten präsentiert, die das Programm übernehmen kann. Der Begriff Bild steht dabei für Metadateien. Nach dem Einfügen werden diese bei Massstab 1:1 (100%) im Allgemeinen in der Originalgröße dargestellt.

5.2.9.2 Bilder aus Dateien erzeugen

Klicken Sie in der Palette auf das Bildsymbol  und danach in die Zeichnung. Es erscheint ein leeres Bild mit gestricheltem Rahmen.

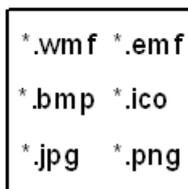


Abbildung 13: Leeres Bild

Im Kontextmenü des Bildes haben Sie nun zwei Möglichkeiten:

1. *Bild einfügen*: Fügt den Inhalt der Zwischenablage in das Bild ein. Dies entspricht dem direkten Einfügen aus der Zwischenablage mit dem *Einfügen*-Befehl wie oben beschrieben.
2. *Bild laden...*: Im nachfolgenden Dateidialog können Sie eine wmf-, emf-, bmp- oder ico-Datei in das Bildobjekt laden. Das Bild wird in Originalgröße dargestellt (bei Massstab 1:1).

5.2.9.3 Bilder bearbeiten

Den Inhalt des Bildes können Sie nicht bearbeiten. Sie können aber das Bild beliebig vergrößern und verkleinern. Um wieder zur Originalgröße zurückzukehren, wählen Sie im Kontextmenü *Originalgröße wieder herstellen*.

Sie können den Inhalt des Bildes auswechseln, indem Sie über das Kontextmenü ein neues Bild von der Zwischenablage einfügen oder von einer Datei laden.

5.2.9.4 Bilder umwandeln

Sie können Metadateien (wmf,emf) in SimApp-Zeichnungsobjekte umwandeln. Wählen Sie im Kontextmenü *Bild umwandeln*. Beim Umwandeln werden Zeichnungsobjekte, die von SimApp nicht unterstützt werden, weggelassen.

5.2.10 Pfeil

Der Pfeil ist ein Sonderobjekt und wird nur bei Anwenderblöcken benötigt. Er verdeutlicht die Richtung des Signalflusses. Andere Pfeile beliebiger Gestalt und Größe können Sie mit dem Polygonwerkzeug erzeugen.

5.3 Objekte formatieren

Jedes neu eingefügte Objekt erhält eine Formatierung. Zur Formatierung gehören z. B. die Attribute Füllung, Linienart und Textart. Neue Objekte erhalten eine Standardformatierung, die Sie jedoch jederzeit verändern können. Dazu stehen verschiedene Möglichkeiten der Eingabe zur Verfügung. Ein Objekt, das Sie formatieren möchten, müssen Sie immer zuerst markieren. Wenn Sie kein Objekt markiert haben, wird die Standardformatierung geändert. Die Formateigenschaften eines Objektes gelten nur für dieses Objekt, wobei die Standardformatierung für alle neu erzeugten Objekte gilt.

5.3.1 Formatierungssymboleiste

Für die Formatierung existiert eine eigene Symboleiste.



Abbildung 14: Formatierungssymboleiste

Damit können Sie die wichtigsten Formatierungen für Text, Füllung und Linien vornehmen. Wenn ein Objekt markiert ist, ändern Sie damit das Aussehen dieses Objektes, ansonsten wird die Standardformatierung verändert. Die Änderungen werden immer sofort ausgeführt.

5.3.2 Formateigenschaften

Alle Formatierungsattribute eines Objektes sind über das Dialogfeld *Formateigenschaften* zugänglich. Sie können dieses Dialogfeld über das Menü *Format* oder über das Kontextmenu des Objektes aufrufen.

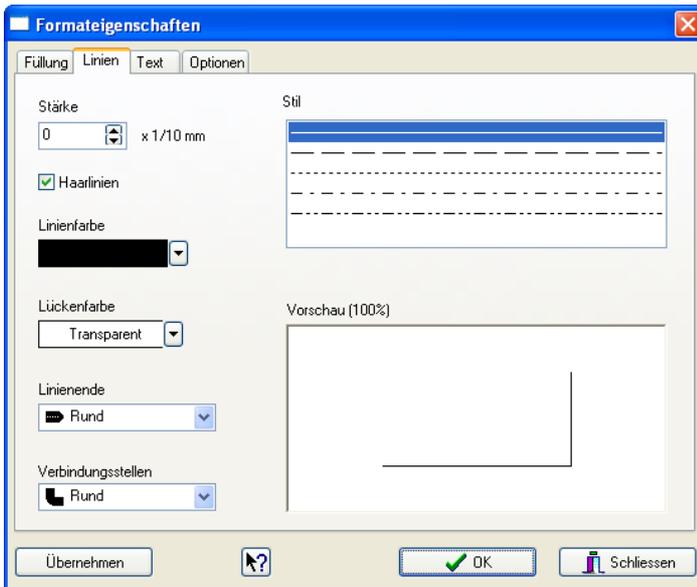


Abbildung 15: Formateigenschaften für Linien

Das Dialogfeld besteht aus vier Ebenen. Es ist kein modaler Dialog, d.h. nach erfolgter Eingabe müssen Sie das Dialogfeld nicht schliessen. Es liegt immer über der Zeichnung und zeigt die aktuellen Formatierungen an. Wenn ein Objekt markiert ist, zeigt es die aktuellen Attribute dieses Objekts an. Wenn kein Objekt markiert ist, zeigt es die Standardformatierung an. Wenn Sie die Formatierung geändert haben, werden die Änderungen, im Gegensatz zur Symboleiste erst ausgeführt, wenn Sie *OK* oder *Übernehmen* drücken.

5.3.2.1 Optionen

Auf der Ebene Optionen können erweiterte Eigenschaften für das markiert Objekt gesperrt oder zugelassen werden. Diese Optionen sind v.a. wichtig für Objekte, die Teil eines Symbols eines Anwenderblockes sind (siehe Kapitel: *Anwenderblöcke*). Sie werden deshalb im Detail dort besprochen.

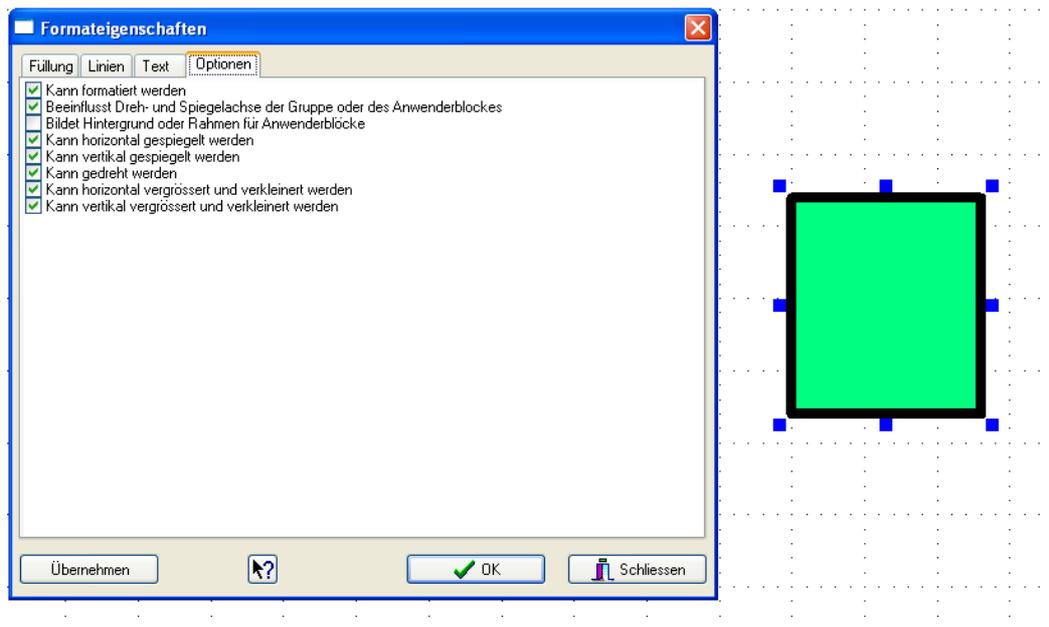


Abbildung 16: Optionen für ein Rechteck

Die Schaltflächen können drei Zustände annehmen. Wenn das Häkchen grau ist, wird die entsprechende Option beim Übernehmen nicht verändert. Bei Gruppen bedeuten graue Häkchen zusätzlich, dass nicht alle Objekte der Gruppe identische Optionszustände aufweisen. Wenn Sie ein graues Häkchen löschen oder auf schwarz stellen, erhalten alle Objekte der Gruppe den gleichen Zustand. Benutzen Sie die ?-Taste, wenn Sie genauere Informationen zu den einzelnen Optionen benötigen.

5.3.3 Standardformatierung

Wenn ein Objekt neu erstellt wird, wählt SimApp die Formatierungsattribute gemäss der aktuellen Standardeinstellung. Um diese Einstellung zu ändern, öffnen Sie das Dialogfeld Formateigenschaften und achten Sie darauf, dass nichts markiert ist. Alle Änderungen, die Sie nun vornehmen, wirken sich auf die Standardeinstellung der Anwendung aus. Die neue Standardeinstellung behält ihre Gültigkeit, bis SimApp beendet wird.

5.4 Objekte ändern und anordnen

Es stehen Ihnen mehrere Befehle zum Ändern und Anordnen von Objekten zur Verfügung. Diese sind über das Hauptmenü, das Objekt-Kontextmenü oder über eine Symbolleiste verfügbar.

Die Symbolleiste *Zeichnen* stellt die wichtigsten Befehle zur Verfügung:



Abbildung 17: Symbolleiste zum Ändern und Anordnen von Objekten

5.4.1 Objekte drehen und spiegeln

Markierte Objekte lassen sich in 90 Grad Schritten nach links  oder rechts  drehen. Die Spiegelung ist horizontal  und vertikal  möglich. Die Spiegelachse verläuft immer durch das Zentrum des Objektes, so dass das gespiegelte Objekt wieder den gleichen Platz einnimmt.

5.4.2 Objekte ordnen

Die innere Ordnung einer Zeichnung ist nur sichtbar, wenn sich die Objekte überlappen. In solchen Fällen entscheidet die Reihenfolge der Speicherung darüber, welche Objekte zuerst oder zuletzt gezeichnet oder gedruckt werden. Die Reihenfolge der Objekte ist standardmässig identisch mit der Reihenfolge, in der sie erstellt worden sind. Neuere Objekte liegen über den Älteren. Sie können diese Reihenfolge jederzeit ändern. Wenn Sie ein Objekt in den Vordergrund  verschieben, kommt es über alle anderen Objekte zu liegen. Neue Objekte werden aber wiederum darüber gelegt. Wenn Sie es in den Hintergrund  verschieben, liegt es danach unter allen anderen Objekten.

5.4.3 Objekte auf Raster verschieben

Wenn Sie Objekte freihändig zeichnen, ist es schwierig, sie auf einer Linie zu halten. Für solche Zwecke verfügt SimApp über ein Raster, das mit Punkten in der Zeichnung angedeutet wird. Sie können auswählen, ob die Objekte beim Zeichnen automatisch auf das Raster ausgerichtet werden. Diese Eigenschaft ist besonders für das Zeichnen von Blockdiagrammen wichtig. Das Verknüpfen von Objekten mit Signalleitungen wird dadurch wesentlich erleichtert, ansonsten haben Sie Mühe die Knoten genau zu treffen. Das Raster ist so eingestellt, dass die Knoten immer genau darauf liegen.

Sie können das Einrasten vorübergehend ausschalten (mit Schaltfläche ) , wenn das Raster für einen bestimmten Zeichnungsvorgang zu grob ist. Wenn Sie aber nachher vergessen, es wieder einzuschalten, platzieren Sie alle nachfolgenden Objekte in der Regel neben das Raster. Wenn Sie nun das Einrasten wieder einschalten, werden Sie feststellen, dass es Ihnen nicht gelingt, diejenigen Objekte, die Sie ohne Einrasten platziert haben, mit den übrigen Objekten in Übereinstimmung zu bringen, da die beiden Objektgruppen offenbar ein nicht deckungsgleiches Raster aufweisen.

SimApp unterscheidet zwischen einem globalen und einem objektbezogenen Raster. Das globale Raster entspricht den Rasterpunkten in der Zeichnung. Das objektbezogene Raster bezieht seinen Ursprung jedoch immer auf den Standort des Objekts. Die Distanz zwischen den Rasterpunkten ist jedoch identisch. Wenn Sie also ein Objekt ohne Rasterung verschieben, ist sein Raster nicht mehr deckungsgleich mit dem globalen Raster.

Klicken Sie auf die Schaltfläche  um ein Objekt oder eine Auswahl wieder auf das globale Raster auszurichten oder benutzen Sie den Befehl *Auf Raster* im Kontextmenü.

Tip: Schalten Sie das Einrasten nicht aus, wenn Sie ein Objekt genauer platzieren möchten. Drücken Sie während der Zeichnungsaktion stattdessen die Alt-Taste. Solange Sie die Alt-Taste gedrückt halten, ist das Einrasten ausgeschaltet.

5.4.4 Objekte gruppieren

Objekte lassen sich gruppieren, so dass das gleichzeitige Bearbeiten mehrerer Objekte erleichtert wird. Markieren Sie hierfür alle gewünschten Objekte und wählen den Befehl *Gruppieren* im Menü *Extras* oder die Schaltfläche . Die Gruppierung der Objekte kann jederzeit mit *Extras + Gruppierung aufheben* oder der Schaltfläche  wieder rückgängig gemacht werden.

5.5 Wichtige Hilfstasten

Beim Arbeiten mit SimApp benötigen Sie vor allem die Maus. Bei einigen Aktionen brauchen Sie jedoch zusätzlich eine oder zwei Hilfstasten.

Alt-Taste

Die Alt-Taste hat verschiedene Funktionen:

- In der Zeichnung: Während einer Zeichnungsaktion das Raster temporär ausschalten.
- Palette und Bibliotheken im Bearbeitungsmodus: Drücken Sie die Alt-Taste vor der Maustaste, wenn Sie Palettenseiten oder einzelne Schaltflächen darin verschieben möchten.

Strg-Taste (Ctrl)

Die Strg-Taste hat verschiedene Funktionen:

- Beim Verschieben von Objekten: Wenn Sie ein Objekt verschieben und kopieren möchten, drücken Sie vor dem Loslassen der Maustaste die Strg-Taste. Dies gilt auch zum Kopieren von Objekten in die Palette oder in Bibliotheken.
- Beim Einfügen von Objekten aus der Palette: Kehrt die Standardeinstellung in den Programmoptionen "*Objekte nach dem Einsetzen auswählen*" um.

Umschalt-Taste (Shift)

Die Umschalttaste bewirkt das Ausrichten von Objekten. Sie benötigen diese Taste, wenn Sie exakte Quadrate und Kreise zeichnen möchten. Bei Linienobjekten bewirkt sie das senkrechte, waagrechte oder 45 Grad Zeichnen oder schaltet es vorübergehend aus, wenn es fest eingeschaltet ist.

Die Umschalttaste bewirkt auch, dass Signallinien voneinander getrennt werden, wenn Funktionsglieder, mehrere Objekte auf einmal oder einzelne Knoten von Signallinien verschoben werden. In den Programmoptionen können Sie bei Bedarf auch einstellen, dass sich Signallinien gegenseitig immer auftrennen. In diesem Fall hat die Umschalttaste die gegenteilige Auswirkung.

Esc-Taste

Die Escape-Taste ist die Abbruchtaste. Linienzüge, Signalleitungen und Texte können Sie damit wieder verwerfen, bevor Sie die Zeichnungsaktion abgeschlossen haben.

Leer-Taste

Die Leertaste benutzen Sie, um das Autoscrolling zu unterbinden, z.B. wenn Sie ein Objekt aus der Zeichnung in die Palette (siehe Kapitel [Palette bearbeiten](#)) verschieben möchten.

6 Simulationsobjekte

6.1 Beschreibung

Die Simulationsobjekte sind die Grundbausteine für die Blockdiagramme. Sie bestehen aus den Standard-Funktionsgliedern, den Signalleitungen sowie den Anwenderblöcken. Der Pfeil in der unteren rechten Ecke der Glieder zeigt die Richtung des Signalfusses durch das Objekt an. Die Ausgänge sind immer auf derjenigen Seite, auf die der Pfeil hinzeigt. Die Signalrichtung ist grundsätzlich immer von links nach rechts, also die Eingänge links und die Ausgänge rechts. Für rückgekoppelte Systeme können Sie die Objekte mit der Schaltfläche  drehen.

Sie können auch eigene Objekte, Anwenderblöcke, entwerfen, die aus einem verknüpften System von Standardobjekten oder weiteren Anwenderblöcken bestehen (siehe Kapitel [Anwenderblöcke](#)).

Die linearen Funktionsglieder bieten in der Regel zwei unterschiedliche Symbole zur Auswahl: Das Standardsymbol zeigt die schematische Darstellung der Schrittantwort oder des Frequenzganges und das Zweite die mathematische Darstellung der Übertragungsfunktion im Zeit oder Frequenzbereich.

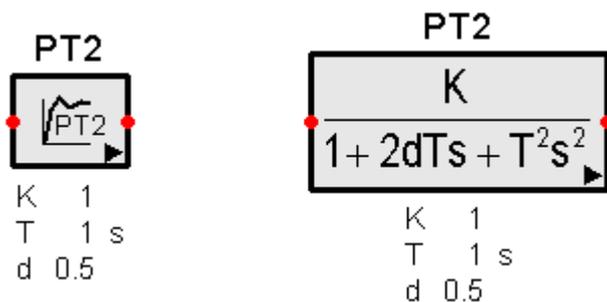


Abbildung 18: Das PT₂-Glied und seine beiden Symbole

Das Standardsymbol ist in den Programmoptionen wählbar. Über das Kontextmenu des Gliedes kann jedoch jederzeit auf das alternative Symbol umgeschaltete werden.

Das Übertragungsverhalten eines Gliedes wird durch seine Parameter und anderen Optionen beeinflusst. Die wichtigsten Parameter eines Gliedes werden in der Zeichnung unterhalb des Symbols angezeigt und können dort direkt verändert werden (Wert anklicken). Andere Parameter sind nur über das Dialogfeld *Simulationseigenschaften* (im Kontextmenü) erreichbar. Parameter, die normalerweise in der Zeichnung nicht sichtbar sind, werden dennoch angezeigt, wenn ihr aktueller Wert vom Standardwert abweicht. Ob ein Parameter immer in der Zeichnung angezeigt werden soll oder nicht, kann individuell in den Simulationseigenschaften der Glieder festgelegt werden.

6.2 Funktionsglieder verbinden

Um den Signalfuss zu gewährleisten, werden die Knoten der Funktionsglieder mit Signalleitungen verbunden.

Signalleitungen verhalten sich bezüglich ihrer Bearbeitung ähnlich wie Linienzüge (siehe Abschnitt [Linienzüge](#)). Die Abweichungen werden im Folgenden erklärt:

Das Werkzeug zum Zeichnen von Signallinien finden Sie im dauernd sichtbaren Bereich der Palette. Es wird jedoch auch automatisch aktiviert, wenn Sie die Maus über einem Knoten eines nicht markierten Funktionsgliedes platzieren.

Eine neue Signalleitung starten Sie durch Drücken der Maustaste. Ecken werden erzeugt, indem Sie die Maus kurz loslassen. Eine Signalleitung wird abgeschlossen, indem Sie mit der Maus doppelklicken oder die Maus über dem Zielknoten loslassen. Achtung: Durch unsaubere Mausführung können sehr kurze Segmente oder unnötige Ecken entstehen.

Die Signalleitungen haben die gleichen Bearbeitungsmodi wie die Linienzüge wobei standardmässig die Punktbearbeitung eingestellt ist.

Wenn Sie Knoten verschieben, werden die verbundenen Knoten anderer Signalleitungen automatisch mitgezogen. Glieder werden im Gegensatz nie mitgezogen. Wenn beim Verschieben die Umschalttaste (Shift) gedrückt wird, werden die Verknüpfungen zu den anderen Knoten aufgetrennt.

Die Formatierung der Signalleitungen wird in den Programmoptionen (Menü *Extras + Optionen...*) eingestellt. Individuelle Formatierungen sind nicht möglich.

Da Signalleitungen nur Signale in einer Richtung transportieren können, ergeben sich beim Zeichnen einige Einschränkungen. Zwei Ausgänge können nicht miteinander verbunden werden. Die Wirkrichtung von Signalleitungen und somit den Pfeil können Sie über *Pfeil drehen* im Kontextmenü oder mit der *Ende-* und *Pos1(Home)*-Taste umkehren.

6.2.1 Additionsstelle

Mit Hilfe eines Summiergliedes lässt sich eine Additionsstelle erzeugen oder ein Signal invertieren.

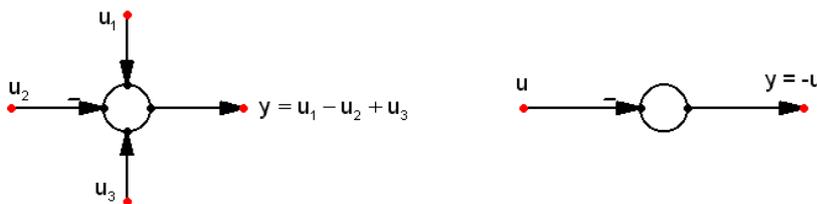


Abbildung 19: Additionsstelle und Signalinversion

Summierglieder können auch mehrere Ausgänge haben. In diesem Fall führen alle Ausgänge das gleiche Signal.

Der Wirkungssinn der Addition wird durch das Vorzeichen bei der Pfeilspitze verdeutlicht und ist standardmässig positiv. Eine Umkehrung kann via Kontextmenü oder mit der Minus- und Plus-Taste im numerischen Tastenfeld erreicht werden. Drücken Sie die Alt-Taste, wenn Sie eine Signalleitung an ein Summierglied anschliessen wollen und wegen des automatischen Einrastens Mühe haben, es zu treffen. Das Einrasten wird damit vorübergehend ausgeschaltet. Zum Abschliessen der Leitung genügt es, den Endknoten irgendwo innerhalb des Kreises zu positionieren.

6.2.2 Verzweigungsstelle

Verzweigungsstellen benötigen Sie, wenn Sie ein Signal gleichzeitig an mehrere Orte führen müssen.

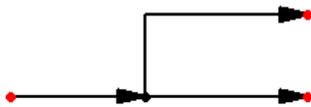


Abbildung 20: Verzweigungsstelle

Wenn Sie den Anfangsknoten einer Signalleitung auf ein Segment oder auf eine Ecke einer anderen Signalleitung platzieren, wird automatisch eine Verzweigungsstelle erzeugt. Um eine Signalleitung von einer Verzweigungsstelle wieder zu trennen, müssen Sie beim Verschieben die Umschalttaste (Shift) drücken. Dasselbe gilt, wenn Sie ein Funktionsglied verschieben und die angeschlossenen Signalleitungen nicht mitziehen möchten.

Bemerkung: Das gegenseitige Trennen von Signalleitungen und die Verwendung der Shift-Taste können Sie in den Programmeinstellungen bei Bedarf dauerhaft umkehren. Standardmässig werden Signalleitungen von anderen Signalleitungen mitgezogen. Sie können jedoch bestimmen, dass dies nicht der Fall ist und ein Mitziehen nur erfolgt, wenn die Shift-Taste gedrückt wird.

6.3 Direkteinstellungen

Direkteinstellungen sind Veränderungen von Attributen oder Parametern direkt in der Zeichnung, also ohne Umweg über ein Dialogfeld. Für Funktionsglieder trifft dies auf den Titel und die angezeigten Parameter zu.

6.3.1 Titel verschieben und editieren

Um den Titel eines Funktionsgliedes zu verändern klicken Sie einfach darauf so dass er als Ganzes markiert wird. Durch einen zweiten Klick lässt sich der Cursor im Text an einer bestimmten Stelle platzieren. Der Editiervorgang wird mit der Eingabetaste abgeschlossen. Zeilenumbrüche werden mit Strg + Eingabetaste eingefügt.

Um den Titel zu verschieben, klicken Sie ihn mit gedrückter Alt-Taste an.

Hinweis: Bei Textänderungen wird der Titel automatisch neu positioniert. Wenn Sie ihn dauerhaft an einer anderen Stelle platzieren möchten, müssen Sie die entsprechende Option in den Simulationseigenschaften aktivieren.

6.3.2 Parameter ändern

Klicken Sie einfach auf den Zahlenwert des Parameters. Wenn Sie die Alt-Taste gedrückt halten, wird die ganze Parametertabelle markiert und kann verschoben werden.

Bei Parameteränderungen wird die Parametertabelle automatisch neu positioniert wenn nicht die entsprechende Option in den Simulationseigenschaften des Gliedes deaktiviert wird.

6.4 Simulationseigenschaften

Dieses Dialogfeld dient für alle objektspezifischen Einstellungen, die mit der Simulation zusammenhängen. Sie öffnen es durch Doppelklick auf das Funktionsglied oder mit dem Befehl *Simulationseigenschaften* in dessen Kontextmenü.

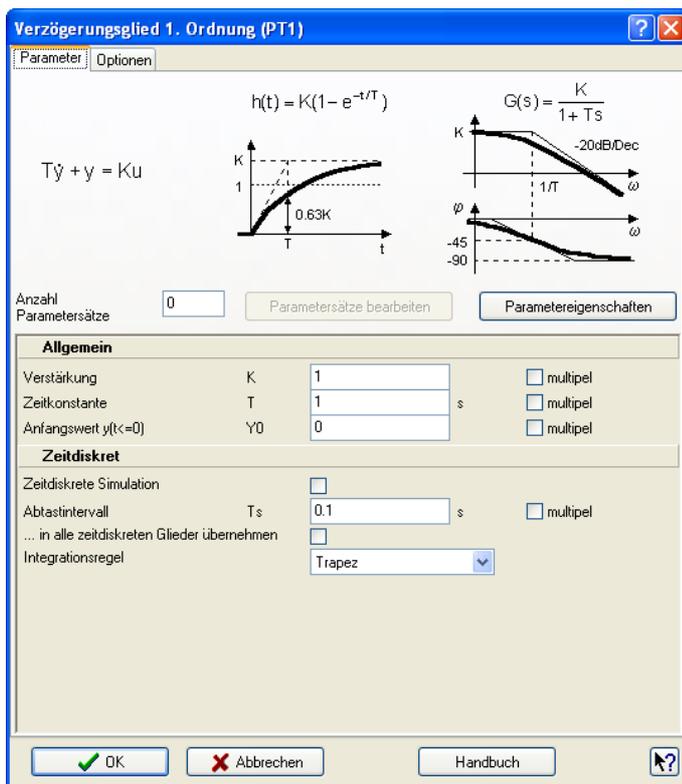


Abbildung 21: Simulationseigenschaften des PT₁-Gliedes

Das Dialogfeld enthält normalerweise zwei Ebenen. Auf der Ebene *Daten* werden die Parameterwerte und andere simulationsspezifische Einstellungen angezeigt. Auf der Seite *Optionen* wird das Aussehen der Objekte beeinflusst. Einige Glieder enthalten noch weitere Ebenen mit zusätzlichen Einstellmöglichkeiten.

Anzahl Parametersätze, *Parametersätze bearbeiten* sowie die Kontrollkästchen *multipel* beziehen sich auf die Parametervariation (siehe Kapitel [Parametervariation](#)).

6.4.1 Parametereigenschaften

Im Hauptdialog können Sie bloss die Zahlenwerte der Parameter verändern. Es ist jedoch auch möglich, weitere Einstellungen vorzunehmen. Drücken Sie dazu die Taste *Parametereigenschaften*.

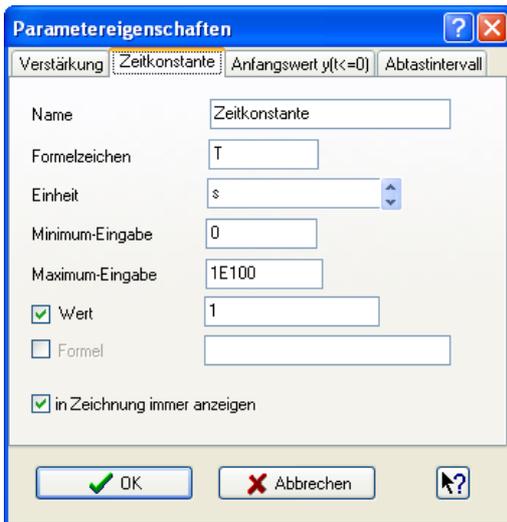


Abbildung 22: Parametereigenschaften

Wenn Sie die Minimum- und Maximumeingabe je auf null setzen, findet keine Bereichsprüfung statt. Der Zustand der Schaltfläche *In Zeichnung immer anzeigen* bestimmt, ob der aktuelle Parameter in der Zeichnung dauernd angezeigt werden soll, also nicht nur, wenn sein Wert vom Standardwert abweicht. Die Eingabe *Formel* dient nur für diejenigen Glieder, die in der Blockmappe in einen Anwenderblock integriert werden (siehe Kapitel [Anwenderblöcke](#)). Ist das Glied nicht in einem Anwenderblock, wird die Formel nicht beachtet, auch wenn das Schaltkästchen aktiviert ist.

6.4.2 Einheiten

SimApp kennt folgende Einheiten:

Zeit	ps	ns	μs	ms	s	min	h	d	a			
Steigung	ps ⁻¹	ns ⁻¹	μs ⁻¹	ms ⁻¹	s⁻¹	min ⁻¹	h ⁻¹	d ⁻¹	a ⁻¹			
Frequenz	μrad/s	μHz	mrاد/s	mHz	rad/s	Hz	krad/s	kHz	Mrad/s	MHz	Grad/s	GHz

Die Grundeinheiten sind **fett** markiert.

SimApp erkennt diese Einheiten und benutzt für den Parameterwert intern die korrekten Umrechnungsfaktoren.

Wenn im Eingabefeld für die Einheit eine Standardeinheit steht, erscheint rechts eine Auf-/Abschaltfläche, die ein schnelles Umschalten innerhalb des gleichen Typs ermöglicht.

Wenn SimApp eine Einheit nicht erkennt, wird der Parameterwert unverändert benutzt. Wenn Sie also anstelle **a** den Buchstaben **J** für die Einheit Jahr benutzen rechnet SimApp intern mit der Einheit **s** (Sekunden).

Achtung: Bei Einheiten müssen Sie unbedingt die Gross-/Kleinschreibung beachten. Wenn Sie z.B. für die magnetische Induktivität **h** anstelle von **H** (Henry) eingeben, wird der Parameterwert in Stunden interpretiert und somit mit dem Faktor 3600 (Sekunden) multipliziert. Da H eine unbekannte Einheit ist, würde bei korrekter Eingabe keine Multiplikation stattfinden.

Mikro μ wird durch u repräsentiert und entspricht 1E-6.

6.4.3 Optionen

Auf der Ebene *Optionen* können Sie den Titel des Gliedes ändern und bestimmen, welche Teile der Parameterspezifikation direkt in der Zeichnung angezeigt werden. Standardmässig werden Formelzeichen, Wert und Einheit (falls vorhanden) angezeigt. Wenn kein angezeigter Parameter eines Gliedes eine Einheit aufweist, sollten Sie diese ausblenden, da sonst die Parametertabelle nicht korrekt zentriert wird.

6.4.4 Glieder und Signale bezeichnen

Es ist wichtig, dass Sie Ihre Blockdiagramme sorgfältig strukturieren und beschriften. Die Namen der Glieder und Signale werden im Diagrammfenster zur Bezeichnung der Diagrammkurven und der Spalten in den Datentabellen benutzt.

Jedes Glied besitzt einen Namen, der oberhalb des Symbols als Titel angezeigt wird. Der Standardname entspricht in der Regel dem Typ des Übertragungsgliedes. Geben Sie dem Glied möglichst einen treffenden und eindeutigen Namen, der dessen Funktion im System möglichst gut bezeichnet. Ein Wassertank kann z.B. durch ein Integrationsglied modelliert werden. Geben Sie also dem Glied den Namen „Tank“.

Die Signalleitungen verwenden den Namen desjenigen Gliedes aus dem sie die Daten entnehmen. Wenn ein Glied mehrere Ausgänge aufweist, haben jedoch alle Ausgangssignale den gleichen Namen. Um individuelle Signalnamen festzulegen, die sich vom Objektamen unterscheiden, benutzen Sie das Kontextmenü der Ausgangsknoten der Glieder. Klicken Sie mit der rechten Maustaste auf einen Ausgangsknoten (das Glied darf selber aber nicht markiert sein!) und geben Sie einen Namen ein. Den Namen müssen Sie danach noch exakt positionieren. Benutzen Sie zum Positionieren die Alt-Taste, damit das Einrasten vorübergehend ausgeschaltet wird.

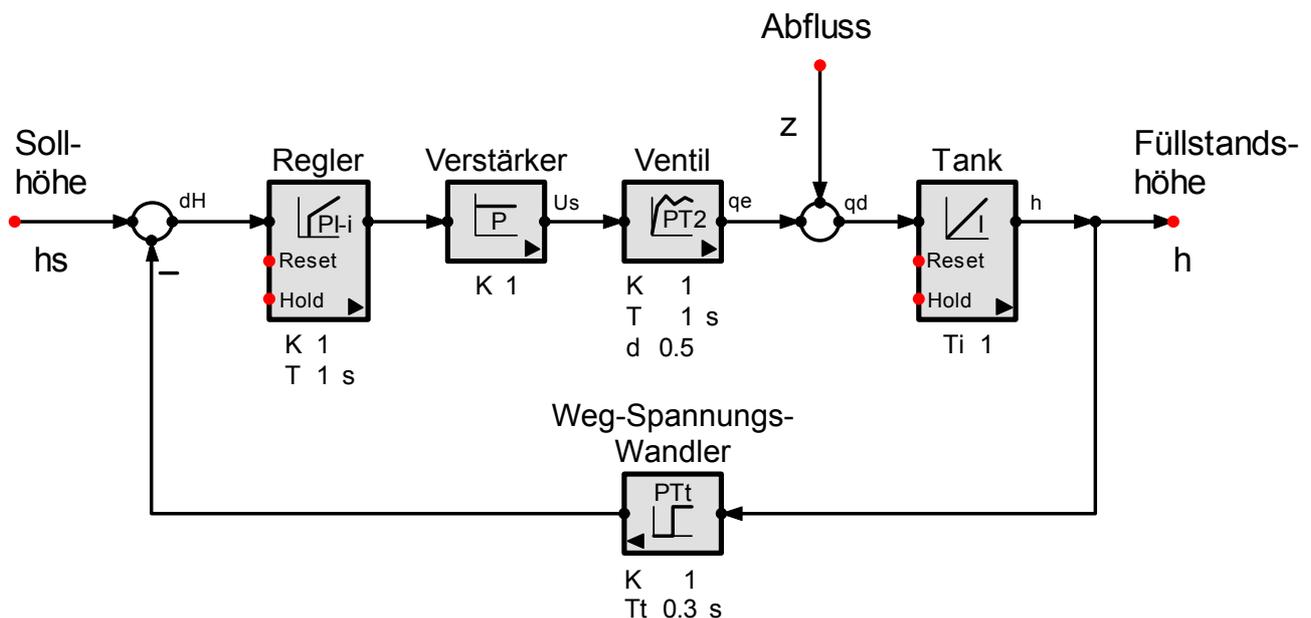


Abbildung 23: Glieder und Signale korrekt beschriften

Den Namen eines Knotens eines Funktionsgliedes lässt sich durch Rechtsklick auf den Knoten verändern. Er wird automatisch innerhalb des Blockrahmens neben den Knoten platziert. Sie können ihn verschieben, wenn Sie ihn anklicken und dabei die ALT-Taste gedrückt halten.

Die Knoten von Signalleitungen können nicht bezeichnet werden. Die Signalleitung trägt automatisch den Namen des Gliedes das die Leitung speist oder des Knotens, falls dieser bezeichnet ist.

Der Ausgangsknoten des Summiergliedes kann nicht bezeichnet werden, weil dieser zur Signalleitung gehört. Benutzen Sie stattdessen den Objektamen des Summiergliedes.

7 Frequenzsimulation

In diesem Kapitel erhalten sie detaillierte Informationen zur Frequenzsimulation. Die Grundlagen dazu haben Sie bereits im [Einführungsbeispiel](#) vermittelt bekommen.

7.1 Blockdiagramm erstellen, Modellieren

Erstellen Sie ein Blockdiagramm setzen Sie die korrekten Werte der Parameter. Beachten Sie aber, dass Sie für die Frequenzsimulationen nur lineare Glieder verwenden können, dazu gehören auch die zeitdiskreten Glieder. Nichtlineare Systemteile müssen sie durch entsprechende lineare Glieder ersetzen.

Hinweis: Im nichtlinearen Reiter Objektpalette, existieren einige Glieder mit wählbaren Übertragungsfunktionen (Verknüpfungsglied, Funktionsglieder mit 1 oder 2 Eingängen). Wenn Sie eine lineare Funktion auswählen, können Sie auch diese Glieder für die Frequenzsimulationen benutzen.

7.2 Frequenzsonden

Wenn Sie das Blockdiagramm erstellt haben, wählen Sie die Übertragungsstrecken die Sie näher untersuchen möchten. Der einfachste Fall liegt vor, wenn das System nur einen Eingang und mindestens einen Ausgang aufweist. In diesem Fall können Sie gleich den Startknopf drücken.

Hinweis: Bei mehreren Eingängen können Sie auch die nicht gewünschten Eingangsknoten auf inaktiv setzen (Rechtsklick). Einzelne Glieder weisen besondere Eingangsknoten auf (z.B. Reset beim Integrator), die für die Frequenzsimulation nicht relevant sind. Diese Knoten werden von SimApp automatisch weggelassen und müssen nicht speziell beachtet werden.

Durch den Einsatz von Frequenzsonden lassen sich aber auch beliebige Strecken innerhalb des Systems untersuchen, selbst wenn diese keine expliziten Ein- und Ausgänge besitzen.

Frequenz

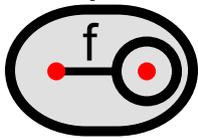


Abbildung 24: Frequenzsonde

Jede Frequenzsonde besitzt zwei Anschlüsse. Der linke Anschluss ist ein Ausgang für die Anregung des Systems mit einem sinusförmigen Signal. Der rechte Anschluss ist ein Eingang mit dem mehrere Punkte des Systems ausgemessen werden können. Die Sonde wird durch normale Signalleitungen mit den Systemknoten verbunden. Die Einspeisung kann an jedem beliebigen Punkt vorgenommen werden. SimApp deaktiviert automatisch die Ausgänge derjenigen Glieder, die ebenfalls auf diesen Punkt wirken. Die Anzahl der Frequenzsonden ist unbeschränkt. Pro Einspeisungspunkt wird eine Sonde benötigt. Alle Messsignale aller Sonden werden im gleichen Diagramm angezeigt und können miteinander verglichen werden. Die Namen der Sonden erscheinen in den Legenden der Diagramme als Überschriften.

Hinweis: Die Frequenzsimulation unterscheidet sich von der Art der Berechnung völlig von der Zeitsimulation. Die Einspeisung des Sinussignals erfolgt nur virtuell. SimApp berechnet aus den Strecken die Übertragungsfunktionen, worin die Frequenz als variable Größe auftritt. Dadurch können einfach und schnell die Diagrammkurven und die Eigenwerte berechnet werden. Verzögerungsfreie Rückkopplungen stellen kein Problem dar. Das System muss jedoch rein linear sein.

Eine Standardanwendung für Frequenzsonden ist die Untersuchung eines offenen und geschlossenen Regelkreises:

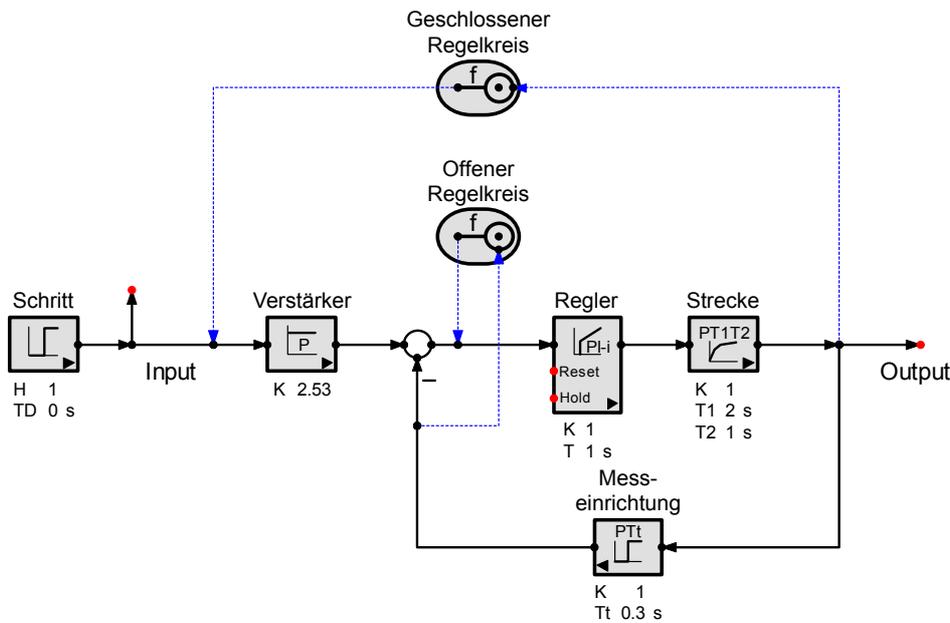


Abbildung 25: Offener und geschlossener Regelkreis mit Frequenzsonden ausmessen

Bemerkungen:

- Eingänge und Ausgänge sind alle rot gezeichneten Knoten. Wenn noch weitere Knoten rot sind, bedeutet dies, dass diese nicht verbunden sind. Dies ist meistens auf ungenaues Platzieren zurückzuführen.
- Das Blockdiagramm darf keine Kurzschlüsse und ungespeiste (signallose) Knoten aufweisen. Strukturfehler erkennen Sie im Allgemeinen daran, dass ein Knoten oder eine Signalleitung im kritischen Teil gelb gezeichnet wird.
- Zur Frequenzganguntersuchung benötigen Sie keine Signalquellen. Wenn sich dennoch eine oder mehrere Quellen in der Zeichnung befinden, werden diese durch SimApp nicht beachtet. Der Knoten, der mit der Quelle verbunden ist, wird jedoch nicht automatisch zum Eingangsknoten, falls sie keine Frequenzsonden einsetzen.
- Geben Sie jeder Sonde einen eigenen Namen, damit Sie sich in den Diagrammen und Datentabellen einfacher orientieren können.

7.3 Simulationsparameter

Bevor Sie eine Frequenzsimulation starten, sollten Sie noch die Parameter festlegen, die den Simulationsablauf steuern. Öffnen Sie dazu das dazugehörige Dialogfeld entweder unter Menü *Frequenzsimulation + Optionen...*, über die Simulationssymboleiste oder über das Kontextmenü der Zeichnung.

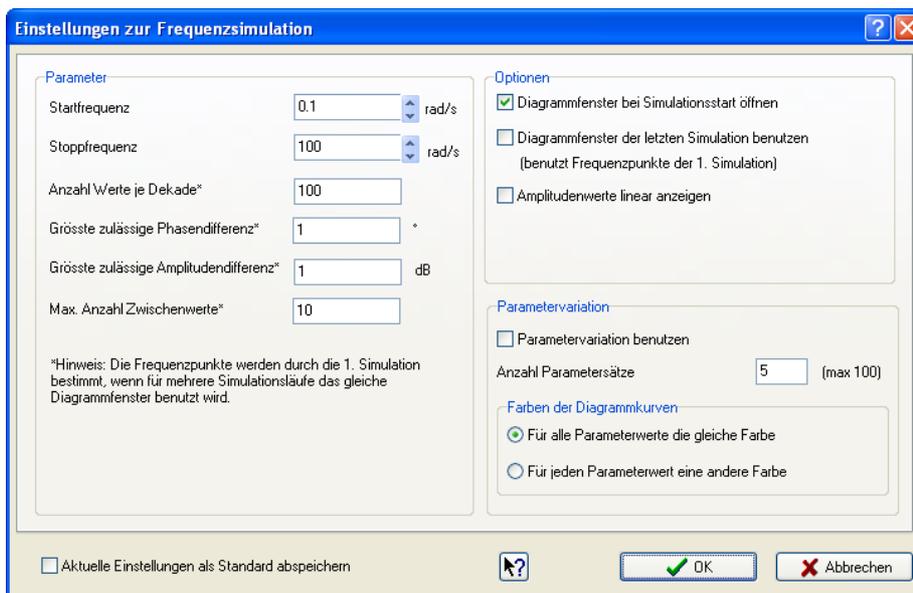


Abbildung 26: Parametereingabe für die Frequenzsimulation

Geben Sie je eine Start- und Stoppfrequenz ein und die Anzahl der Frequenzpunkte die pro Dekade berechnet werden soll. Im Allgemeinen treten bei der Phase starke Unterschiede im Übertragungsverhalten zwischen den verschiedenen Dekaden auf. Falls sie steile Kennlinien erhalten, benötigen Sie unter Umständen eine höhere Auflösung und setzen die Anzahl der Frequenzpunkte hinauf. Dies gilt dann für alle Dekaden.

Eine veränderliche Auflösung ermöglichen die folgenden drei Parameter:

Wählen Sie je eine maximale Phasen- und Amplitudendifferenz, die zwischen zwei berechneten Frequenzpunkten nicht überschritten werden dürfen und legen Sie fest, wie viele Zwischenpunkte andernfalls zusätzlich berechnet werden müssen.

Weitere Informationen zu den verschiedenen Einstellungen erhalten Sie in der Programmhilfe durch Benutzung der ?-Schaltfläche. Weitere Angaben zur Parametervariation finden Sie im Kapitel [Parametervariation](#).

7.4 Frequenzsimulation starten

Starten Sie die Frequenzsimulation durch Menü *Frequenzsimulation + Start* oder  in der Symbolleiste. Die Rechenzeit ist abhängig von der Grösse des Systems und den gewählten Einstellungen. Während einer längeren Simulation können Sie an einer anderen Zeichnung weiterarbeiten.

7.5 Resultate der Frequenzsimulation

7.5.1 Diagramme

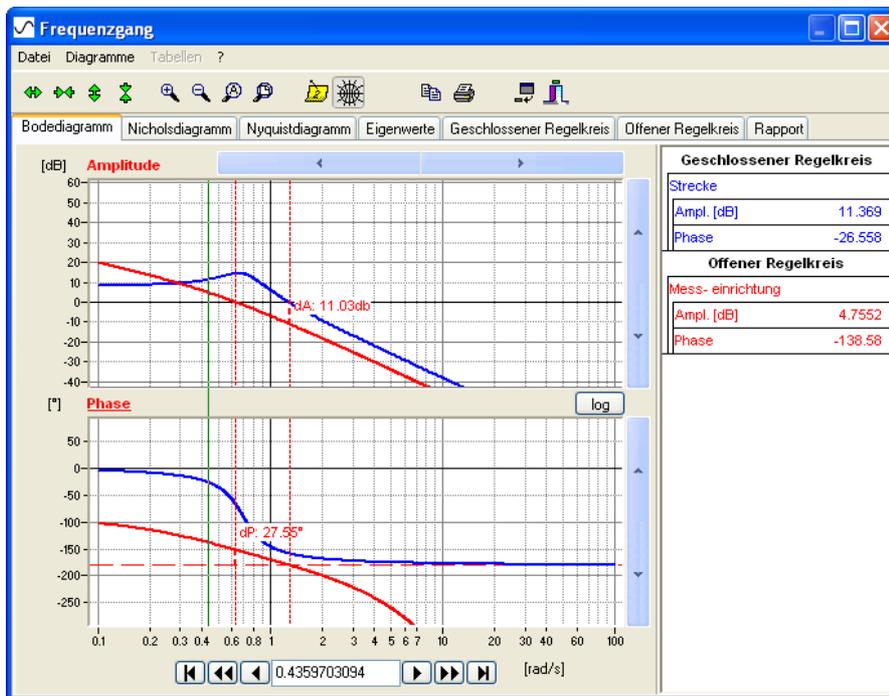


Abbildung 27: Bodediagramm

Ein Diagramm kann mehrere Kurven enthalten, die sich farblich voneinander unterscheiden. Jede Kurve besitzt ein eigenes Kontextmenü (Klick mit der rechten Maustaste auf die Kurve oder die Legende), mit dem Sie verschiedene Einstellungen vornehmen können.

In der Symbolleiste finden Sie mehrere Bearbeitungswerkzeuge, mit denen Sie den aktuellen Ausschnitt verändern und verschieben können. Beim Bodediagramm wirken diese Werkzeuge immer auf das aktive, fokussierte Diagramm, das Sie am unterstrichenen Diagrammtitel erkennen können. Durch Linksklick in das andere Diagramm lässt sich der Fokus wechseln.

Die Maus enthält eine integrierte Lupenfunktion mit der Sie rechteckige Bereiche auswählen können. Der Inhalt eines solchen Bereiches wird unmittelbar nach dem Loslassen der linken Maustaste auf Diagrammgrösse vergrößert, falls dies möglich ist. Eine Grössenanpassung erreichen Sie auch über die vier Lupenknöpfe in der Symbolleiste.

Das Bodediagramm enthält eine Messlinie und das Nyquist- und Nicholsdiagramm auf jeder Kurve einen Messpunkt die sich mit der Maus oder den Pfeiltasten bewegen lassen.

Die Zahlenwerte in der Legende zeigen die aktuellen Signalwerte am Ort der Messlinie oder der Messpunkte. Die Messlinie im Bodediagramm und die Messpunkte im Nyquist-/Nicholsdiagramm sind immer im Gleichschritt.

7.5.2 Besondere Effekte

Die Frequenzkennlinien werden durch SimApp numerisch berechnet. Folgende Effekte können dabei auftreten:

7.5.2.1 Abweichung der Startphase

Die Berechnung der Phase aus dem Real- und Imaginärteil der Frequenzgangmatrix erfolgt durch die Anwendung der Arctan-Funktion auf Real- und Imaginärteil der einzelnen Elemente [1]. Wegen der Periodizität der Arctan-Funktion von 360° kommt die Startphase immer innerhalb $\pm 180^\circ$ zu liegen. Ist die reale Startphase jedoch ausserhalb dieses Bereichs, kann dies SimApp nicht erkennen und es entsteht ein konstanter Phasenfehler über den ganzen Simulationsbereich als ein Mehrfaches von 360° . Drei Integratoren in Serie ergeben in der Simulation eine konstante Phase von 90° anstelle von -270° .

7.5.2.2 Phasensprünge

Bei sehr steilem Phasenverlauf kann die Phasendifferenz zwischen zwei Punkten grösser als $\pm 180^\circ$ werden. Wegen der Periodizität der Arctan-Funktion entstehen dann falsche Phasensprünge, die jedoch in den Diagrammen leicht zu erkennen sind. Dieser Effekt tritt wegen der starken Phasendrehung vor allem beim Einsatz von Totzeitgliedern auf.

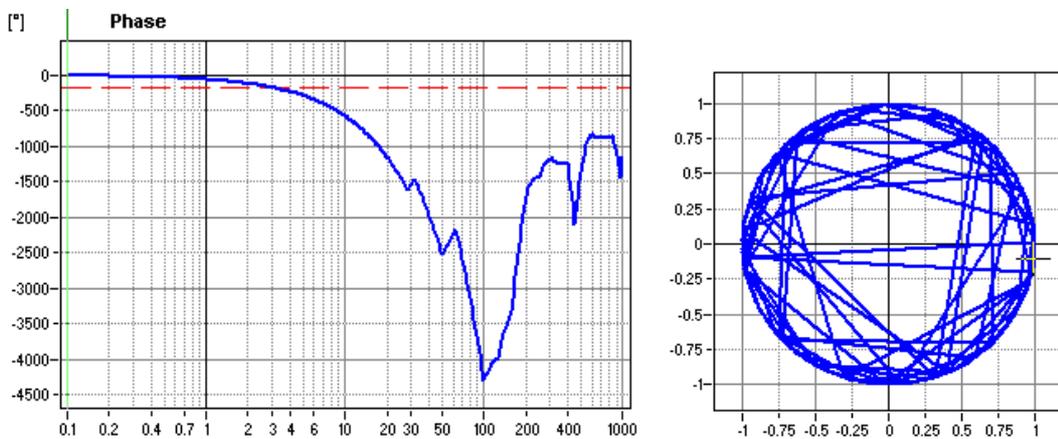


Abbildung 28: Falsche Phasensprünge mit Totzeitgliedern

In der Abbildung sind die Bereiche mit falschem Phasenverlauf durch die abrupten Phasenänderungen erkennbar. Diesen Effekt können Sie vermindern, indem Sie die Anzahl Werte je Dekade erhöhen, die zulässige Phasendifferenz vermindern und die Anzahl der Zwischenwerte vergrössern. Zu Beachten ist jedoch, dass dieser Effekt im Allgemeinen erst ausserhalb des interessierenden Frequenzbereichs auftritt. Um die störende Erscheinung im Diagramm zu vermeiden, genügt es deshalb meistens, die Stoppfrequenz zu vermindern.

7.5.3 Eigenwerte

Für jede untersuchte Übertragungsstrecke werden die Eigenwerte berechnet und in einer Tabelle angezeigt. Das Totzeitglied wurde im folgenden Beispiel durch einen Padé-Allpass 6. Ordnung angenähert.

	Geschlossener Regelkreis	Offener Regelkreis
0	-16.817+30.126j	-16.773+29.951j
1	-16.817-30.126j	-16.773-29.951j
2	-29.661+10.417j	-24.904+17.508j
3	-29.661-10.417j	-24.904-17.508j
4	-23.601+15.386j	-28.323+5.783j
5	-23.601-15.386j	-28.323-5.783j
6	-1+0j	-0.5+0j
7	-0.17158+0.67766j	-1+0j
8	-0.17158-0.67766j	-3.363E-08+0j

Abbildung 29: Die Eigenwerttabelle

Für jede Sonde wird eine eigene Spalte angelegt. Folgende Punkte sind zu beachten:

- Die Eigenwerte werden nur für lineare zeitkontinuierliche (ohne zeitdiskrete Glieder) und zeitinvariante Systeme ohne Totzeitglieder berechnet. Für die Totzeitglieder könnten sie jedoch eine geeignete Padé-Approximation auswählen.
- Diejenigen Systemteile, die keine Auswirkung auf den Messausgang haben, werden nicht berücksichtigt. In die Eigenwertberechnung werden nur diejenigen Systemteile einbezogen, die auf die gemessenen Ausgänge eine Auswirkung haben. Existieren mehrere Messpunkte, so gelten die Eigenwerte für das gesamte Teilsystem, das alle Übertragungstrecken enthält. Benötigen Sie die Eigenwerte für eine ganz bestimmte Teilstrecke, dann dürfen Sie nur den Ausgang dieser Strecke mit der Frequenzsonde verbinden.

7.5.4 Datentabellen

Für jede Frequenzsonde wird eine eigene Datentabelle erstellt, die alle Simulationsdaten der Diagramme enthält. Beachten Sie, dass für jede Sonde andere Frequenzzwischenwerte berechnet werden, da diese von den jeweiligen Signalverläufen abhängen.

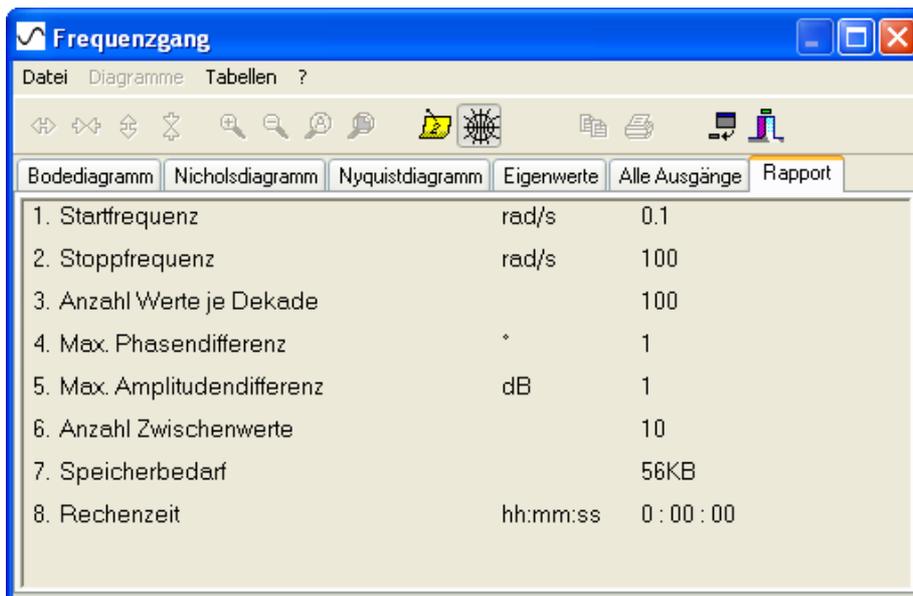
	Frequenz [rad/s]	Bode Strecke Ampl. [dB]	Bode Strecke Phase	Nyquist Strecke Realteil	Nyquist Strecke Imag. teil
0	0.1	8.2198	-4.0877	2.5697	-0.18365
1	0.10233	8.2272	-4.1867	2.5716	-0.18825
2	0.10471	8.235	-4.2883	2.5736	-0.19298
3	0.10715	8.2432	-4.3926	2.5756	-0.19785
4	0.10965	8.2518	-4.4996	2.5778	-0.20286
5	0.1122	8.2608	-4.6094	2.5801	-0.20801
6	0.11482	8.2702	-4.7222	2.5825	-0.21332
7	0.11749	8.2801	-4.8379	2.585	-0.21879
8	0.12023	8.2905	-4.9569	2.5876	-0.22442
9	0.12303	8.3013	-5.079	2.5903	-0.23023
10	0.12589	8.3127	-5.2045	2.5932	-0.23621
11	0.12882	8.3246	-5.3334	2.5963	-0.24237
12	0.13183	8.3371	-5.4659	2.5994	-0.24874

Abbildung 30: Datentabelle der Frequenzsimulation

Datentabellen können als Ganzes oder durch Zellen-, Zeilen- oder Spaltenmarkierung in Auszügen ausgedruckt oder in die Zwischenablage kopiert werden. Die Frequenzspalte wird immer mitgeführt, auch wenn sie nicht innerhalb des markierten Bereichs liegt.

7.5.5 Rapport

Im Rapport werden die wichtigsten Eckdaten der Simulation zusammengefasst. Der angegebene Datenspeicher umfasst die Simulationsergebnisse, die in den Tabellen angezeigt werden.



The screenshot shows a software window titled 'Frequenzgang' with a menu bar (Datei, Diagramme, Tabellen, ?) and a toolbar. Below the toolbar are tabs for 'Bodediagramm', 'Nicholsdiagramm', 'Nyquistdiagramm', 'Eigenwerte', 'Alle Ausgänge', and 'Rapport'. The 'Rapport' tab is active, displaying a table of simulation parameters.

1. Startfrequenz	rad/s	0.1
2. Stoppfrequenz	rad/s	100
3. Anzahl Werte je Dekade		100
4. Max. Phasendifferenz	°	1
5. Max. Amplitudendifferenz	dB	1
6. Anzahl Zwischenwerte		10
7. Speicherbedarf		56KB
8. Rechenzeit	hh:mm:ss	0 : 00 : 00

Abbildung 31: Rapport der Frequenzsimulation

8 Zeitsimulation

In diesem Kapitel erhalten sie detaillierte Informationen zur Zeitsimulation. Die Grundlagen dazu haben Sie bereits im [Einführungsbeispiel](#) vermittelt bekommen.

8.1 Blockdiagramm

Bevor Sie eine Simulation durchführen können, müssen Sie ein Blockdiagramm erstellen. Die Zeitsimulation ist im Gegensatz zur Frequenzsimulation nicht auf die linearen und zeitdiskreten Glieder beschränkt. Jetzt steht Ihnen die ganze Objektpalette zur Verfügung.

8.2 Quellen und Zeitsonden

Zur Durchführung einer Simulation benötigen Sie einerseits Quellen, die das System anregen, und andererseits Sonden, die es Ihnen ermöglichen, einzelne Systempunkte zu messen.

Zeitsonden sind jedoch nicht unbedingt erforderlich, da alle nicht verbundenen Ausgangsknoten (rot gezeichnet) oder die Ausgänge der Ausgangsglieder automatisch erfasst werden, sofern keine Zeitsonden vorhanden sind.

Bei der Frequenzsimulation übernehmen die Sonden auch die Funktion der Anregung (Sinussignal). Dies ist bei der Zeitsimulation nicht der Fall. Die Systemanregung erfolgt einzig durch die vorhandenen Quellglieder. Die Zeitsonden enthalten nur einen Eingangsknoten, der mit beliebig vielen Systempunkten verbunden werden kann.



Abbildung 32: Zeitsonde und Ausgangsglied

Sonden und Quellen werden mittels Signalleitungen mit dem System verbunden. Diejenigen Signalleitungen, die mit den Sonden verbunden sind, unterscheiden sich grafisch von den anderen Verbindungen.

Offene Eingänge (rote Knoten) in der obersten Zeichnungsebene (also nicht innerhalb von Anwenderblöcken) werden automatisch mit null angesteuert. Im Allgemeinen sollten Sie jedoch mindestens eine Hauptquelle als Systemanregung (Führungsgröße, Sollwert) und ev. weitere Quellen als Störgrössenaufschaltung benutzen.

Das Ausgangsglied können Sie verwenden, wenn Sie das Signal zur Darstellung noch skalieren und speziell beschriften möchten. Bei der Verwendung von Ausgangsgliedern sind die Ausgänge auch grafisch besser als solche erkennbar.

Wie bereits erwähnt, ist die Verwendung von Zeitsonden nicht obligatorisch. Wenn Ihr System jedoch zahlreiche Ausgänge besitzt, von denen Sie nur wenige untersuchen möchten, ist es vorteilhafter, wenn Sie diese auf eine Zeitsonde zu führen, da sonst das Zeitdiagramm mit unnützen Kurven überladen wird.

Jede Zeitsonde führt, im Gegensatz zu den Frequenzsonden, zu einem eigenen Diagramm. Mehrere Zeitsonden sind also nur sinnvoll, wenn Sie die Messsignale in Gruppen aufteilen wollen.

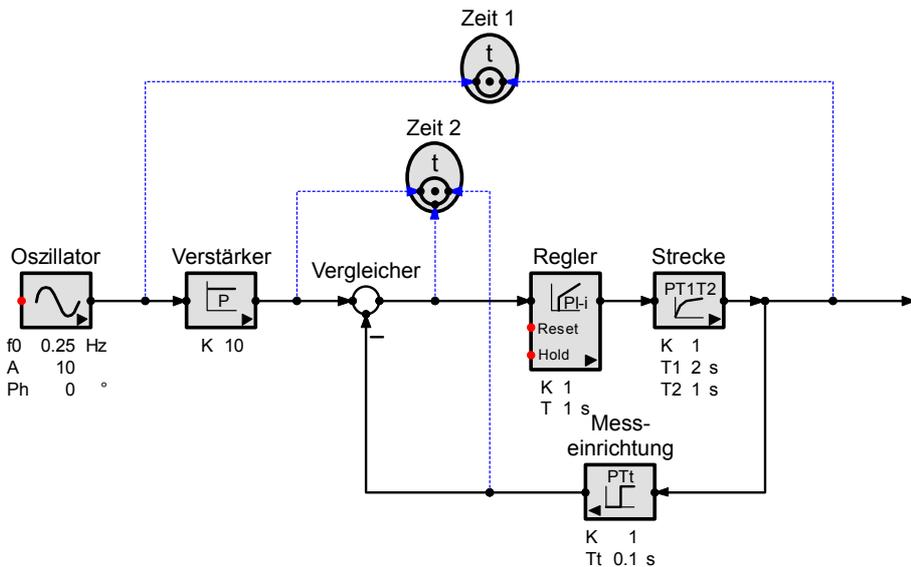


Abbildung 33: Der Einsatz von Zeitsonden

8.3 Quellengruppierung

Mehrere Quellen können zu einer Quellengruppe zusammengefasst werden (nicht zu verwechseln mit einer Zeichnungsgruppe). Die Zugehörigkeit zu einer bestimmten Gruppe wird durch die Gruppennummer bestimmt (Grp), die eine beliebige positive ganze Zahl sein kann und in der Zeichnung nur sichtbar ist, wenn sie grösser als null ist.

Oszillator



Grp 1
 f0 1 Hz
 A 1
 Ph 0 °

Abbildung 34: Quelle mit Gruppennummer (Grp)

Für jede Gruppe wird hintereinander eine vollständige Simulation durchgeführt. Die Ausgänge derjenigen Quellen, die zu anderen Gruppen gehören, geben den Wert 0 aus. Die Messsignale der einzelnen Gruppen werden jedoch alle im gleichen Diagramm dargestellt. Somit können Sie ein System mit verschiedenen Signalmustern anregen und die Ergebnisse direkt miteinander vergleichen.

Eine besondere Stellung haben die Quellen der Gruppe 0. Diese Quellen erscheinen in allen Gruppen. Sie könnten also eine Quelle als Führungsgrösse (Gruppe 0) und zwei andere Quellen als Störgrössen (Gruppe 1 und 2) vorsehen. SimApp führt dann zwei Simulationen aus, einmal mit der Störgrösse 1 und einmal mit der Störgrösse 2, wobei jeweils die gleiche Führungsgrösse benutzt wird.

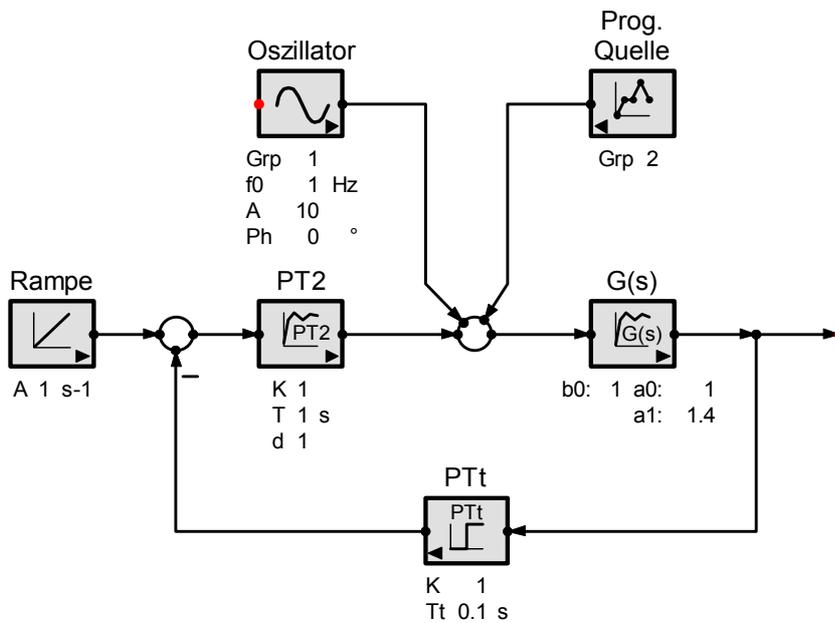


Abbildung 35: Beispiel zur Quellengruppierung

8.4 Weitere Bemerkungen zur Zeitsimulation

Eingänge und Ausgänge sind solche Knoten, die in der Zeichnung rot erscheinen. Wenn noch weitere Knoten rot sind, dann bedeutet dies, dass diese nicht verbunden sind. Dies ist meistens auf ungenaues Platzieren zurückzuführen.

Wenn gelbe Knoten oder gelbe Signalleitungen auftreten, liegt ein Verbindungsfehler oder ein unvollständiges Teilsystem vor.

Systemteile, die keine Auswirkung auf die gemessenen Knoten haben, werden nicht in die Simulation mit einbezogen.

Starre Regelschleifen ohne Zeitverzögerungen - sogenannte algebraische Schleifen - sind im Gegensatz zur Frequenzsimulation nicht erlaubt. Fügen Sie in solchen Fällen ein fiktives Verzögerungsglied 1. Ordnung (PT1-Glied) oder ein Totzeitglied in die Schleife ein und wählen Sie eine Zeitkonstante oder Totzeit, die genügend kurz ist, so dass das Systemverhalten nicht wesentlich gestört wird. Gegebenenfalls müssen Sie noch die Integrationsschrittweite verkürzen (ca. 1/10 der kürzesten Zeitkonstante des Systems).

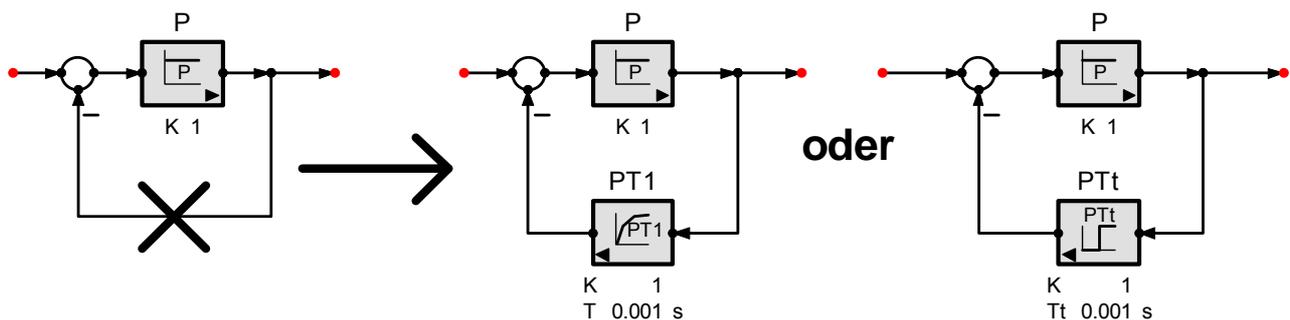


Abbildung 36: Starre Regelschleifen vermeiden

8.5 XY-Diagramme

Mit dem XY-Diagrammblock können zwei beliebige Signale des simulierten Systems in eine zweidimensionale Beziehung gebracht und dargestellt werden. Das Diagramm und seine Wertetabelle werden wiederum innerhalb des Zeitfensters dargestellt. Zeitsonden und XY-Diagramme können gleichzeitig und auch mehrfach eingesetzt werden.

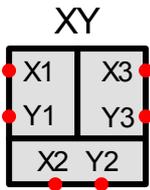


Abbildung 37: XY-Diagrammblock

Die XY-Diagrammblocke weisen drei Eingangskanäle zum simultanen Messen und Anzeigen von bis zu drei Signalpaaren auf. Unbenutzte Kanäle können einfach offen gelassen werden und bleiben unbeachtet.

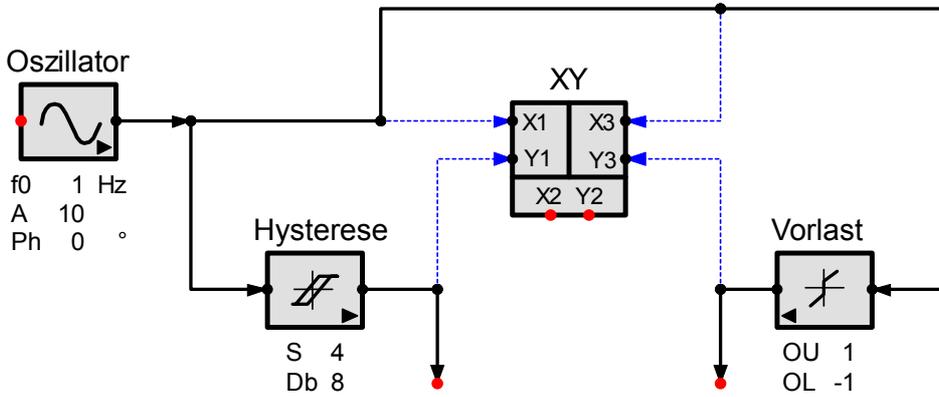


Abbildung 38: Hysterese- und Vorlastkennlinie mittels XY-Diagramm vergleichen

8.6 Simulationsparameter

Der Ablauf der Simulation wird durch mehrere Parameter und Optionen beeinflusst. Öffnen Sie das entsprechende Dialogfeld über das Hauptmenu *Zeitsimulation + Optionen...*, über die Simulationssymbolleiste oder über das Kontextmenü der Zeichnung.

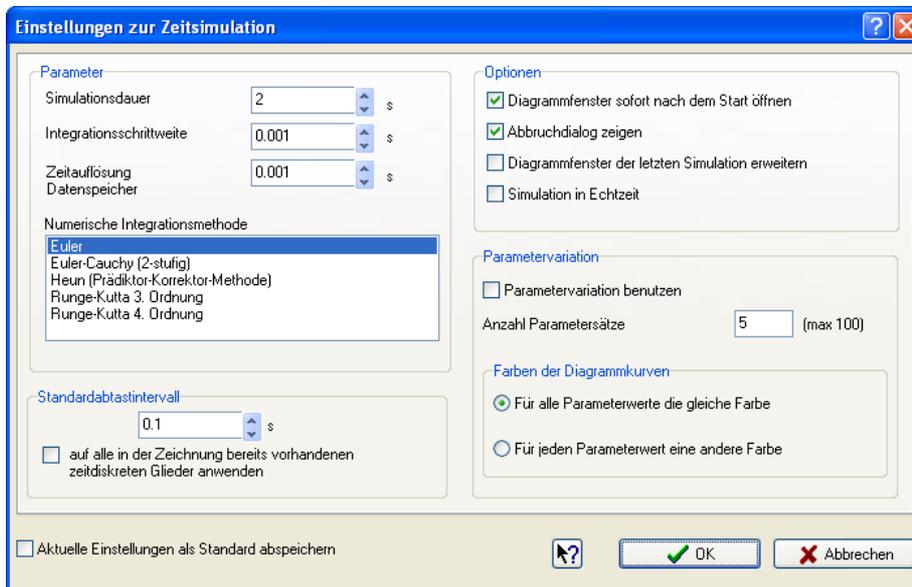


Abbildung 39: Einstellungen für die Zeitsimulation

Wählen Sie die Simulationszeit (Simulationsdauer). Eine Simulation beginnt immer zum Zeitpunkt null.

Numerische Integration

Die Simulation erfolgt durch die numerische Integration des Differentialgleichungssystems, das durch das System in grafischer Form beschrieben wird. Bei der numerischen Integration wird die Simulationszeit in eine Folge von identischen Zeitintervallen aufgeteilt. Die Breite eines solchen Intervalls wird Schrittweite oder Integrationsschrittweite genannt. Innerhalb eines solchen Intervalls wird die Lösungsfunktion durch eine geeignete Approximation, die von der gewählten Integrationsmethode abhängt, ersetzt und jeweils der nächste Punkt der Lösungskurve berechnet. Die numerische Integration stellt also eine Annäherung dar.

Diese Annäherung ist i. A. umso besser, je kürzer die Schrittweite und je aufwändiger die Integrationsmethode ist. Eine kurze Schrittweite und eine aufwändige Integrationsmethode bedeuten jedoch auch immer eine längere Rechenzeit.

Die optimale Schrittweite und die korrekte Integrationsmethode ergeben sich allein aus dem zu simulierenden System. Bei schwierigen Systemen, die zu numerischen Instabilitäten neigen, benötigen Sie gegebenenfalls mehrere Versuche. Als allgemeine Richtlinien können folgende Punkte dienen:

- Wählen Sie eine Schrittweite, die sicher kürzer als das Reaktionsverhalten des Systems ist. Als allgemeine Regel für gute Simulationsergebnisse gilt, die Schrittweite 10 Mal kleiner als die kürzeste Zeitkonstante, die im System vorkommt, zu wählen. Gemäss dem Anwendungsbereich aus dem Ihr System stammt, können Sie bereits eine grobe Einteilung vornehmen. Im Heizungs- und Lüftungsbereich existieren gewöhnlich Reaktionszeiten von Sekunden bis mehreren Stunden, bei Motoren sind sie im Mikro- und Millisekundenbereich.
- Die Verzögerungszeit von Totzeitgliedern muss genau ein Mehrfaches der Schrittweite sein. Wenn dies nicht der Fall ist, kann die tatsächliche Totzeit um die Schrittweite kleiner oder grösser sein. Wenn die Totzeit kleiner als die Hälfte der Schrittweite ist, verschwindet sie.
- Wählen Sie in erster Linie die Integrationsmethode von Euler. Sie ist die einfachste numerische Methode und garantiert die kürzesten Rechenzeiten. Bei kritischen Systemen, die zu numerischen Instabilitäten neigen, sollten Sie die Runge-Kutta-Methode vierter Ordnung benutzen. Allenfalls müssen Sie noch die Schrittweite verkürzen. Numerische Instabilitäten können Sie oft daran erkennen, dass Simulationsdurchgänge mit unterschiedlichen Schrittweiten und Integrationsmethoden zu ganz unterschiedlichen Antworten führen.
- Bei sprungförmigen Signalen können die mehrstufigen Methoden jedoch Ungenauigkeiten im Übergangsbereich erzeugen (Überschwingen, Verzerrungen). Benutzen Sie in solchen Fällen die Euler-Methode mit kurzer Schrittweite.

Die Güte, mit der Sie die berechneten Kurvenverläufe darstellen und abspeichern wollen, wird durch die Zeitauflösung des Datenspeichers bestimmt. Diese Einstellung übt keinen Einfluss auf die Rechengenauigkeit aus. Im Allgemeinen benötigen Sie zur Darstellung wesentlich weniger Stützstellen als zur Simulation. Die Zeitauflösung beeinflusst direkt den benötigten freien Arbeitsspeicher Ihres Computers, der für die Simulation bereitstehen muss. Bei zu grosser Zeitauflösung (kleine numerische Werte) kann der benötigte freie Arbeitsspeicher leicht mehrere Dutzend MBytes umfassen. Im Allgemeinen sollte die Zeitauflösung mindestens 10 Mal länger sein als die Integrations-schrittweite.

8.7 Zeitsimulation starten

Starten Sie die Zeitsimulation im Menü *Zeitsimulation + Start* oder mit der Startschaltfläche . Die Dauer der Simulation ist abhängig von der Systemgrösse und der Wahl der Simulationsparameter. Während einer längeren Simulation ist die Zeichnung gesperrt. In dieser Zeit können Sie jedoch an anderen Zeichnungen weiterarbeiten.

8.8 Resultate der Zeitsimulation

8.8.1 Zeitdiagramm

Das Zeitdiagramm enthält alle gemessenen Signalverläufe. Es gelten im Wesentlichen die gleichen Aussagen wie für das Bodediagramm bei der Frequenzsimulation.

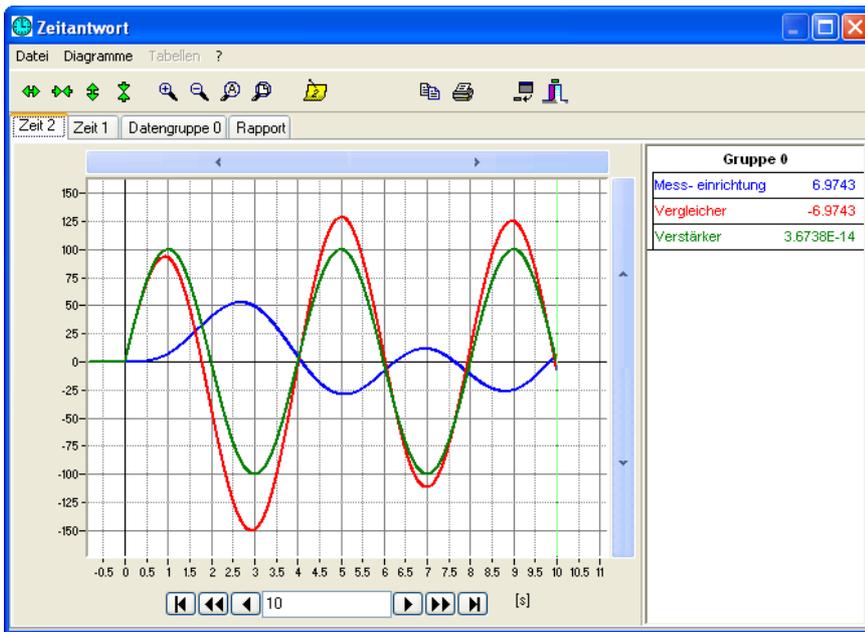


Abbildung 40: Zeitdiagramm

Die Legende ist nach Sonden und Quellengruppen hierarchisch gegliedert. Für jede Quellengruppe entsteht ein eigener Legendenabschnitt.

8.8.2 Datentabellen

Die Werte der Signalverläufe werden in Tabellen (ähnlich Frequenzsimulation) angezeigt. Je Quellengruppe wird eine eigene Tabelle erzeugt.

The screenshot shows the 'Zeitantwort' window with the 'Tabellen' view selected. It displays a table with the following data:

Index	Zeit [s]	Alle Ausgänge l.y(t)	Alle Ausgänge Produkt.y(t)
-1	-0.1	1	0
0	0	1	0
1	0.1	1	-0.2
2	0.2	0.98	-0.38416
3	0.3	0.94158	-0.53195
4	0.4	0.88839	-0.63139
5	0.5	0.82525	-0.68104
6	0.6	0.75715	-0.68793
7	0.7	0.68835	-0.66336
8	0.8	0.62202	-0.61905
9	0.9	0.56011	-0.56471
10	1	0.50364	-0.50731
11	1.1	0.45291	-0.45128
12	1.2	0.40778	-0.39909

Abbildung 41 Datentabelle

Der Werte bei Index -1 zur Zeit -0.1 sind die Anfangswerte vor Simulationsstart.

Die Datentabellen können Sie als Ganzes oder durch Zellen-, Zeilen- oder Spaltenmarkierung in Auszügen ausdrucken. Die Zeitspalte wird immer mitgedruckt, auch wenn sie nicht innerhalb des markierten Bereichs liegt.

8.8.3 Rapport

Der Rapport (Abbildung siehe Frequenzsimulation) zeigt eine Zusammenfassung der wichtigsten Eckdaten der Simulation. Der angegebene Speicherbedarf umfasst nur die simulierten Datenwerte.

8.9 Beispiele zur Zeitsimulation

8.9.1 Numerische Lösung von Differentialgleichungen

Ein einfaches Beispiel zeigt Ihnen, wie Sie mit SimApp eigene Differentialgleichungen lösen können:

Gegeben sei die Anfangswertaufgabe: $y' = -2 x y^2$, $y(0) = 1$

Gesucht werden die Näherungswerte mit verschiedenen Integrationsmethoden. Diese sollen in einer TABELLendarstellung miteinander verglichen werden.

Aus der Differentialgleichung lässt sich das folgende Blockdiagramm zeichnen:

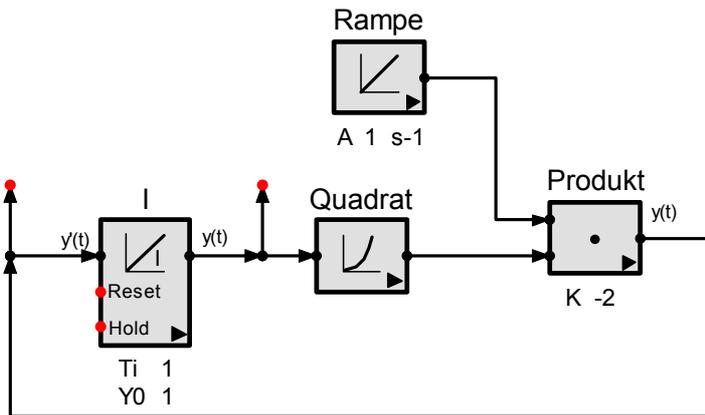


Abbildung 42: Blockdiagramm zum Lösen der Differentialgleichung $y' = -2xy$

Berechnen Sie nun für jede Integrationsmethode die ersten 10 Werte mit einer Schrittweite $h = 0.1$. Kopieren Sie diese Spaltenweise via Zwischenablage in eine Tabelle eines Tabellenkalkulationsprogramms (z.B. Microsoft Excel) und vergleichen sie mit der exakten Lösung $y(x) = 1 / (x^2 + 1)$.

x	exakt	Euler	Euler 2-stufig	Heun	Runge-Kutta 3. Ordnung	Runge-Kutta 4. Ordnung
0	1	1	1	1	1	1
0.1	0.9901	1	0.99	0.99	0.99013	0.9901
0.2	0.96154	0.98	0.96118	0.96137	0.9616	0.96154
0.3	0.91743	0.94158	0.91674	0.91725	0.91751	0.91743
0.4	0.86207	0.88839	0.8611	0.86195	0.86216	0.86207
0.5	0.8	0.82525	0.79889	0.80003	0.80009	0.8
0.6	0.73529	0.75715	0.73418	0.73553	0.73537	0.73529
0.7	0.67114	0.68835	0.67014	0.67159	0.6712	0.67114
0.8	0.609756	0.62202	0.60895	0.6104	0.6098	0.60976
0.9	0.552486	0.56011	0.55191	0.55329	0.55252	0.55249
1	0.5	0.50364	0.49964	0.50092	0.50002	0.5

Da die Funktion keine Sprungstellen enthält, bietet die Runge-Kutta Methode 4. Ordnung in allen Punkten die beste Annäherung, wobei aber auch der Rechenaufwand am grössten ist. Eine ausführlichere Diskussion des Themas finden Sie in [6].

9 Parametervariation

Mit der Parametervariation kann das Systemverhalten mit unterschiedlichen Parameterwerten schnell und einfach untersucht werden. Jeder Parameter kann bis zu 100 unterschiedliche Werte annehmen. SimApp führt automatisch für jeden Wert eine Simulation durch und stellt alle Ergebnisse im gleichen Diagrammfenster dar.

Haben mehrere Parameter variable Werte, spricht man von Parametersätzen. Z. B. alle i -ten variablen Werte der Parameter eines Gliedes bilden zusammen den i -ten Parametersatz des Gliedes. Alle i -ten Parametersätze aller Glieder zusammen bilden den i -ten Parametersatz der Zeichnung.

Die Anzahl der Parametersätze ist für jedes Glied individuell wählbar. Die Anzahl der Parametersätze, die für eine Simulation benutzt werden, kann in den Einstellungen zur Zeit- und Frequenzsimulation festgelegt werden. Die Anzahl der Parametersätze der einzelnen Glieder darf jedoch grösser oder kleiner sein.

Die Standardwerte der Parameter, die unterhalb des Blocksymbols in der Zeichnung und in den Simulationseigenschaften dargestellt werden, nehmen an der Parametervariation im Allgemeinen nicht teil. Es werden dafür Zusatzwerte benutzt.

Als Alternative zur Parametervariation könnte man auch mehrere Einzelsimulationen mit unterschiedlichen Parameterwerten durchführen und die Ergebnisse im gleichen Diagramm darstellen. Der Nachteil dieser Methode ist jedoch, dass das Diagramm unübersichtlicher wird und die Parameterwerte nicht im Voraus festgelegt werden können. Bei der Parametervariation werden die aktuellen Parameterwerte nicht verändert und müssen nicht nachträglich wieder zurückgestellt werden.

9.1 Einstellungen zur Parametervariation

Für die Parametervariation können an verschiedenen Orten Einstellungen vorgenommen werden.

9.1.1 Simulationseigenschaften

Hier werden für ein Glied die Anzahl der Parametersätze und diejenigen Parameter bestimmt, die an der Parametervariation teilnehmen. Durch drücken der Schaltfläche *Parametersätze bearbeiten* können, dann die Werte eingegeben werden.

Allgemein				
Verstärkung	K	1		<input checked="" type="checkbox"/> multipel
Zeitkonstante	T	1	s	<input checked="" type="checkbox"/> multipel
Dämpfung	d	0.5		<input checked="" type="checkbox"/> multipel
Anfangswert $y(t \leq 0)$	Y_0	0		<input type="checkbox"/> multipel
Anfangssteigung	Y_0'	0		<input type="checkbox"/> multipel

Abbildung 43: Parametervariation in den Simulationseigenschaften eines Gliedes

Folgende Einstellungen sind für die Parametervariation wichtig:

multipel

Für jeden Parameter, der an der Parametervariation teilnimmt, muss das Kontrollkästchen *multipel* aktiviert werden.

Anzahl Parametersätze

Legt die Anzahl der Parametersätze fest. Jeder Parameter, der die Option *multipel* aktiviert hat, kann diese Anzahl von Zusatzwerten aufnehmen.

Parametersätze bearbeiten

Öffnet ein Eingabefeld für die variablen Parameterwerte. Für jeden Parameter, dessen Option *multipel* aktiviert ist, wird darin eine Eingabespalte erzeugt.

In der Zeichnung unter dem Blocksymbol werden alle Parameter, die variable Parameterwerte aufweisen, rot ausgegeben.

9.1.2 Eingabefenster für die Parametersätze

Nr.	K	T [s]	d
0	1	1	0.5
1	2	1	0.01
2	3	1.5	0.02
3	4.5	2	0.06
4	5	2.5	0.1
5	6	3	0.2
6	7	3.5	0.3

Abbildung 44: Eingabefenster für variierende Parameterwerte

Die Werte in der gelb markierten Zeile 0 sind die aktuellen Parameterwerte und können hier nicht verändert werden. Sie dienen nur zur besseren Orientierung. In der Zeile 1 folgt der 1. Parametersatz. Diejenigen Parameter, die nicht an der Parametervariation teilnehmen, erscheinen hier nicht. Während der Parametervariation werden ihre aktuellen Werte benutzt.

9.1.3 Simulationseinstellungen

Abbildung 45: Einstellungen zur Parametervariation in den Simulationseinstellungen

Das Kontrollkästchen *Parametervariation benutzen* muss aktiviert sein.

Im Eingabefeld *Anzahl Parametersätze* wird die Anzahl der Parametersätze festgelegt, die für die Parametervariation benutzt werden. Diese Zahl muss nicht zwingend mit der Anzahl der Parametersätze in den Gliedern übereinstimmen. Die Glieder dürfen auch mehr oder weniger haben.

Wenn die Einstellungen der Anzahl Parameter in den Gliedern unterschiedlich sind, gelten die folgenden Regeln:

- Wenn ein Glied mehr Parametersätze aufweist, werden die Überzähligen nicht beachtet.
- Wenn ein Glied weniger Parametersätze aufweist, wird der letzte vorhandene Parametersatz weiterbenutzt
- Wenn ein Glied keine Parametersätze besitzt, werden die aktuellen Werte der Parameter benutzt.

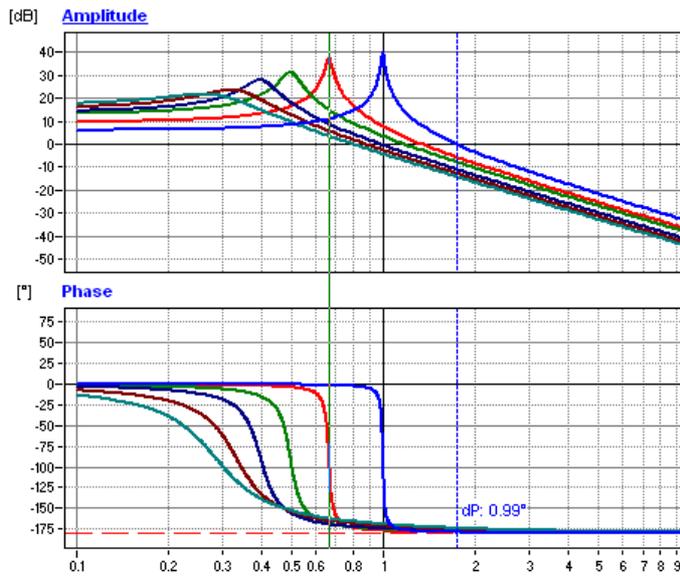
Für die Zeit- und Frequenzsimulation können unterschiedliche Werte vorgegeben werden.

9.2 Parametervariation starten

Der Start erfolgt wie bei einer normalen Simulation. Die Parametervariation wird durchgeführt, wenn das entsprechende Kontrollkästchen in den Simulationseinstellungen aktiviert ist.

Für jeden Parametersatz wird automatisch eine Simulation durchgeführt und alle Resultate im gleichen Diagrammfenster dargestellt.

In der Legende werden die Indizes der Parametersätze in Klammern angegeben. Die Farbenwahl kann in den Simulationseinstellungen beeinflusst werden.



Alle Ausgänge	
PT2	
(1) Ampl. [dB]	11.118
(1) Phase	-1.3738
(2) Ampl. [dB]	37.501
(2) Phase	-88.465
(3) Ampl. [dB]	15.096
(3) Phase	-168.34
(4) Ampl. [dB]	8.8493
(4) Phase	-169.36
(5) Ampl. [dB]	5.7362
(5) Phase	-165.05
(6) Ampl. [dB]	3.5471
(6) Phase	-162.5

10 Anwenderblöcke

Grosse und komplexe Blockdiagramme können sehr unübersichtlich werden und sind schwierig zu handhaben. Sie bestehen meistens aus mehreren, in sich abgeschlossenen Teilsystemen, die aber wegen der grossen Zahl von Gliedern nicht mehr klar voneinander abgrenzbar sind. Mit den Zeichnungsfunktionen von SimApp könnten Sie die Teilsysteme durch Ziehen von Linien und Rechtecken und den Einsatz von Farben und erläuternden Texten übersichtlicher gestalten. Dies würde das Problem aber nur teilweise lösen, da die Zeichnungen dadurch eher noch grösser werden und Änderungen aufwändiger machen.

Im Allgemeinen ist es nicht notwendig, alle Systemteile bis ins kleinste Detail darzustellen, da in jedem Anwendungsgebiet Teilsysteme existieren, deren Strukturen bestens bekannt sind und deshalb ohne Verlust an Information durch ein Ersatzsymbol dargestellt werden könnten. Meistens ändern von Anwendung zu Anwendung nur einzelne Parameterwerte.

Die Aufteilung in mehrere Teilsysteme und Bildung von wieder verwendbaren Einheiten lassen sich in SimApp durch Anwenderblöcke erreichen. Anwenderblöcke bestehen aus einem beliebigen System von Standardgliedern der Objektpalette und weiteren Anwenderblöcken. Sie besitzen ein eigenes Symbol, einen Namen, einen Parametersatz und verhalten sich wie Standardglieder. Sie lassen sich in die Objektpalette integrieren oder zu Bibliotheken zusammenfassen.

SimApp bietet zwei Methoden an, um aus Teilsystemen höher integrierte Blöcke zu bilden:

Bei der **ersten Methode** können Sie in der aktuellen Zeichnung einen beliebigen Ausschnitt auswählen und mit einem einzigen Befehl zu einem Block zusammenfassen. Bei der **zweiten Methode** erzeugen Sie den neuen Block in einem speziellen Entwurfswindow, der Blockmappe.

Die erste Methode ist praktisch und schnell und eignet sich besonders für spontane Vereinfachungen des Blockdiagramms, ohne Anspruch auf Wiederverwendbarkeit. Die zweite Methode ist aufwändiger, bietet aber keine Einschränkungen bezüglich der Parameterwahl und der grafischen Darstellung und unterstützt deshalb die Wiederverwendbarkeit in besserer Masse.

10.1 Einfache Blöcke durch Auswählen erzeugen

In einem Blockdiagramm können Sie einen beliebigen Ausschnitt durch Markieren auswählen und daraus einen Block erzeugen. Dieses Teilsystem darf sowohl Objekte der Objektpalette, also auch reine Zeichenobjekte wie Linien und Formen und andere Anwenderblöcke enthalten. Nach dem Auswählen benutzen Sie den Menübefehl *Extras + Einfacher Block bilden*. Die Auswahl wird sofort durch einen einzelnen Block mit einem Standardsymbol ersetzt. Er enthält die Ein- und Ausgangsknoten, einer Titelzeile sowie die Parameter der enthaltenen Glieder. Deaktivierte Blockknoten (Rechtsklick auf Knoten und *aktiv deaktivieren*) erscheinen nicht (siehe Reset und Hold-Eingang im nachfolgenden Beispiel).

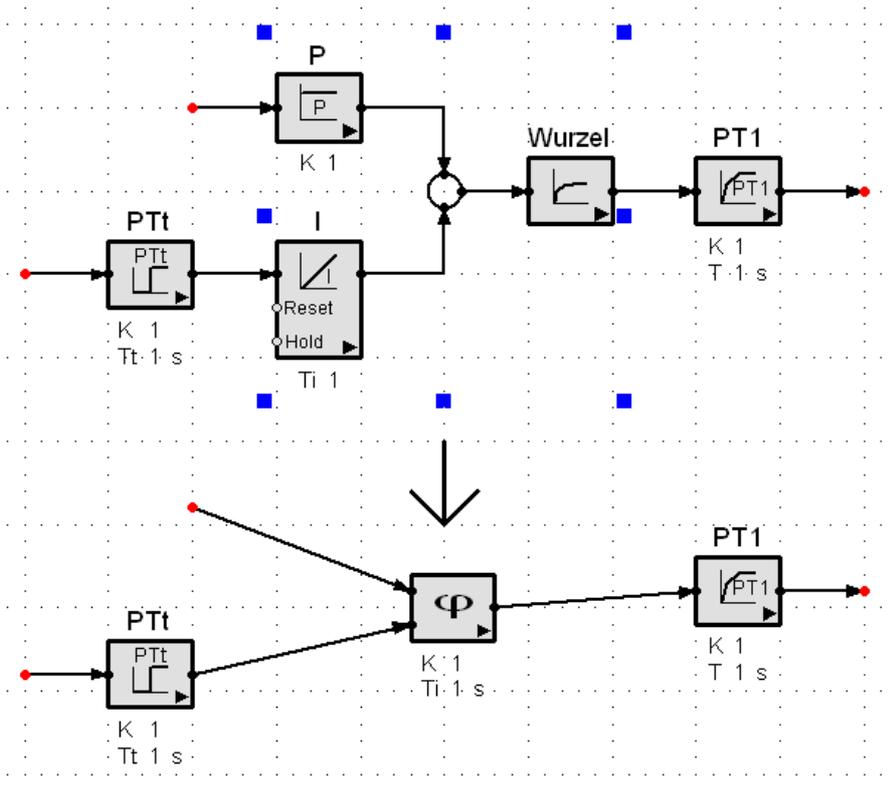


Abbildung 46: Objekte auswählen und Blöcke bilden

Die Ein- und Ausgangsknoten entstehen aus den Signalleitungen zu den nicht markierten Gliedern und aus den offenen Knoten innerhalb der Auswahl. Die Verbindungen zu den äusseren Gliedern werden nach der Blockbildung im Allgemeinen wieder hergestellt.

Bemerkungen zum Bilden von einfachen Blöcken:

- Signalleitungen zu nicht markierten Gliedern sollten ebenfalls nicht in die Markierung einbezogen werden, da sonst die nicht markierten Glieder abgetrennt werden.
- Die Ausgänge der Summierglieder müssen mit allen direkt verbundenen Signalleitungen markiert werden.
- Alle Parameter der markierten Glieder, werden als veränderliche Parameter in den neuen Block übernommen. Diese Parameter erscheinen auch auf der Datensseite der Simulationseigenschaften des neuen Blockes und können wiederum in einen noch höher integrierten Block übernommen werden. Die Eigenschaften dieser Parameter sind wiederum frei einstellbar (Name, Formelzeichen, etc.).
- Diejenigen Parameter, die sie nicht als veränderliche Parameter in den neuen Block übernehmen möchten, müssen Sie vor der Blockbildung ausblenden. Sie behalten ihre ursprünglichen Werte als Konstanten.
- Deaktivierte Knoten werden nicht übernommen.
- Die Parameter gruppierter Glieder sind nicht sichtbar.

Wenn Ihnen das Standardsymbol nicht gefällt oder zuwenig aussagekräftig ist, können Sie es in der Blockmappe weiter bearbeiten. In einer Blockmappe haben Sie auch Zugriff auf die innere Struktur des Blockes und die Möglichkeit neue Parameter zu erzeugen. Wählen Sie im Kontextmenu *Anwenderblock in Blockmappe öffnen*.

Einen Anwenderblock können Sie jederzeit mit dem Befehl *Extras + Block aufheben* wieder rückgängig machen. An die Stelle des Symbols tritt dann wieder sein inneres System.

10.2 Blöcke in der Blockmappe erzeugen

Das Erzeugen von Blöcken in der Blockmappe ist wegen der besseren Bearbeitungsmöglichkeit i. A. dem einfachen Erzeugen von Blöcken durch Auswählen vorzuziehen, da Sie die Knoten, die Parameter und das Symbol individueller bearbeiten können.

10.2.1 Einführungsbeispiel

Die Aufgabe besteht darin, aus dem Blockdiagramm eines Gleichstromantriebs einen parametrierbaren Anwenderblock mit eigenem Symbol zu erzeugen.

10.2.1.1 Mathematische Systembeschreibung

Die laplacierte Grundgleichungen des Gleichstromantriebes lauten [1]:

$$\begin{array}{ll}
 \text{Ankerstrom:} & I_A = \frac{K_A}{1+sT_A} U_A \quad K_A = \frac{1}{R_A} \text{ und } T_A = \frac{L_A}{R_A} \\
 \text{Differenzspannung:} & U_R = U_A - e_m \quad U_A = \text{Ankerspannung} \\
 \text{Gegenspannung:} & e_m = K_F \omega \quad K_F = \text{Motor-Konstante} \\
 \text{Antriebsmoment:} & M_A = K_F I_A \\
 \text{Differenzmoment:} & M_B = M_A - M_L \quad M_L = \text{Lastmoment} \\
 \text{Winkelgeschwindigkeit:} & \omega = \frac{1}{J s} M_B \quad J = \text{Trägheitsmoment von Motor und Last} \\
 \text{Drehzahl [min}^{-1}\text{]:} & n = 60\omega/2\pi
 \end{array}$$

Die Grundgleichungen beschreiben das innere Verhalten des Blockes. Die innere Struktur ist über drei Knoten mit der Aussenwelt verbunden. Als Steuereingang wirkt die Ankerspannung U_A , als Ausgang die Drehzahl n und als Störgrösse das Lastmoment M_L .

Um den Block für verschiedene Ausführungen von Gleichstrommotoren benutzen zu können, müssen mehrere Grössen von aussen einstellbar sein. Daraus ergibt sich der folgende Parametersatz:

Name	Formelzeichen	Einheit
Ankerinduktivität	L_A	H
Ankerwiderstand	R_A	Ohm
Motor-Konstante	K_F	Nm/A
Trägheitsmoment	J	kgm ²

Beachte: Das Hoch- und Tiefsetzen sowie die Verwendung von Sonderzeichen (z.B. Ω) ist nicht möglich.

10.2.1.2 Blockmappe öffnen

Zum Öffnen einer neuen Blockmappe wählen Sie im Menü *Datei + Neue Blockmappe*. Die Blockmappe besteht aus zwei Blättern und einer Tabelle. Das erste Blatt dient zur Bearbeitung des Symbols und das Zweite zum Entwurf der inneren Struktur des neuen Blockes.

10.2.1.3 Blockdiagramm entwerfen

Aus den Gleichungen des Gleichstromantriebes lässt sich direkt das folgende Blockdiagramm zeichnen:

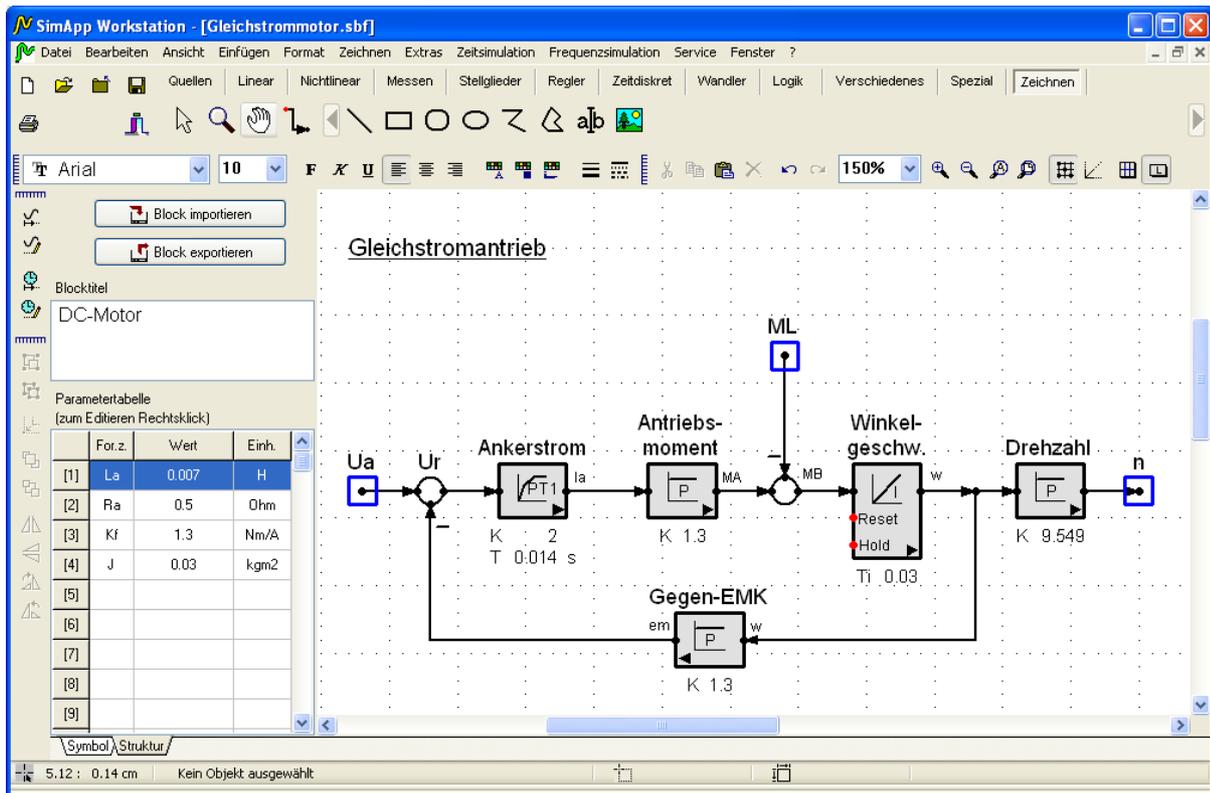


Abbildung 47: Die Blockmappe

Die Eingangs- und Ausgangsknoten werden mit dem Knotenobjekt der Palettenseite *Spezial* gebildet. Die Knoten werden automatisch nummeriert, wobei es jedoch vorteilhaft ist, die Nummern durch die korrekten Signalnamen zu ersetzen. Die Nummern oder Namen erscheinen auch im Symbolblatt.

10.2.1.4 Parameter erzeugen

Die Parameter der Standardglieder entsprechen nicht den neuen Parametern des Motors. Die Zeitkonstante des PT1-Gliedes ist z.B. der Quotient aus Ankerinduktivität und Ankerwiderstand. Wir benötigen also neue Parameter, die sich mit den Parametern der inneren Glieder via Formeln verbinden lassen.

1. Doppelklicken Sie auf die erste Zeile in der Parametertabelle und geben Sie die Eigenschaften für die Ankerinduktivität L_a ein:

The dialog box 'Blockparameter [1]' is used to define the parameters for the first block in the table. It contains the following fields:

- Name: Ankerinduktivität
- Formelzeichen: L_a (obligatorisch)
- Einheit: H
- Wert: 0.007
- Minimum-Eingabe: 0
- Maximum-Eingabe: 0

Buttons: OK, Abbrechen

Abbildung 48: Blockparameter festlegen

2. Als Wert wählen Sie einen typischen Wert für ein konkretes Beispiel.
3. Schliessen Sie das Dialogfeld. Das Formelzeichen, der numerische Wert und die Einheit erscheinen nun in der ersten Zeile der Parametertabelle.
4. Wiederholen Sie nun Punkt 1 und 2 für den Ankerwiderstand, die Motorkonstante und das Trägheitsmoment.
5. Nachdem die neuen Motorparameter erzeugt sind, müssen die Beziehungen zu den Parametern der inneren Glieder definiert werden. Öffnen Sie das Dialogfeld des PT1-Gliedes, das den Titel „Ankerstrom“

trägt und danach die Parametereigenschaften und klicken auf den Reiter *Zeitkonstante*. Im Eingabefeld *Formel* geben Sie nun die Formel $[1]/[2]$ ein.

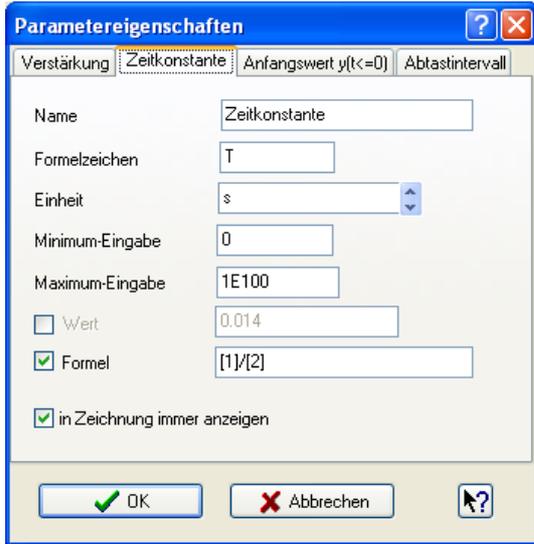


Abbildung 49: Formel eingeben

6. Die Werte in eckigen Klammern sind die Indizes der neuen Blockparameter in der Parametertabelle. [1] ist der Platzhalter für die Ankerinduktivität L_a . Die Formel $[1]/[2]$ bedeutet also den Quotient aus L_a und R_a . Die Formel wird durch SimApp bei jeder Simulation ausgewertet. Neben den Grundoperationen +, -, *, / sind eine Reihe weiterer Funktionen sowie eine beliebige Klammerung erlaubt (siehe weiter unten). Nachdem Sie die Formel eingegeben haben, wiederholen Sie dasselbe für die Verstärkung.
7. Sie haben nun die Abhängigkeit der Standardparameter des PT1-Gliedes von den neuen Motorparametern festgelegt. Wiederholen Sie Punkt 4 für die übrigen Glieder. Für die Drehzahl beim Proportionalglied benötigen Sie keine Formel. Geben Sie einfach den Wert 9.549 ein. Erlaubt wäre aber auch die Formel $(60/(2*\pi))$, die zum selben Wert führt.

10.2.1.5 Blocksymbol entwerfen

Nach dem Entwurf der inneren Struktur – spätere Änderungen sind jederzeit möglich – wenden wir uns dem äusseren Erscheinungsbild, dem Blocksymbol, zu. Nachdem Sie ins Symbolblatt gewechselt haben, erblicken Sie einen leeren Blockrahmen und drei Knoten, welche die Namen der Ein- und Ausgänge tragen. Falls Sie keine Namen zugewiesen haben, sind sie mit (1), (2) und (3) durchnummeriert. Dies sind die neuen Blockknoten, die Sie nun geeignet auf dem Rahmen platzieren können, am besten Eingänge links und Ausgänge rechts. Stellen Sie sicher, dass das automatische Einrasten aktiviert ist! Die Knoten müssen mit dem Raster unbedingt übereinstimmen.

Der leere Rahmen hat die Grösse eines Standardgliedes auf $(10 \times 8 \text{ mm}^2)$. Sie können ihn aber auch vergrössern. Zeichnen Sie nun darin ein aussagekräftiges Symbol.

10.2.1.6 Block zusammenführen

Jetzt werden die beiden Teile zu einem Ganzen vereint. Wählen Sie im Menü *Extra + Blockmappe + Block in die Zwischenablage exportieren* oder drücken Sie einfach den Knopf *Block exportieren*. Der vollständige Block liegt nun in der Windows-Zwischenablage.

Wechseln Sie nun in eine andere Zeichnung (keine Blockmappe !) oder öffnen Sie eine neue. Fügen Sie den Block aus der Zwischenablage in die Zeichnung ein.

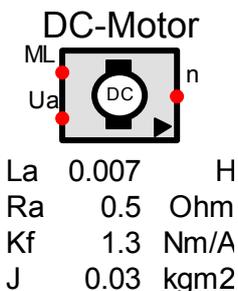


Abbildung 50: Der fertige Block

Falls die Parameterliste nicht korrekt ist, müssen Sie zurück ins Strukturblatt der Blockmappe wechseln und die notwendigen Korrekturen vornehmen. Den neuen Block können Sie nun wie jedes andere Glied verwenden.

den, ihn in eine Bibliothek abspeichern (siehe Kapitel [Bibliotheken](#)) oder in die Palette einfügen (siehe Kapitel [Palette bearbeiten](#)).

10.2.1.7 Zusammenfassung

Die wichtigsten Schritte nochmals in Kürze:

1. Neue Blockmappe öffnen.
2. Blockparameter in Parametertabelle erzeugen.
3. Im Strukturblatt das Blockdiagramm des inneren Systems entwerfen.
4. Formeln für die inneren Parameter eingeben.
5. Innere Struktur austesten.
6. Ein- und Ausgangsknoten mit Knotenobjekten verbinden.
7. Im Symbolblatt ein geeignetes Blocksymbol entwerfen und die Knoten korrekt platzieren (auf Raster!).
8. Block exportieren um Struktur und Symbol zusammenzuführen (Windows Zwischenablage).
9. Block aus der Windows Zwischenablage in eine beliebige Zeichnung einfügen.

10.2.2 Beziehung zwischen Symbol- und Strukturblatt

Die innere Struktur eines Anwenderblockes wird über die äusseren Blockknoten mit anderen Gliedern verbunden. Diese Knoten können aber nicht direkt im Symbolblatt erstellt werden. Sie entstehen automatisch, durch Setzen von Knotenobjekten im Strukturblatt. Für jedes neue Knotenobjekt wird im Symbolblatt automatisch ein Blockknoten erzeugt. Knotenobjekte und Blockknoten stellen ein untrennbares Paar dar. Wenn Sie im Strukturblatt das Knotenobjekt löschen, verschwindet auch der zugehörige Blockknoten im Symbolblatt. Dies funktioniert auch in der entgegengesetzten Richtung. Knotenobjekt und Blockknoten erhalten automatisch eine identische Erkennungsnummer um ihre Zusammengehörigkeit zu verdeutlichen. Es ist vorteilhaft, die Nummer durch den korrekten Signalnamen zu ersetzen.

10.2.3 Die Parametertabelle

In der Parametertabelle werden die äusseren Parameter des Anwenderblockes entworfen. Diese Parameter erscheinen im neuen Block unterhalb des Blocksymbols und weisen die gleichen Eigenschaften wie diejenigen der Standardglieder auf. Jede einzelne Zeile kann einen Parameter enthalten. In der ersten Spalte ist der Platzhalter (Nummer in Klammer) sichtbar, der für die Formeleingabe in den inneren Gliedern benutzt wird.

Doppelklicken Sie in eine beliebige Zeile oder drücken Sie die Eingabetaste um einen neuen Parameter zu erzeugen oder einen Bestehenden abzuändern. Die Zeilen sind frei wählbar und Leerzeilen sind erlaubt. Im fertigen Block erscheinen die Parameter in der gleichen Reihenfolge, wobei die freien Zeilen aber entfernt sind. Durch die Lehrzeilen haben Sie nachträglich noch die Möglichkeit, neue Parameter einzufügen ohne die bestehende Parameter verschieben zu müssen. Die Parameter lassen sich durch *Strg+Del* wieder löschen.

Das Festlegen eines Formelzeichens im Eigenschaftendialog ist obligatorisch. Wenn es fehlt, wird der Parameter nicht angenommen oder gelöscht. Die Zahlenwerte der Parameter können willkürlich sein, da sie im fertigen Block jederzeit wieder verändert werden können. Es ist jedoch ratsam, typische Werte eines bekannten Systems zu verwenden, damit SimApp die Verknüpfungen mit den Parametern der inneren Glieder korrekt testen kann. Ausserdem erscheinen diese Werte beim Einsatz des Anwenderblockes als Standardwerte der Parameter.

10.2.4 Innere Struktur entwerfen

Zum Entwurf der inneren Struktur können alle Objekte, also auch Formen, der Palette benutzt werden. Beachten Sie, dass durch innere Quellen, die nicht die Gruppennummer 0 besitzen, zusätzliche Simulationsdurchläufe ausgelöst werden können. Verbinden Sie die Gruppennummern der inneren Quellen gegebenenfalls mit einem äusseren Parameter.

10.2.5 Formeln für die inneren Parameter festlegen

Die inneren Parameter haben entweder feste Werte, die im fertigen Block von aussen nicht mehr verändert werden können, oder sind durch die Eingabe einer Formel von einem oder mehreren Blockparametern abhängig. Der Bezug auf die Blockparameter kann sehr einfach sein. Zum Beispiel könnte der Werte eines inneren Parameters immer genau dem Wert eines äusseren Parameters entsprechen. Es sind jedoch auch komplexere Bezüge möglich, die als Funktion eingegeben werden. Die Eingabe der Bezüge erfolgt in den Parametereigenschaften der inneren Glieder. Wenn Sie das Kontrollkästchen für die Formel aktivieren, wird während einer Simulation der aktuelle Wert mit der Formel und den Werten der Blockparameter berechnet.

Zur Auswertung der Formel wirkt im Hintergrund ein Formelinterpreter. Die Formel wird als freier Text eingegeben, der jedoch bestimmten Regeln unterworfen ist.

Funktionen, Operatoren und Konstanten des Formelinterpreters:

Funktion	Beschreibung
+ - * / ^	Addition/ Subtraktion/ Multiplikation/ Division/ Potenz
!	Fakultät
sin()	Sinus (Radiant)
cos()	Cosinus (Radiant)
tan()	Tangens (Radiant)
arcsin()	Arcussinus (Radiant)
arccos()	Arcuscosinus (Radiant)
arctan()	Arcustangens (Radiant)
ln()	natürlicher Logarithmus
log()	dekadischer Logarithmus
exp()	e hoch x
sqr()	Quadrat
sqrt()	Wurzel
int()	ganzzahliger Anteil
frac()	nicht ganzzahliger Anteil
abs()	Absolutwert
rnd()	Runden auf ganzzahlige Werte
Pi	Zahl $\pi = 3.14159...$

Für die Eingabe einer Formel sind beliebige Klammerungen erlaubt. Zwischen Gross- und Kleinschreibung wird nicht unterschieden. Die Formel wird beim Schliessen des Dialogfeldes syntaktisch geprüft und numerisch ausgewertet. Wenn sie Bezüge auf Blockparameter enthält, werden deren aktuelle Werte benutzt. Wenn die Formel einen syntaktischen oder numerischen Fehler (z.B. Division durch 0, Zahlenüberlauf, etc.) enthält, wird in einer Fehlermeldung die Stelle in der Formel, wo der Fehler aufgetreten ist, mit einem Fragezeichen (?) markiert. Durch die Eingabe von sinnvollen und typischen Werten für die Blockparameter können Sie Abbrüche während der Simulation durch numerische Fehler verhindern.

Hier einige Beispiele für syntaktisch korrekte Formeln:

5.67	Der Parameter hat den konstanten Wert 5.67
[3]	Der Parameterwert ist gleich dem Werte des Blockparameters aus der 3. Zeile der Parametertabelle.
sin([3])	Sinus des Blockparameterwertes aus der 3. Tabellenzeile
2.5 * exp(-4/[6]) * sin(1.76)	
Pi^[4]	
[2] * (1 - [4] * (Pi - sin([1])))	

Achtung: Wenn ein Bezug auf einen nicht existierenden Blockparameter erfolgt, wird automatisch der Wert 0 benutzt. Es erfolgt keine Fehlermeldung, falls kein syntaktischer oder numerischer Fehler auftritt!

10.2.6 Inneres System austesten

Bevor Sie den neuen Block bilden, sollten Sie die innere Struktur austesten. Im Strukturblatt können Sie, wie in einem normalen Zeichnungsfenster, Zeit- und Frequenzsimulationen durchführen. Allerdings lassen sich die Ergebnisse nicht abspeichern und gehen beim Schliessen der Blockmappe verloren. Besser ist es, das Blockschaltbild in ein normales Zeichnungsfenster zu kopieren und dort auszutesten. Knotenobjekte werden bei Simulationen ohne Messsonden zu den Ein- und Ausgängen gezählt. In einem normalen Zeichnungsfenster werden die Formeln nicht beachtet. Es werden stets die aktuellen Parameterwerte benutzt.

10.2.7 Knotenobjekte und Blockknoten

Diejenigen Knoten im Strukturblatt, die einen Ein- oder Ausgang des Blockes darstellen, müssen mit einem Knotenobjekt verbunden werden. Das Einsetzen von Knotenobjekten im Strukturblatt bewirkt ein automatisches Erzeugen von Blockknoten im Symbolblatt. Zusammengehörnde Knotenobjekte und Blockknoten weisen immer identische Namen auf. Wenn Sie in einem Blatt die Knotennamen ändern, werden sie im anderen Blatt sofort aktualisiert.

10.2.8 Blocksymbol entwerfen

Der Blockrahmen mit dem Richtungspfeil wird automatisch erzeugt und lässt sich beliebig vergrössern und verkleinern.. Wenn Sie ihn aus Versehen löschen, lässt er sich einfach wieder herstellen. Fügen Sie ein Rechteck in die Zeichnung ein und aktivieren Sie in den Formateigenschaften die Option *Bildet Hintergrund oder Rahmen für Anwenderblock*. Den Richtungspfeil ist auf der Seite *Spezial* der Palette zu finden.

Im Symbolblatt dürfen nur Formen und keine Glieder verwendet werden. Das Einfügen von Gliedern wird durch SimApp nicht verhindert. Diese werden jedoch beim Exportieren des Blockes wieder entfernt.

Für die Titelzeile des Blockes steht oberhalb der Parametertabelle ein Eingabefeld zur Verfügung.

Entwerfen Sie ein Symbol, das die Funktionalität des Blockes deutlich macht. Benützen Sie gegebenenfalls Bildobjekte aus anderen Zeichnungsprogrammen.

Das Verhalten und die Bearbeitungsmöglichkeiten der Zeichnungsobjekte, die das Symbol bilden, können durch verschiedene Optionen beeinflusst werden.

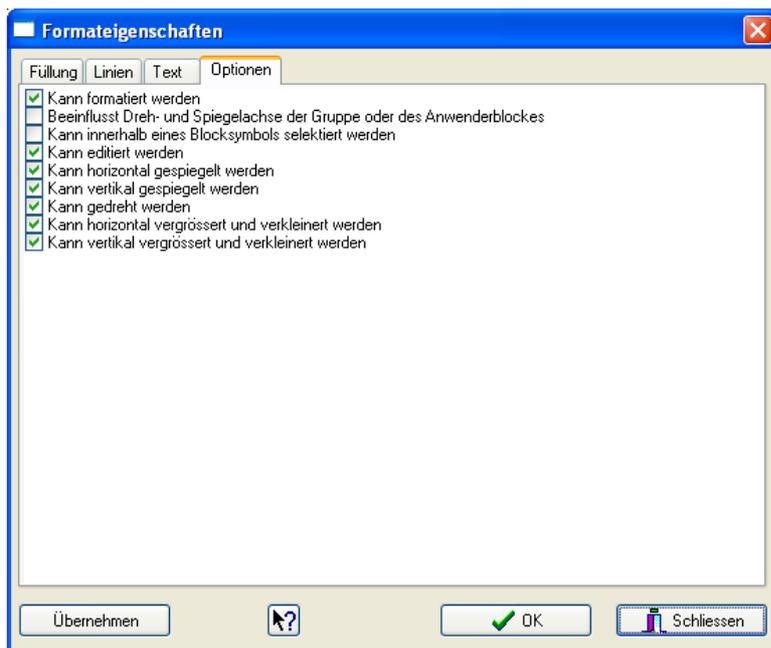


Abbildung 51: Optionen für ein Textobjekt

Die Kontrollkästchen können drei Zustände annehmen. Graue Häkchen entstehen, wenn mehrere Objekte selektiert sind, die nicht identische Einstellungen aufweisen. Beim Schliessen führen graue Häkchen zu keinen Veränderungen. Durch Klicken kann die Option so verändert werden, dass sie für alle Objekte gilt.

Benutzen Sie die kontextbezogene Programmhilfe, um mehr Informationen über die Einstellmöglichkeiten zu erhalten.

10.2.9 Struktur und Symbol zusammenführen (exportieren)

Das Zusammenführen von Struktur und Symbol erfolgt, wenn Sie den Block in die Windows-Zwischenablage exportieren. Dazu dürfen Sie aber nicht die Standardkopierfunktion benutzen. Wählen Sie im Menü *Extras +Blockmappe + Block in die Zwischenablage exportieren* oder drücken Sie die *Exportieren*-Taste.

10.2.10 Blöcke verwenden und abspeichern

Nach dem Kopieren in die Zwischenablage können Sie Ihren neuen Block wie jedes andere Standardglied einsetzen. Sie sollten jedoch nicht vergessen, ihn mindestens an einem Ort abzuspeichern. Dies kann innerhalb einer normalen Zeichnungsdatei, in der Palette oder in einer Bibliothek sein.

Wenn Sie den Block gesichert haben, benötigen Sie den Inhalt der Blockmappe nicht mehr. Alle Informationen sind im Block selber gespeichert, den Sie jederzeit wieder abändern können.

10.2.11 Blöcke überarbeiten

Anwenderblöcke können zur Überarbeitung jederzeit wieder in eine Blockmappe kopiert werden. Kopieren Sie den Block mit *Bearbeiten + Kopieren* in die Zwischenablage. Zum Einfügen in die Blockmappe dürfen Sie jedoch nicht den Standardbefehl *Bearbeiten + Einfügen* benutzen. Dies würde den vollständigen Block in das Symbol- oder in das Strukturblatt einfügen, ohne ihn zu öffnen. Wählen Sie stattdessen *Extras + Blockmappe + Block aus Zwischenablage importieren*. Der Block wird geöffnet und in die beiden Blätter der Blockmappe eingefügt. Es spielt dabei keine Rolle, welches Blatt gerade sichtbar ist.

Noch einfacher geht es, wenn Sie im Kontextmenu eines Anwenderblockes auf *Anwenderblock in Blockmappe öffnen* klicken.

10.2.12 Weitere Bemerkungen zum Entwurf von Anwenderblöcken

- Sie können mehrere Blockmappen öffnen und gleichzeitig mehrere Anwenderblöcke erstellen.
- Das Struktur- und das Symbolblatt können gemeinsam in eine Datei abgespeichert werden. Wählen Sie *Datei speichern* oder *Datei speichern unter*. Die Datei erhält die Namensweiterung für Blockmappen (**.sbf**).
- SimApp akzeptiert auch Zeit- und Frequenzsonden innerhalb eines Blockes. Sie wirken sich gleich aus wie äussere Sonden. Der Einsatz solcher Sonden ist fraglich, da sie in der Zeichnung nicht sichtbar sind, aber dennoch die Simulation beeinflussen.
- Anwenderblöcke besitzen im Gegensatz zu vielen Standardgliedern nur ein Symbol.
- Wenn Ihr Block ein oder mehrere zeitdiskrete Glieder enthält, müssen Sie für deren Abtastintervalle einen Blockparameter vorsehen. Der Bezug muss direkt sein (z.B. [3]) und darf keine Formelausdrücke enthalten ([1]/[2] oder $5*[5]$).
- Zu Testzwecken können innere Signal auch ohne Knoten mit einem Sende-Empfängerpaar nach aussen geführt werden.

11 Palette bearbeiten

Die Palette enthält alle Standardobjekte, die Sie in ihre Zeichnungen einfügen können. Sie besteht aus mehreren übereinander liegenden Seiten, die über Reiter zugänglich sind. Jede Seite enthält eine bestimmte Kategorie von Objekten.

Die Palette lässt sich auch verändern und erweitern, indem z.B. neue Seiten erstellt und mit beliebigen Objekten (Anwenderblöcke, Standardglieder, Formen, Teilsysteme) aufgefüllt werden.

Folgende Möglichkeiten zur Palettenbearbeitung stehen zur Verfügung:

- Neue Seiten erzeugen
- Seiten verschieben, Anordnung verändern
- Palette durch neue Objekte erweitern
- Seiten umbenennen
- Seiten löschen
- Schaltflächen verschieben
- Palette laden und speichern

11.1 Seiten erzeugen, löschen und umbenennen

Für diese Funktionen benötigen Sie das Kontextmenü der Palette, das durch Klicken mit der rechten Maustaste geöffnet wird (nicht auf Schaltflächen).

Neue Seite

Klicken Sie auf *Neue Seite*. Geben Sie einen Namen ein, der in der Palette noch nicht existiert.

Seite löschen

Klicken Sie auf *Seite löschen*. Die oberste Seite wird sofort gelöscht. Wenn sich noch Schaltflächen darin befinden, wird eine Bestätigung zum Löschen aller Schaltflächen und der damit verbundenen Objekte verlangt. Standardseiten können nicht gelöscht werden.

Seiten umbenennen

Klicken Sie auf *Seite umbenennen*. Standardseiten können nicht umbenannt werden.

11.2 Reiter und Schaltflächen verschieben

Die Reiter für die Seiten können Sie mit der Maus verschieben. Bei den Schaltflächen muss zusätzlich die Alt-Taste benutzt werden.

11.3 Objekte in Palette speichern

Sie können beliebige Objekte in die Palette abspeichern. Dies können Standardglieder, Anwenderblöcke, Formen, Text oder auch gruppierte Objekte sein. Ursprungsorte dieser Objekte können Zeichnungen oder Bibliotheken sein. Das Ablegen erfolgt durch einfaches Ziehen und Loslassen.

11.3.1 Objekte aus Zeichnungen einfügen

Wählen Sie zuerst eine Seite aus, in die Sie Ihre Objekte speichern möchten. Selektieren Sie dann die gewünschten Objekte in der Zeichnung und verschieben sie in Richtung Palette. Wenn Sie die Objekte in die Palette kopieren möchten (d.h. die Ursprungsobjekte sollen in der Zeichnung erhalten bleiben), müssen Sie während des ganzen Vorganges die Strg-Taste gedrückt halten, ansonsten werden die Objekte in die Palette verschoben und stehen in der Zeichnung nicht mehr zur Verfügung. Wenn Sie mit dem Mauszeiger an den Rand des Zeichnungsfensters kommen, setzt normalerweise das Autoscrolling ein. Sie können dies verhindern, indem Sie zusätzlich die Leertaste gedrückt halten.

Wählen Sie einen geeigneten Ort auf der aktuellen Palettenseite und lassen Sie die Maustaste los. Es erscheint eine neue Schaltfläche mit einem Standardsymbol, die Sie, wie oben beschrieben, noch innerhalb der aktuellen Seite verschieben können. Die Eigenschaften der neuen Schaltfläche können Sie nach ihren Wünschen einstellen (siehe Abschnitt *Schaltflächen bearbeiten*).

11.3.2 Objekte aus Bibliotheken einfügen

Wählen Sie eine Seite in der Palette aus und öffnen Sie eine Bibliothek. Klicken Sie bei gedrückter Alt-Taste auf eine Schaltfläche und verschieben Sie diese in die Palette.

11.4 Schaltflächen bearbeiten

Beim Speichern von Objekten in die Palette werden neue Schaltflächen erzeugt. Die Standardbilder (Index) sind jedoch wenig aussagekräftig. Sie sollten den Text deshalb durch einen geeigneten Namen oder ein Symbol ersetzen.

11.4.1 Eigenschaften von Schaltflächen

Klicken Sie mit der rechten Maustaste auf die gewünschte Schaltfläche und wählen Sie im Kontextmenü *Eigenschaften*.



Abbildung 52: Schaltfläche bearbeiten

Sie können eine beliebige Bitmap mit der Größe 24 x 24 Pixel laden. Wenn eine Bitmap geladen ist, wird jedoch der Titel nicht mehr angezeigt. Die Bitmap können Sie z.B. mit dem Programm Paint von Microsoft Windows oder einem anderen Malprogramm erzeugen.

Sie können der Schaltfläche auch einen Hinweistext zuordnen, der immer dann erscheint, wenn Sie die Maus auf der Schaltfläche positionieren und nicht klicken.

11.4.2 Mit Microsoft Paint eigene Schaltsymbole erzeugen

Paint finden Sie standardmässig im Ordner *Programme + Zubehör*. Öffnen Sie Paint und wählen Sie im Hauptmenü *Bild + Attribute*. Wählen Sie folgende Einstellungen:

Breite:	24
Höhe:	24
Masseinheit:	Pixel
Farben:	Farben

Wählen Sie den grössten Zoomfaktor und malen Sie das Symbol. Um die verschiedenen Schaltzustände zu verdeutlichen, wird die Farbe des Hintergrundes durch SimApp gesteuert, wobei Sie jedoch selber festlegen müssen, welche Pixel zum Hintergrund gezählt werden. SimApp erkennt den Hintergrund anhand des Pixels in der linken unteren Ecke. Alle anderen Pixel mit der gleichen Farbe werden zum Hintergrund gezählt. Benutzen Sie eine Farbe, die in der Grafik sonst nicht erscheint.

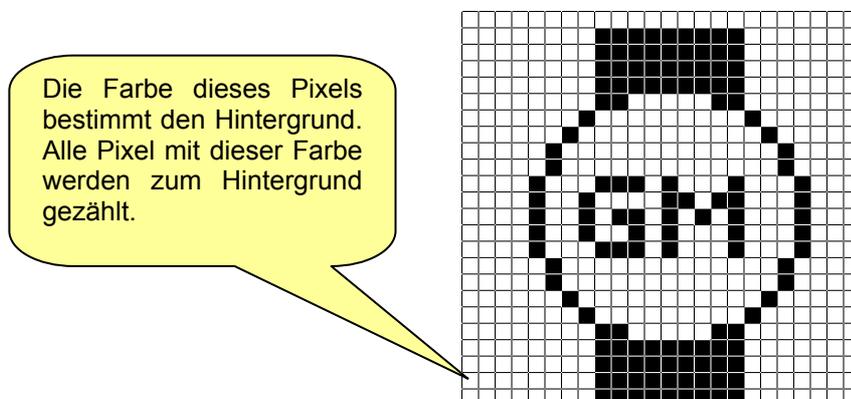


Abbildung 53: Schaltflächensymbol

Speichern Sie das Bild unter einem beliebigen Namen, gehen wieder zurück zu SimApp und laden Sie das Bild in die Schaltfläche. Das Bild wird vollständig kopiert. Die Bitmapdatei wird nach dem Laden nicht mehr benötigt.

11.5 Palette laden, speichern, wieder herstellen

Geänderte Paletten können Sie abspeichern und später wieder laden. Beim Programmstart wird jeweils die letzte aktuelle Palette benutzt. Die aktuelle Palette ist diejenige, die als letzte geladen oder abgespeichert worden ist. Die Originalpalette können Sie jederzeit im Kontextmenu wieder herstellen.

12 Bibliotheken

Sie können einen beliebigen Zeichnungsausschnitt markieren und in eine Bibliothek zur Wiederverwendung ablegen. Selbst Standardglieder der Palette lassen sich in Bibliotheken kopieren. Bibliotheken verhalten sich wie normale Symbolleisten.

12.1 Beispiel

Anhand des folgenden Beispiels werden die Funktionen zur Erzeugung und Benutzung von Bibliotheken gezeigt. Damit Sie keinen grossen Zeichnungsaufwand bewältigen müssen, soll eine neue Bibliothek erzeugt werden, die bloss mit Standardgliedern belegt wird, auch wenn dies keinen grossen Sinn macht. Die Abläufe sind jedoch in jedem Fall die gleichen.

Öffnen Sie eine neue Zeichnung und platzieren Sie darin zwei Standardglieder aus der Palette. Erzeugen Sie eine neue Bibliothek im Menü *Extras + Bibliothek + Neu*. Es erscheint ein Fenster mit gelbem Hintergrund. Klicken Sie auf ein Glied in der Zeichnung und verschieben Sie es in das Bibliotheksfenster. Nehmen Sie nun das zweite Glied, drücken jetzt aber noch die Strg-Taste und verschieben es ebenfalls in das Bibliotheksfenster. Beachten Sie, dass sich das erste Objekt verschoben (steht in der Zeichnung nicht mehr zur Verfügung) und das zweite Glied kopiert hat. Wählen Sie nun in der Palette ein drittes Objekt aus. Bevor Sie auf die Schaltfläche klicken drücken Sie die Alt-Taste. Verschieben Sie die Schaltfläche in das Bibliotheksfenster. Ihr Bibliotheksfenster hat nun etwa folgendes Aussehen:

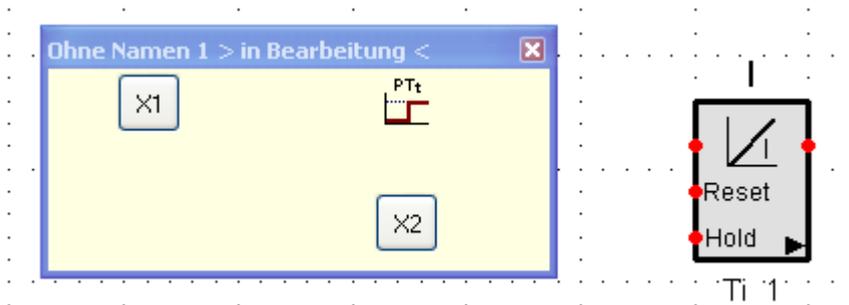


Abbildung 54: Bibliothek erzeugen

Sie können die Schaltflächen durch anklicken und verschieben neu anordnen. Klicken Sie mit der rechten Maustaste in das Bibliotheksfenster und wählen Sie im Kontextmenü *Umbenennen*. Geben Sie den Namen *Test* ein. Dies ist nun der neue Name der Bibliothek, der in der Titelleiste angezeigt wird.

Wenden wir uns nun den Symbolen der drei Schaltflächen zu:

Die beiden Objekte, die Sie aus der Zeichnung eingefügt haben, besitzen nun je eine graue Schaltfläche mit den Bezeichnungen X1 und X2. Das Objekt, das Sie von der Palette eingefügt haben, hat keine Veränderung erfahren. Die Texte in den Schaltflächen X1 und X2 können Sie verändern oder durch ein Bild ersetzen. Das Bearbeiten von Schaltflächen in Bibliotheken gestaltet sich gleich wie in der Palette (Kapitel [Palette bearbeiten](#)).

Öffnen Sie wieder das Kontextmenü der Bibliothek und wählen Sie *Bearbeitungsmodus*. Durch das Ausschalten des Bearbeitungsmodus wird das Fenster augenblicklich verkleinert und die Schaltflächen werden zusammen geschoben, erscheinen jedoch in der gleichen Reihenfolge.

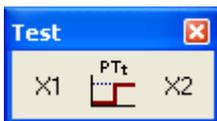


Abbildung 55: Die fertige Testbibliothek

Wählen Sie im Kontextmenü (Rechtsklick auf Titelleiste) *Speichern* und speichern Sie die Bibliothek unter dem Namen *Test.slb* ab.

Wählen Sie wieder im Kontextmenü *Ausblenden*. Die Bibliothek verschwindet, bleibt aber trotzdem noch im Speicher verfügbar, da sie bloss nicht mehr sichtbar ist. Sie können sie im Menü *Ansicht + Symbolleisten....* wieder einblenden.

Wählen Sie im Kontextmenü der Bibliothek *Schliessen*. Wenn Sie die Bibliothek noch nicht gesichert haben, erfolgt nun zuerst eine Aufforderung zum Speichern.

Zum Laden klicken Sie im Menü auf *Extras + Bibliothek laden....* und wählen im Öffnen-Dialog die Datei *Test.slb* aus.

13 Katalog der Standardglieder

13.1 Quellen

Zum Einstellen der Signalverläufe stehen Ihnen verschiedene, quellenspezifische Parameter zur Verfügung. Die meisten Quellen verfügen über folgende drei Grundparameter:

Offset

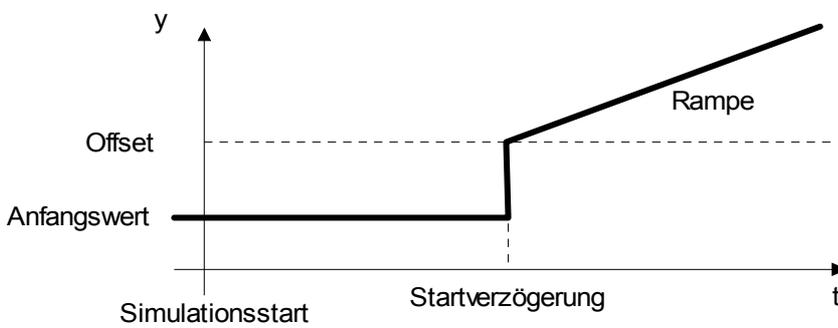
Mit dem Offset wird der Signalverlauf nach oben oder nach unten verschoben.

Startverzögerung

Mit der Startverzögerung wird eine begrenzte Zeitdauer definiert, die verstreicht, bis die Quelle nach dem Simulationsstart den vorgegebenen Signalverlauf ausgibt. Mit der Startverzögerung lassen sich mehrere Quellen miteinander synchronisieren.

Anfangswert

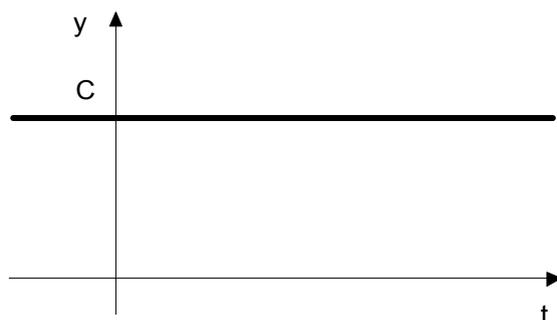
Der Anfangswert bestimmt den Ruhezustand vor dem Simulationsstart und die Ausgangsgröße nach dem Start, bis die Startverzögerung vorüber ist.



13.1.1 Konstante

Mit der Konstantenquelle kann ein beliebiger positiver oder negativer Wert in das System eingespeist werden, der sich über die gesamte Simulationszeit nie ändert. Offset, Startverzögerung und Anfangswert existieren bei dieser Quelle nicht.

Funktion: $y(t) = C$



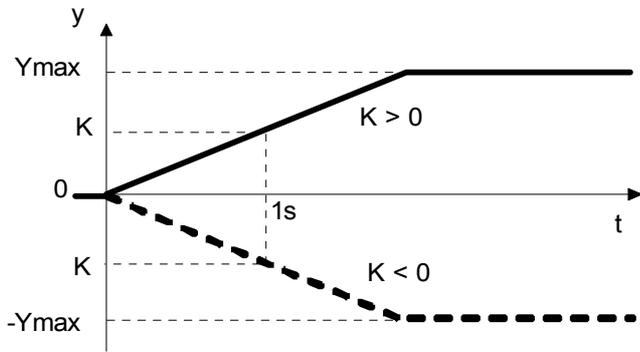
Konstante



13.1.2 Rampe

Die Rampe weist bis zum Maximalwert eine konstante Steigung auf. Der Sättigungswert Y_{max} wird immer als positive Zahl eingegeben. Wenn die Steigung positiv ist, wirkt er als Maximalwert, andernfalls als Minimalwert mit negativem Vorzeichen.

Funktion: $y(t) = A t$



Rampe

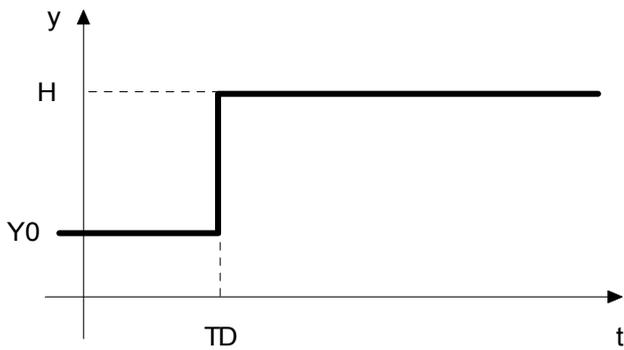


A 1 s⁻¹

13.1.3 Schritt

Der Zeitpunkt des Schrittes wird mit der Startverzögerung festgelegt.

Funktion: $y(t) = H \sigma(t-TD)$



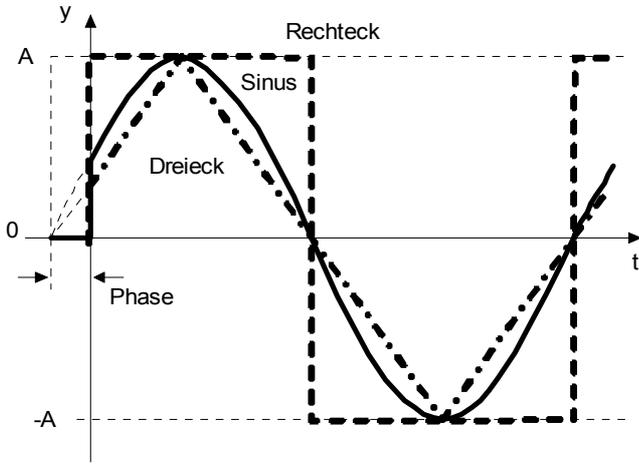
Schritt



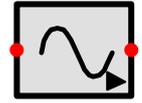
H 1
TD 0 s

13.1.4 Oszillator (Sinus, Dreieck, Rechteck)

Mit dem Oszillator lassen sich Sinus-, Rechteck- und Dreieckssignale generieren. Frequenz, Amplitude und Anfangsphase sind frei wählbar.



Oszillator

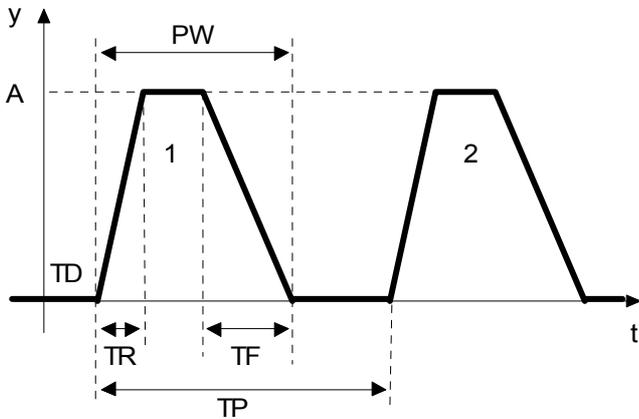


f0 1 Hz
A 1
Ph 0 °

Mit dem Modulationseingang kann die Ausgangsfrequenz gesteuert werden und der Generator als VCO (Voltage Controlled Oscillator, Spannungsgesteuerter Oszillator) betrieben werden. Die Ausgangsfrequenz ist entweder proportional zum Eingangssignal ($f = f_0 \times \text{Mod}$, lineare Modulation) oder exponentiell davon abhängig ($f_0 \times 10^{\text{Mod} \times x}$, exponentielle Modulation). Wenn der Modulationseingang nicht benutzt wird, ist die Ausgangsfrequenz konstant $f = f_0$.

13.1.5 Pulsgenerator

Der Pulsgenerator erzeugt eine Folge von identischen Pulsen. Die Folge kann entweder aus nur einem ($N=1$), aus mehreren (Bsp. $N = 6$) oder aus unendlich ($N < 0$) vielen Pulsen bestehen.



Puls

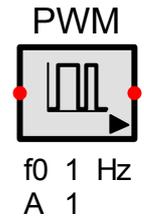
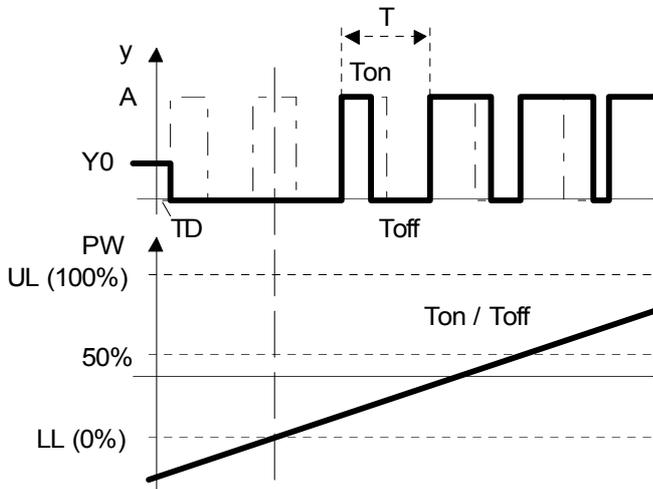


A 1
PW 1 s
TP 2 s
N 1
TD 0 s

Die ansteigende Flanke hat gegenüber der Absteigenden den Vorrang. Bevor die absteigende Flanke beginnen kann, wird zuerst die ansteigende Flanke zu Ende geführt. Wenn die ansteigende Flanke länger als die Pulsweite dauert, wird sie am Ende der Pulsweite unterbrochen und die absteigende Flanke entfällt.

13.1.6 Pulsweitenmodulator (PWM)

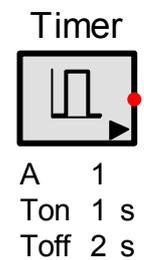
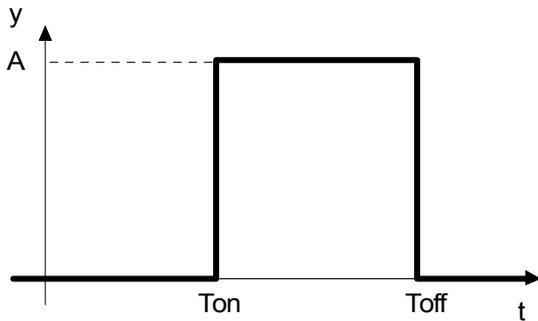
Der Pulsweitenmodulator (oder Pulsbreitenmodulator) gibt ein Rechtecksignal mit einer gewünschten Frequenz, Phase und Amplitude aus. Das Verhältnis zwischen Einschalt- und Ausschaltzeitdauer Ton / Toff kann über den Eingang zwischen 0 und 100% gesteuert werden. Der Einschaltzeitpunkt ist immer zu Beginn einer neuen Periode der Grundschwingung. Die Grenzwerte für 0 und 100% sind beliebig wählbar, wobei auch ein negativer Wirkungssinn zulässig ist (0%-Wert grösser als 100%-Wert).



13.1.7 Timer

Der Timer generiert einen einzelnen Rechteckimpuls. Ein-, Ausschaltzeitpunkt und Impulshöhe können vorgegeben werden.

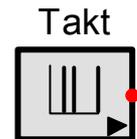
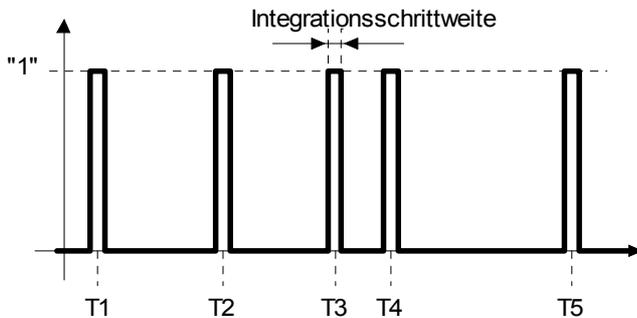
Funktion: $y(t) = H [\sigma(t-Ton) - \sigma(t-Toff)]$



13.1.8 Taktquelle

Die Taktquelle gibt eine vorgegebene Sequenz von Taktimpulsen aus. Die Höhe der Impulse entspricht dem logisch 1 Pegel und deren Breite der aktuellen Integrationsschrittweite. Die Taktquelle eignet sich zum Auslösen von zeitlich festgelegten Ereignissen in einem gesteuerten Prozess.

Die Taktsequenz, also die genauen Impulszeiten, wird in den Simulationseigenschaften der Quelle festgelegt. Durch Setzen der Option *Iterativ* wird die Sequenz nach dem Erreichen des letzten Taktimpulses bis zum Ende der Simulation fortwährend wiederholt. Wenn der erste Taktimpuls beim Zeitpunkt 0 liegt, wird dieser von den Wiederholungen ausgeschlossen, weil der Letzte Vorrang hat.



Die Folge der Zeitwerte T_n kann in eine ANSI/ASCII-Textdatei abgespeichert werden. Das Laden ist jederzeit möglich. Die maximale Anzahl der Impulse ist auf 10'000 beschränkt.

Das Format der Zeitwerte in der Textdatei entspricht demjenigen der Dateiquelle und der programmierbaren Quelle, wobei hier jedoch kein Wertepaar, sondern nur ein Zeitwert pro Zeile benötigt wird.

Beschreibung des Datenformats in der Textdatei

- Pro Zeile ein Zeitwert. Als Dezimaltrennzeichen ist Komma oder Punkt zulässig. Eine Datenzeile kann mit einem Kommentar, eingeleitet mit einem Sternzeichen *, abgeschlossen werden.
- Kommentarzeilen mit einer Zahl am Anfang, müssen mit einem Sternzeichen * beginnen (Leerzeichen davor erlaubt).
- Die Datei muss eine ANSI- oder ASCII-Textdatei sein. Als Namenserweiterung wird **.txt** vorgezogen, aber es kann auch jede andere gültige Windows Namenserweiterung sein.

Werden die Daten in eine Datei gespeichert, ist das Dezimalzeichen abhängig von den Einstellungen in den Programmoptionen.

Beispiel für Taktquelle:

Erzeugt durch Bruno Büsser, 29. November, 2002

0 *Erster Impuls bei Simulationsstart. Wird nicht wiederholt.

3.67 *Zweiter Impuls. Die Zeitwerte sind in aufsteigender Reihenfolge
(>Leerzeilen werden übersprungen)

5

6,2 *Dezimalzeichen kann auch ein Komma sein

8 *Dies ist der letzte Impuls

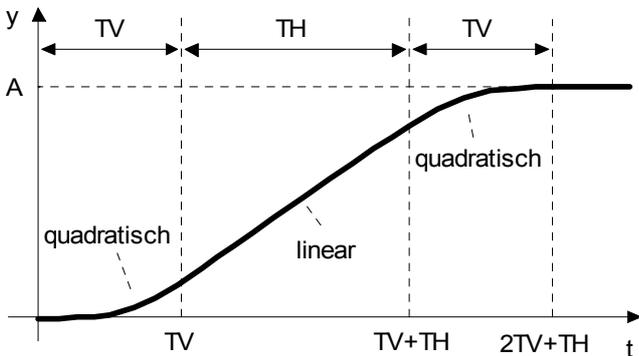
* 5 Zeitwerte insgesamt >> * wird benötigt, da Kommentar mit einer Zahl beginnt
Durch Setzen der Option *Iterativ* wird die Sequenz von 3.67 bis 8 fortlaufend wiederholt.

Ende der Beispieldatei

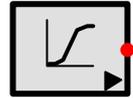
13.1.9 Fahrkurve

Mit der Fahrkurve werden Anfahrverläufe simuliert.

Funktion: $y(t) = f(A, TV, TH)$



Fahrkurve

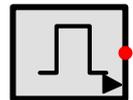


A 1
TV 1 s
TH 5 s

13.1.10 Rauschquelle, Zufallszahlengenerator

Mittels einer Rauschquelle wird das dynamische Verhalten eines Systems beim Auftreten von nichtdeterministischen Störungen untersucht.

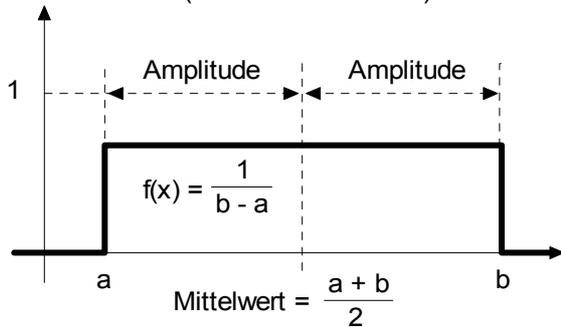
Rauschen



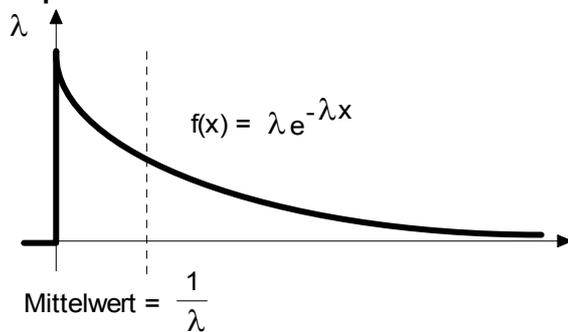
M 1
SD 0.4
A 1

Die Rauschquelle unterstützt vier verschiedene Rauschverteilungen:

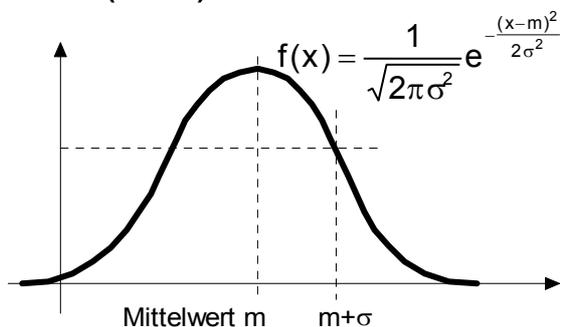
Gleichverteilt (Weisses Rauschen)



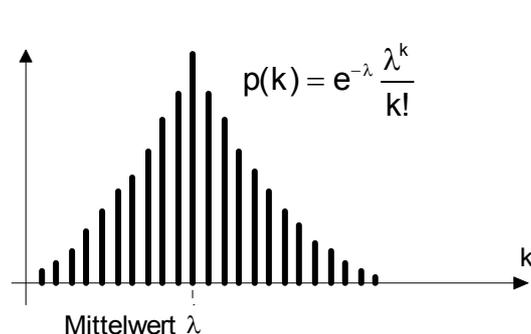
Exponentiell



Normal (Gauss)



Poisson



Optionen

Die Option *Reproduzierbar* bestimmt den Startpunkt des internen Zufallszahlengenerators.

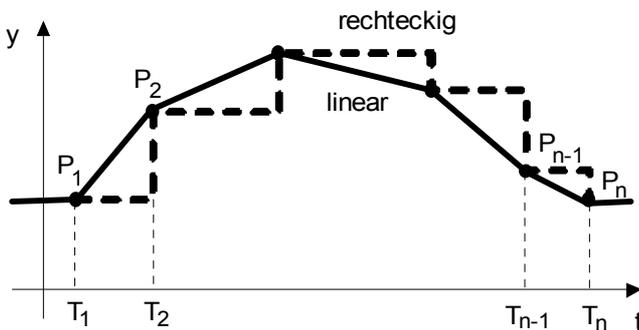
Wenn *Reproduzierbar* aktiviert ist, gibt die Rauschquelle bei jeder Simulation die gleiche Sequenz aus, auch wenn die Zeichnung abgespeichert und wieder geladen wird.

Wenn *Reproduzierbar* ausgeschaltet ist, liefert die Quelle bei jeder Simulation eine andere Rauschsequenz. Sind mehrere Rauschquellen im System vorhanden, gibt jede eine andere Sequenz aus (unabhängig von der Option *Reproduzierbar*)

13.1.11 Programmierbare Quelle

Diese Quelle erzeugt sein Ausgangssignal aus einer beliebigen Folge von Stützpunkten. Es kann zwischen einer linearen und einer rechteckigen resp. treppenförmigen Interpolation der Punkte gewählt werden.

Funktion: $y(t) = f[P_1 \dots P_n, t]$



Die Funktion kann mit einer Anfangsbedingung oder den ersten Punkt beginnen. Eine Anfangsbedingung wird durch einen ersten fiktiven Punkt mit einer Zeit von kleiner als Null definiert. Der Datenwert gilt dann bis zum Ende der Startverzögerung T_D . Nach der Startverzögerung gilt der Wert des ersten realen Punktes. Fall kein erster Punkt mit einer Zeit kleiner als Null vorhanden ist, gilt von Anfang an der Wert des ersten Punktes.

Die Startverzögerung verzögert die ganze Funktion um ihren Wert (nach rechts). Ein Punkt an der Stelle 2s kommt also bei einer Startverzögerung von 1s auf die Stelle 3s zu liegen.

Sprungstellen entstehen, wenn zwei aufeinander folgende Punkte den gleichen Zeitwert aufweisen.

Das Kausalitätsprinzip muss eingehalten werden, d.h. die Zeitwerte müssen in aufsteigender Reihenfolge geordnet sein. Wenn mehr als zwei aufeinander folgende Punkte den gleichen Zeitwert besitzen, werden nur der erste und letzte Punkt berücksichtigt.

Durch Setzen der Option *Iterativ* wird die Funktion, wenn der letzte Punkt vorüber ist, laufend wiederholt. Wenn der erste Punkt beim Zeitpunkt null liegt, wird er von den Wiederholungen ausgeschlossen, da der letzte Punkt Vorrang hat.

Die Datenwerte werden innerhalb des Gliedes gespeichert, können jedoch auch in eine ANSI/ASCII Textdatei gespeichert und daraus geladen werden. Die Anzahl der Datenpunkte ist auf 10'000 beschränkt. Beim Einlesen einer grösseren Datei werden die überzähligen Punkte abgetrennt.

Beschreibung des Datenformats in einer Textdatei:

- Pro Zeile ein Y-t Wertepaar.
- Y- und t-Werte müssen mit Leerschlag, Tabulator oder Semikolon voneinander getrennt werden. Gültige Dezimalzeichen sind Punkt und Komma. Eine Datenzeile kann mit einem Kommentar, eingeleitet mit einem Sternzeichen *, abgeschlossen werden.
- Alle Zeilen, die nicht mit einer Ziffer, Komma oder Punkt beginnen (ev. nach Leerzeichen) werden als Kommentarzeile interpretiert.
- Die Datei muss eine ANSI oder ASCII-Textdatei sein. Als Namenserweiterung wird **.txt** vorgezogen, aber es kann auch jede andere gültige Windows Namenserweiterung sein. Die Anzahl der Datenpunkte ist auf 10'000 beschränkt. Beim Einlesen einer grösseren Datei werden die überzähligen Punkte abgetrennt.

Wenn die Daten der Quelle in eine Datei gespeichert werden, werden die Wertepaare mit TAB getrennt. Das Dezimalzeichen ist dabei abhängig von den Einstellungen in den Programmoptionen.

Beispieldatenfile

Erzeugt durch Bruno Büsser, 30. November, 2002

Diese Datei enthält 5 Wertpaare

-1; 0.5 *Optionaler fiktiver Punkt für die Anfangsbedingung

2.45; 1 *Dies ist der erste Stützpunkt. Zeit zuerst, dann Ausgangswert

3.67; 2.5 *Zweiter Stützpunkt. Die Zeitwerte sind in aufsteigender Reihenfolge.

5 3.14 *Die Werte werden nun mit einem Leerzeichen getrennt

(>Leerzeilen werden übersprungen)

6.23 2,5 *Dezimaltrennzeichen kann auch ein Komma sein

Zeilen, die nicht mit einer Ziffer, Punkt oder Komma beginnen werden übersprungen

8;0 *Dies ist das letzte Wertepaar

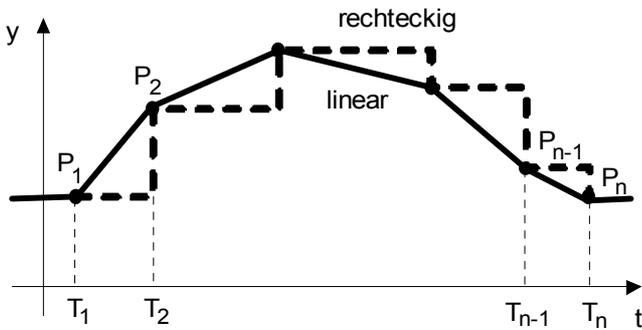
* 5 Wertpaare wurden eingelesen >> * wird benötigt, da Kommentar mit einer Zahl beginnt

Ende der Beispieldatei

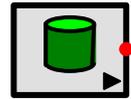
13.1.12 Dateiquelle

Die Dateiquelle kann bis zu 50 simultane Ausgänge aufweisen, deren Daten aus einer Textdatei geladen werden. Der Pfad auf diese Datei wird in der Dateiquelle gespeichert. Beim Übertragen der Quelle auf einen anderen Computer, muss die Datei auch mitkopiert werden (nicht automatisch durch SimApp), wobei der eingetragene Dateipfad wegen der unterschiedlichen Verzeichnisstruktur seine Gültigkeit verlieren kann. Der Pfad kann absolut (mit Laufwerkangabe) oder relativ sein. Relative Pfade beziehen sich auf das Verzeichnis der Zeichnung.

Funktion: $y(t) = f[P_1 \dots P_n, t]$ (Pro Kurve)



Dateiquelle



Für jedes Ausgangssignal wird ein Ausgangsknoten benötigt. Die Anzahl der Datenreihen innerhalb der Textdatei muss mindestens der Anzahl der Ausgangsknoten entsprechen. Überzählige Datenreihen sind zulässig und haben keine Auswirkung.

Die Ausgangssignale können mit einer Anfangsbedingung oder den ersten Punkt beginnen. Eine Anfangsbedingung wird durch einen ersten fiktiven Punkt mit einer Zeit von kleiner als Null definiert. Der Datenwert gilt dann bis zum Ende der Startverzögerung TD. Nach der Startverzögerung gilt der Wert des ersten realen Punktes.

Fall kein erster Punkt mit einer Zeit kleiner als Null vorhanden ist, gilt von Anfang an der Wert des ersten Punktes. Die Startverzögerung verzögert das ganze Signal um ihren Wert (nach rechts). Ein Punkt an der Stelle 2s kommt also bei einer Startverzögerung von 1s auf die Stelle 3s zu liegen.

Durch Setzen der Option *Iterativ*, wird die Kennlinie laufend wiederholt. Wenn der erste Punkt beim Zeitpunkt 0 liegt, wird er von den Wiederholungen ausgeschlossen, da der letzte Punkt Vorrang hat.

Dauert die Simulation länger als die Sequenz der gespeicherten Ausgangssignale und wird keine Wiederholung gewünscht (*Iterativ* nicht gesetzt), so wirkt der jeweils letzte Stützpunkt als konstantes Ausgangssignal über die restliche Simulationszeit.

Beschreibung des Datenformats

Die Datenreihen und die dazugehörige Zeitbasis bilden eine Tabelle. Als Spaltentrennzeichen sind Tab, Leerzeichen und Semikolon (;) zulässig (auch gemischt). Gültige Dezimalzeichen sind Punkte und Kommas. Die erste Spalte enthält die Zeitwerte, alle weiteren Spalten bilden je ein Ausgangssignal. Alle Ausgangswerte einer Zeile beziehen sich auf den Zeitwert in der ersten Spalte. Die Zeitpunkte sind beliebig (kein festes Intervall notwendig), sie müssen jedoch aufsteigend sortiert sein. Sprungstellen können durch zwei aufeinander folgende identische Zeitpunkte realisiert werden.

Eine Datenzeile kann mit einem Kommentar, eingeleitet mit einem Sternzeichen *, abgeschlossen werden. Kommentarzeilen, die mit einer Zahl beginnen (Leerzeichen unbeachtet) müssen mit einem Sternzeichen * beginnen (Leerzeichen davor erlaubt). Die Datei muss eine ANSI oder ASCII-Textdatei sein. Als Namenserverweiterung wird *.txt* vorgezogen, aber es kann auch jede andere gültige Windows Namenserverweiterung sein.

Beispiel Dateiquelle

Erzeugt durch Bruno Büsser, 30. November, 2002

Diese Datei enthält 3 Ausgangssignale mit je 5 Stützpunkten

-1; 0.5; 0.6; 0.8 *Optionaler fiktiver erster Punkt als Anfangsbedingung

2.45; 1; 5; 8 *Dies ist der erste Stützpunkt. Zeit zuerst, dann Ausgangswerte.

3.67; 2.5; 4.3; 6 *Zweiter Stützpunkt. Die Zeitwerte sind in aufsteigender Reihenfolge.

5 3.14 4.5 3.3 *Die Werte werden nun mit einem Leerzeichen getrennt

(>Leerzeilen werden übersprungen)

6.23 2,5 4.1; 2,5 *Dezimaltrennzeichen kann auch ein Komma sein

Zeilen, die nicht mit einer Ziffer, Punkt oder Komma beginnen werden übersprungen

8;0 3,8; 1.4 *Dies ist der letzte Stützpunkt

* 5 Stützpunkte wurden eingelesen >> * wird benötigt, da Zahl am Zeilenanfang

Ende der Beispieldatei

Die Datenreihen werden am besten mit einem Tabellenkalkulationsprogramm (z.B. Microsoft Excel) erstellt und danach als einfache Textdatei abgespeichert.

Achtung: Die Integrationsschrittweite sollte idealerweise so gewählt werden, dass die Integrationsschritte genau auf die Datenpunkte treffen. Wenn dies nicht möglich ist, darf sie nicht länger als die Hälfte des zeitlichen Abstandes zwischen zwei Datenpunkten sein. Bei Verletzung dieser Regel tritt eine Unterabtastung auf, was zu einer ungewollten Signalfilterung und schliesslich zur Verzerrung des Ausgangssignals führt. In Zweifelsfällen sollte eine optische Überprüfung der Ausgangssignale vorgenommen werden.

13.2 Lineare Glieder

Lineare Glieder erfüllen das Superpositions- und Verstärkungsprinzip.

$$f(u_1 + u_2) = f(u_1) + f(u_2) \quad \text{Superpositionsprinzip}$$

$$f(Ku_1) = Kf(u_1) \quad \text{Verstärkungsprinzip}$$

Mit linearen Gliedern können sowohl Zeit- als auch Frequenzsimulationen durchgeführt werden.

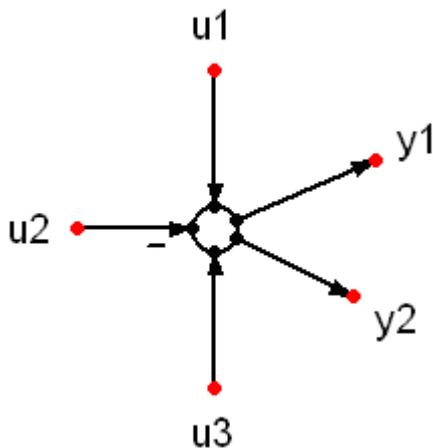
13.2.1 Summierglied (Addierer)

Das Summierglied ist ein Übertragungsglied mit mehreren Eingangs- und einer einzigen Ausgangsgrösse. Mit mehreren Signalleitungen am Eingang und einer Signalleitung am Ausgang bildet es eine Additionsstelle. Für jede Eingangsgrösse kann das Vorzeichen individuell gewählt werden. Mit nur einer negativen Eingangsgrösse kann es auch als Umpolglied verwendet werden.

Das Vorzeichen einer in das Summierglied einmündenden Signalleitung wird durch Markieren der Leitung und Drücken der + oder – Taste im numerischen Tastenfeld oder im Kontextmenü der Signalleitung durch *Vorzeichen ändern* verändert.

$$y(t) = \pm u_1(t) \pm u_2(t) \pm \dots \pm u_n(t)$$

Beispiel: $y_1 = u_1 - u_2 + u_3$
 $y_2 = u_1 - u_2 + u_3$



13.2.2 Proportionalglied, Verstärkungsglied (P-Glied)

Die Eingangsgrösse wird mit der Verstärkung K multipliziert.

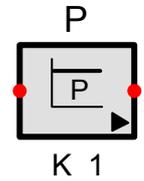
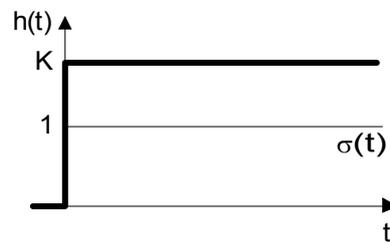
Funktionalbeziehungen

$$y(t) = Ku$$

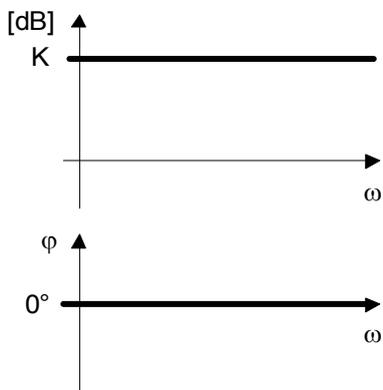
$$G(s) = K$$

Schrittantwort

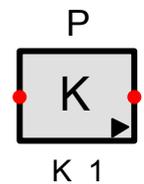
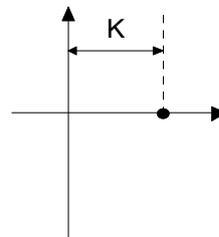
$$h(t) = K\sigma(t)$$



Amplituden- und Phasengang



Ortskurve



13.2.3 Integrator (I-Glied)

Beim Integrator ist die Ausgangsgrösse das Integral der Eingangsgrösse über die Zeit. Er reagiert auf einen konstanten Sprung mit einer zeitlich unbegrenzten rampenförmigen Ausgangsgrösse. Er entspricht somit einem Speicher der unbegrenzt voll läuft.

Für $t \leq 0$ ist die Ausgangsgrösse gleich dem Anfangswert Y_0

Bei der Schrittantwort entspricht die Ausgangsgrösse zur Integrationszeit T_i genau dem Wert der Eingangsgrösse (falls $Y_0 = 0$).

Die Ausgangsgrösse wird auf den Wertebereich $Y_{min} \dots Y_{max}$ begrenzt (Anti-Windup). Die Integration wird beim Erreichen eines dieser Maximalwerte angehalten und erst wieder in entgegengesetzter Richtung fortgesetzt, wenn die Eingangsgrösse das Vorzeichen geändert hat. Die Begrenzungen sind nur aktiv, wenn $Y_{max} > Y_{min}$.

Mit dem Reseteingang lässt sich der Integrator jederzeit auf den Anfangswert Y_0 zurücksetzen (auch wenn er nicht zwischen Y_{min} und Y_{max} liegt). Die logische Schaltschwelle V_{th} liegt standardmässig bei 0.5, kann jedoch individuell eingestellt werden.

In Frequenzsimulationen werden Reset- und Hold Eingang nicht beachtet und dürfen offen gelassen werden.

Bei der Zeitsimulationen dürfen diesen Eingänge, falls sie nicht benötigt werden, ebenfalls offen gelassen werden.

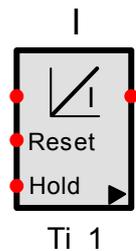
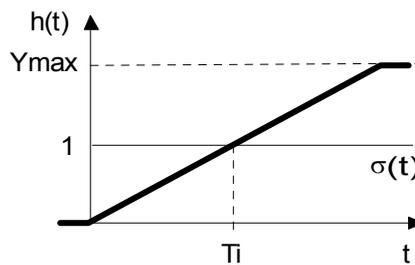
Funktionalbeziehungen

$$y(t) = \frac{1}{T_i} \int_0^t u(\tau) d\tau$$

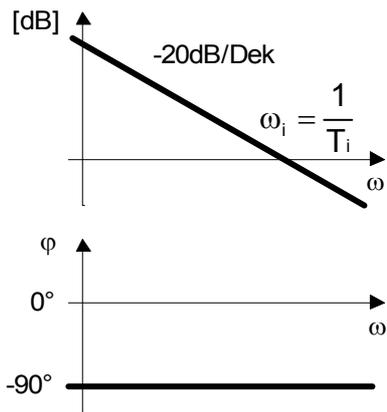
$$G(s) = \frac{1}{sT_i}$$

Schrittantwort

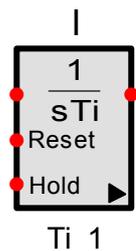
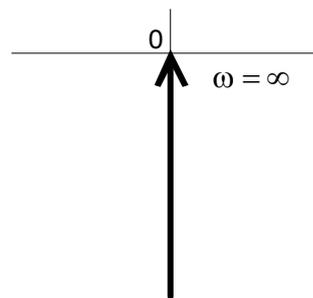
$$h(t) = \frac{1}{T_i} t$$



Amplituden- und Phasengang



Ortskurve



13.2.4 Differenzierer (D-Glied)

Der Differenzierer entspricht einer möglichst guten Annäherung an einen idealen Differenzierer. Bei der Zeitsimulation ist das Verhalten des Gliedes von der Integrationsschrittweite und bei der Frequenzsimulation von der Stoppfrequenz abhängig.

Ein idealer Differenzierer weist bei einem Sprungeingang am Ausgang ein unendlich hoher Stoss verschwindender Länge auf, der jedoch durch die Annäherung in der Höhe auf den Wert T_D/h begrenzt wird und h lang ist, wobei h die Integrationsschrittweite ist. Je kürzer die Integrationsschrittweite, desto besser ist also die Annäherung an das ideale Verhalten. Ein rein differentielles Verhalten kommt in einem realen System nicht vor, so dass die Annäherung i. A. keinen Nachteil bedeutet.

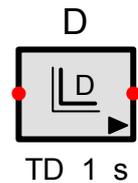
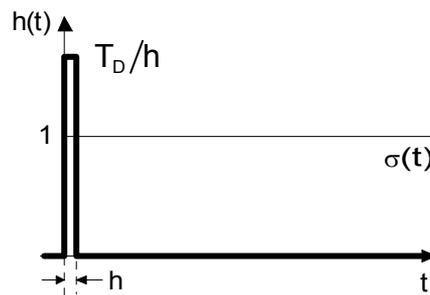
Funktionalbeziehungen

$$y(t) = T_D \dot{u}(t)$$

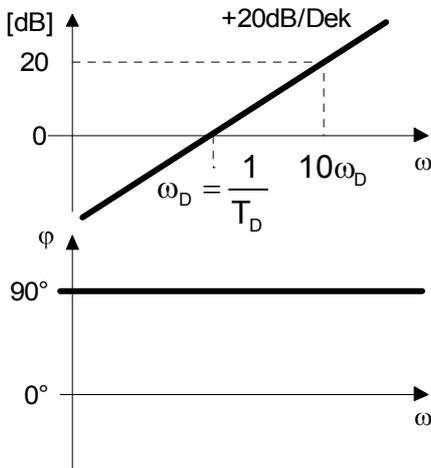
$$G(s) = T_D s$$

Schrittantwort

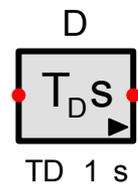
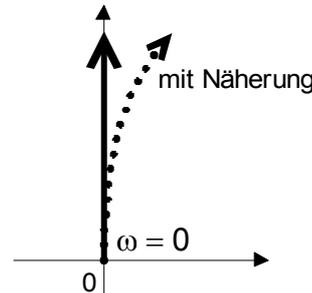
$$h(t) = T_D \delta(t)$$



Amplituden- und Phasengang



Ortskurve



Implementationshinweise

Da ein idealer Differenzierer auch programmtechnisch nicht realisiert werden kann, wird folgende Ersatzfunktion benutzt: $G(s) = \frac{T_D s}{1 + T s}$. Im Idealfall wäre $T = 0$.

Für die **Zeitsimulation** gilt: $T = h$ (Integrationsschrittweite)

Für die **Frequenzsimulation** gilt: $T = \frac{1}{100 \omega_s}$; ω_s = Simulations-Stoppfrequenz (durch Anwender wählbar)

oder $T = T_D / 100$, je nachdem welcher Wert kleiner ist.

13.2.5 Vorhalteglied (DT₁ Glied)

Das DT₁-Glied besteht aus der Kombination eines idealen Differenzierers mit einem Verzögerungsglied 1. Ordnung. Dadurch wird gegenüber dem Differenzierglied die Schrittantwort stark abgedämpft und das anschließende Einschwingverhalten deutlich verlängert.

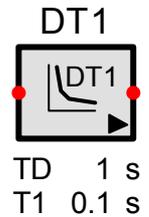
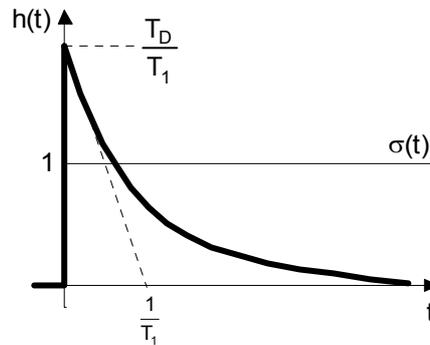
Funktionalbeziehungen

$$y + T_1 \dot{y} = T_D \dot{u}$$

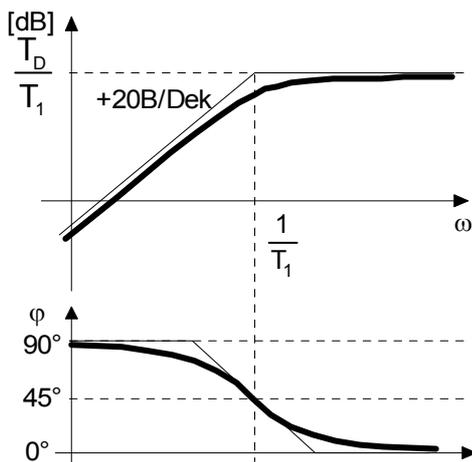
$$G(s) = \frac{T_D s}{1 + T_1 s}$$

Schrittantwort

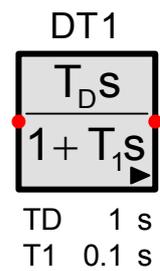
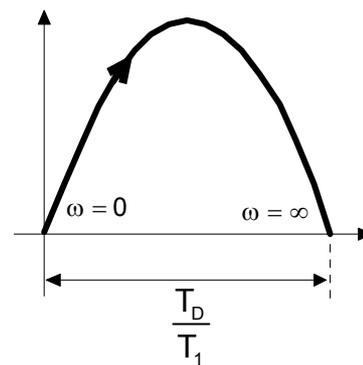
$$h(t) = \frac{T_D}{T_1} e^{-\frac{t}{T_1}}$$



Amplituden- und Phasengang



Ortskurve



13.2.6 Verzögerungsglied 1.Ordnung (PT₁-Glied)

Das PT₁-Glied besitzt einen internen Energiespeicher und folgt deshalb einem Sprung an seinem Eingang nur verzögert. Scharfe Ecken und steile Flanke werden stark geglättet. Die Dämpfung ist so stark, dass kein Überschwingen auftreten kann. Bei der Schrittanregung folgt die Ausgangsgrösse nach längerer Zeit proportional der Eingangsgrösse.

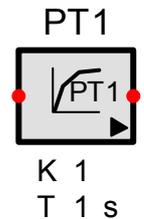
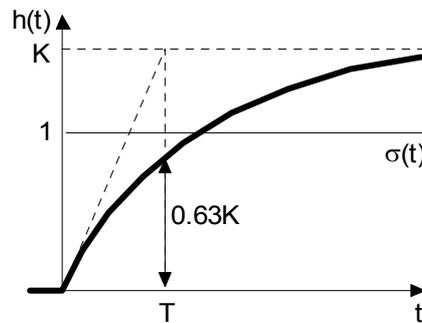
Funktionalbeziehungen

$$T\dot{y} + y = Ku$$

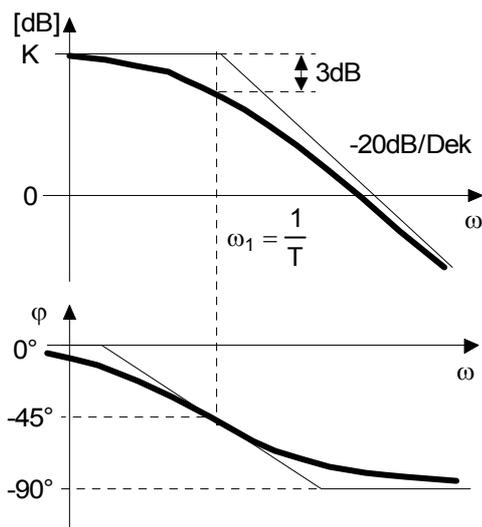
$$G(s) = \frac{K}{1 + Ts}$$

Schrittantwort

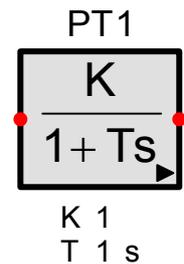
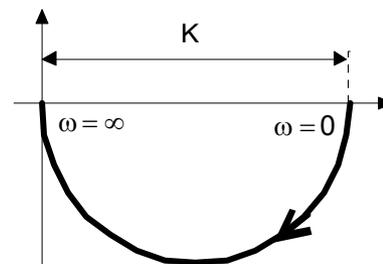
$$h(t) = K(1 - e^{-t/T})$$



Amplituden- und Phasengang



Ortskurve



13.2.7 Verzögerungsglied 2.Ordnung (PT₂-Glied)

Das PT₂-Glied enthält zwei voneinander unabhängige Energiespeicher. In Abhängigkeit der Dämpfung d lassen sich vier Fälle unterscheiden:

1. **Periodischer Fall ohne Dämpfung:** $d=0$
2. **Gedämpfter periodischer Fall:** $0 < d < 1$
3. **Aperiodischer Grenzfall:** $d=1$
4. **Aperiodischer Fall:** $d > 1$

Funktionalbeziehungen

$$T^2\ddot{y} + 2dT\dot{y} + y = Ku$$

Schrittantwort

$$h(t) = K - \frac{K}{\sqrt{1-d^2}} e^{-(d/T)t} \sin \left[\frac{\sqrt{1-d^2}}{T} t + \arctan \left(\frac{\sqrt{1-d^2}}{d} \right) \right]$$

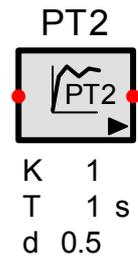
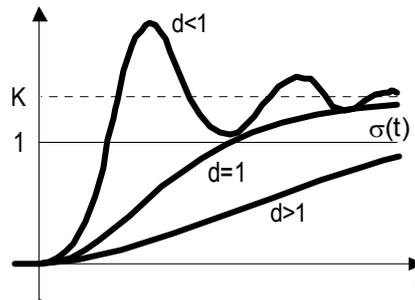
$$G(s) = \frac{K}{1 + 2dT s + T^2 s^2}$$

Für $d \leq 1$ gilt auch:

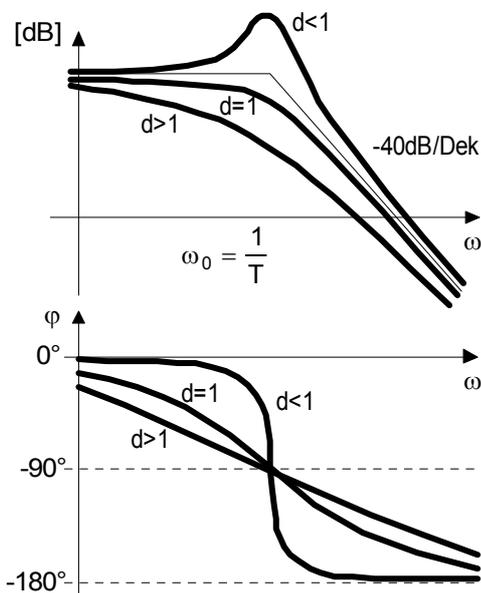
$$G(s) = \frac{K}{(1 + T_1 s)(1 + T_2 s)}$$

$$T_1 = T(d + \sqrt{d^2 - 1})$$

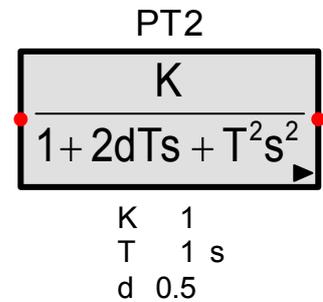
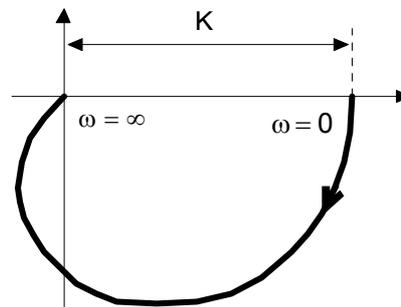
$$T_2 = T(d - \sqrt{d^2 - 1})$$



Amplituden- und Phasengang



Ortskurve



13.2.8 Nicht schwingfähiges Verzögerungsglied 2. Ordnung (PT₁T₂-Glied)

Das PT₁T₂-Glied entspricht funktionell dem PT₂-Glied, wobei die Dämpfung aber immer grösser oder gleich 1 ist. Es liegt also der aperiodische Fall vor. In diesem Fall kann der Nenner der Übertragungsfunktion in die Linearfaktoren mit den beiden Zeitkonstanten T₁ und T₂ zerlegt werden.

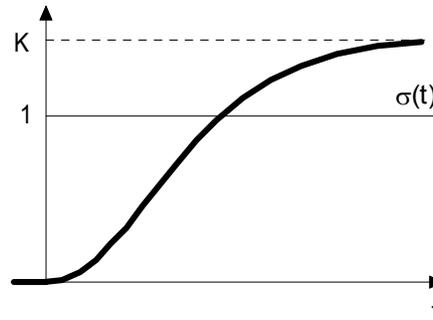
Funktionalbeziehungen

$$y + \dot{y}(T_1 + T_2) + \ddot{y}T_1T_2 = Ku$$

$$G(s) = \frac{K}{(1 + sT_1)(1 + sT_2)}$$

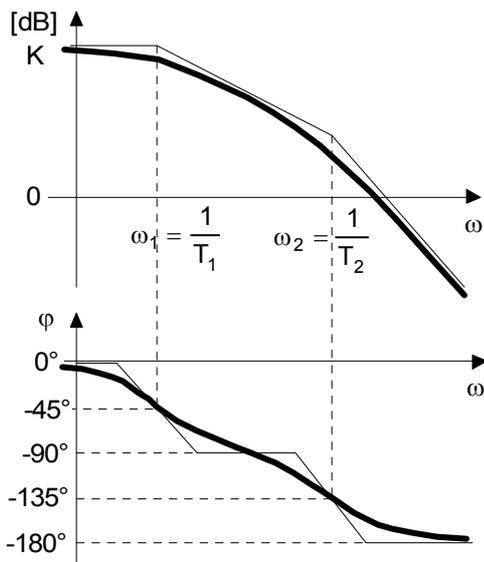
Schrittantwort

$$h(t) = K \left(1 - \frac{T_1 e^{-t/T_1} - T_2 e^{-t/T_2}}{T_1 - T_2} \right)$$

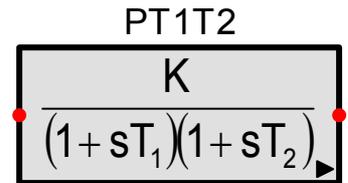
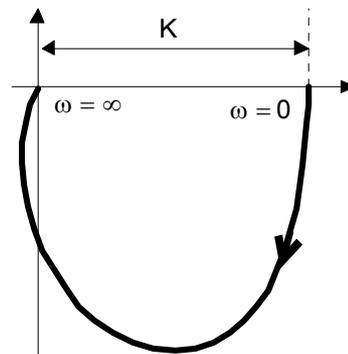


K 1
T1 2 s
T2 1 s

Amplituden- und Phasengang



Ortskurve



K 1
T1 2 s
T2 1 s

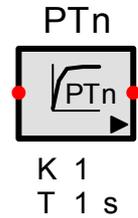
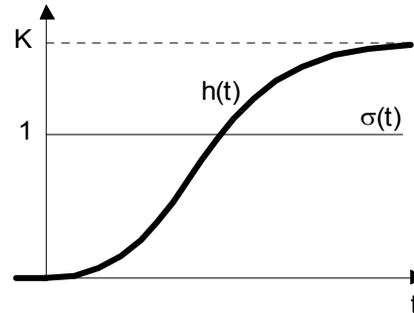
13.2.9 Verzögerungsglied n-ter Ordnung (PT_n-Glied)

Dieses Glied weist neben der Verstärkung K n identische Zeitkonstanten auf. Je grösser n, desto flacher wird die Schrittantwort.

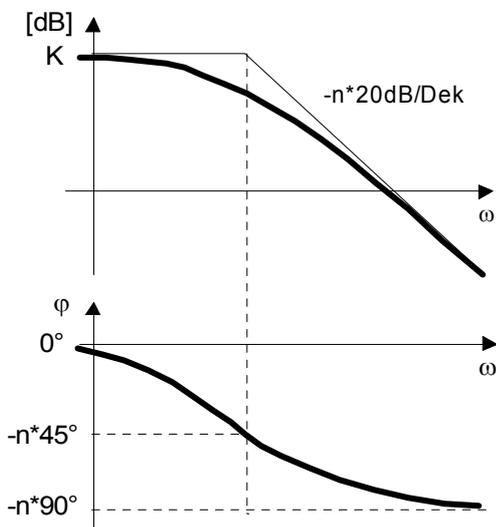
Funktionalbeziehungen

$$G(s) = \frac{K}{(1+Ts)^n}; \quad n \geq 0$$

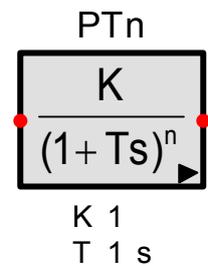
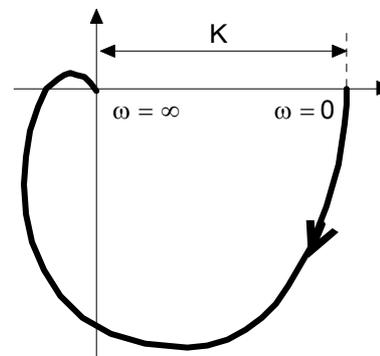
Schrittantwort



Amplituden- und Phasengang



Ortskurve



13.2.10 Lead/Lag-Glied

Das Lead/Lag-Glied ist ein rationales Glied 1. Ordnung, das je nach Verhältnis von T1 zu T2 als voreilendes oder nacheilendes Glied konfiguriert werden kann.

Nicht zu verwechseln ist dieses Glied mit dem Lead/Lag Regler, der eine Übertragungsfunktion 2-ter Ordnung aufweist.

Funktionalbeziehungen

$$y + T_2 \dot{y} = K(u + T_1 \dot{u})$$

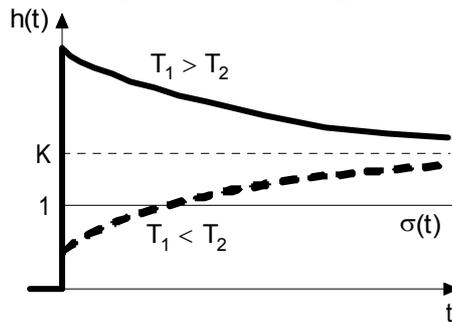
$$G(s) = K \frac{1 + T_1 s}{1 + T_2 s}$$

Voreilend: $T_1 > T_2$

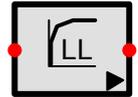
Nacheilend: $T_1 < T_2$

Schrittantwort

$$h(t) = K \left[1 + (T_1 - T_2) e^{-\frac{t}{T_2}} \right]$$

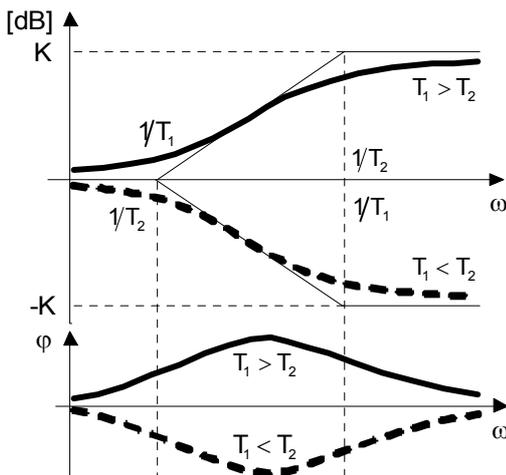


Lead/Lag

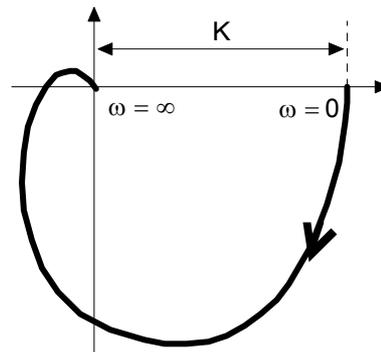


K 1
T1 0.5 s
T2 1 s

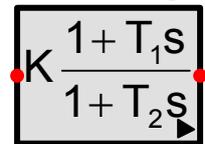
Amplituden- und Phasengang



Ortskurve



Lead/Lag



K 1
T1 0.5 s
T2 1 s

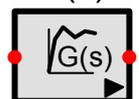
13.2.11 Rationales Übertragungsglied (G(s)-Glied)

Das G(s)-Glied ist die allgemeine Form eines linearen Übertragungsgliedes $Y(s) = G(s) U(s)$ mit G(s) als rationale Funktion.

Übertragungsfunktion:

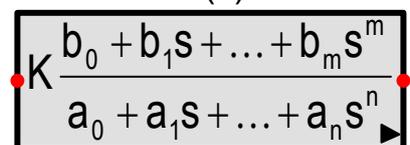
$$G(s) = K \frac{b_0 + b_1 s + \dots + b_m s^m}{a_0 + a_1 s + \dots + a_n s^n}; \quad a_n \neq 0 \text{ und } m \leq n$$

G(s)



b0: 1 a0: 1

G(s)



b0: 1 a0: 1

P-, I-, PT₁-, PT₂- und PT₁T₂-Glieder etc. sind Spezialfälle dieses Gliedes. Da der Zählergrad m nicht grösser als der Nennergrad n sein darf, ist kein Differenzierverhalten möglich. Amplituden- und Phasengang, Ortskurve und Sprungantwort sind abhängig von den Ordnungen der Polynome und der Wahl der Koeffizienten.

13.2.12 Totzeitglied (PTt-Glied)

Beim Totzeitglied tritt der zeitliche Verlauf des Eingangssignals um die Totzeit T_t verspätet am Ausgang auf. Es beeinflusst somit das Laufzeitverhalten von Signalen, wobei die Form der Signale nicht verändert wird und ohne Bedeutung ist.

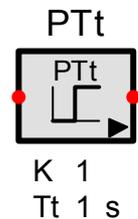
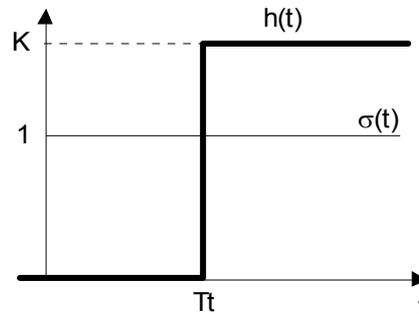
Funktionalbeziehungen

$$y(t) = Ku(t - T_t)$$

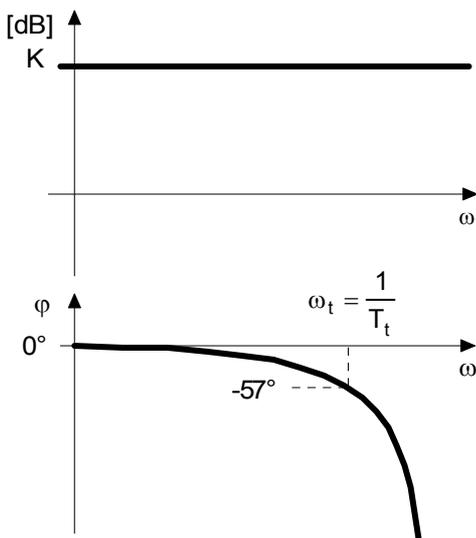
$$G(s) = Ke^{-sT_t}$$

Schrittantwort

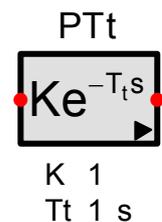
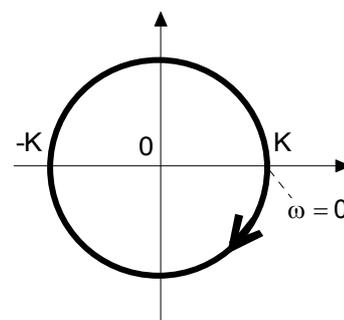
$$h(t) = K\sigma(t - T_t)$$



Amplituden- und Phasengang



Ortskurve



Padé-Ersatzschaltungen

Die Übertragungsfunktion des Totzeitgliedes hat weder ein Zähler- noch ein Nennerpolynom. Da keine Pole und Nullstellen definiert sind, können auch keine Eigenwerte berechnet werden. Die Eigenwerte eines Systems werden nicht berechnet, wenn Totzeitglieder ohne Approximation vorkommen.

Um diesem Umstand zu begegnen, muss das Totzeitglied durch Zustandsvariablen beschrieben werden. Da das ideale Totzeitglied aber nicht durch endlich viele Zustandsvariablen beschrieben werden kann, muss es in geeigneter Weise approximiert werden.

SimApp bietet zwei Approximationsmethoden nach H.E. Padé an[1].

Padé-Allpass

Die rationale Übertragungsfunktion des Padé-Allpasses lautet:

$$G(T_t s) = \frac{1 + \sum_{i=1}^n (-1)^i a_i T_t^i s^i}{1 + \sum_{i=1}^n a_i T_t^i s^i} \quad \text{wobei} \quad a_i = \binom{n}{i} \frac{1}{2n(2n-1)\dots(2n-i+1)} \quad i = 1, \dots, n$$

Der Amplitudengang ist korrekt. Der Phasengang weist bis $10\omega_t$ eine gute Übereinstimmung mit dem idealen Glied auf. Ungünstig ist das Totzeitverhalten jedoch bei der Stossantwort. Der Anfangswert ist je nach Ordnung gleich 1 oder -1, obwohl er eigentlich 0 sein sollte. Anschließend folgt eine starke Oszillation.

Um eine bessere Sprungantwort zu erreichen, bietet sich die zweite Approximation an:

Padé-Approximation

Die rationale Übertragungsfunktion der Padé-Approximation lautet:

$$G(T_t, s) = \frac{1 + \sum_{i=1}^{n-1} b_i T_t^i s^i}{1 + \sum_{i=1}^n a_i T_t^i s^i} \quad \text{wobei} \quad a_i = \binom{n}{i} \frac{1}{(2n-1)(2n-2)\dots(2n-i)} \quad i = 1, \dots, n$$

$$b_i = (-1)^i \binom{n-1}{i} \frac{1}{(2n-1)(2n-2)\dots(2n-i)} \quad i = 1, \dots, n-1$$

Die Padé-Approximation hat nicht mehr den idealen Allpasscharakter des Padé-Allpasses. Das Stossverhalten ist jedoch wesentlich besser. Der Anfangswert ist jetzt korrekt gleich 0.

Welche Methode sollten Sie verwenden?

Wenn Sie eine Zeitsimulation durchführen, sollten Sie im Allgemeinen das ideale Totzeitverhalten, also ohne Approximation, auswählen. Sprünge werden dann absolut korrekt wiedergegeben. Wegen des Bufferspeichers wird der Speicherbedarf bei grossen Totzeiten jedoch sehr hoch. In gewissen Fällen könnte deshalb eine Approximation Vorteile bringen. Wenn Sie sich für eine Approximation entscheiden und das Totzeitglied nahe beim Eingang des Systems liegt, wo noch effektive Sprünge auftreten können, sollten Sie wegen der besseren Stossantwort auf jeden Fall die Padé-Approximation wählen.

Bei der Frequenzsimulation liefert das ideale Totzeitglied für den Amplituden- und Phasengang korrekte Werte. Die Eigenwerte werden jedoch nicht berechnet. Für korrekte Eigenwerte müssen Sie deshalb eine Approximation nach Padé auswählen, wobei der Padé-Allpass das bessere Frequenzverhalten aufweist. Beachten Sie jedoch, dass die Pole und Nullstellen des Padé-Allpasses ins Gesamtsystem einfließen und somit die Eigenwerte in Anzahl und Betrag beeinflussen.

13.2.13 Allpassglied 1. Ordnung (PTa1-Glied)

Das PTa1-Glied weist einen konstanten von der Frequenz unabhängigen Amplitudengang auf. Seine Sprungantwort zeichnet sich dadurch aus, dass die Ausgangsgröße zuerst in die Gegenrichtung springt. Allpass behaftete Regelstrecken sind deshalb schwierig auszuregeln.

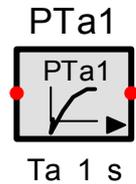
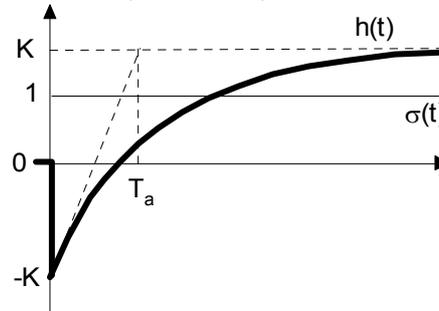
Funktionalbeziehungen

$$y + T_a \dot{y} = K(u - T_a \dot{u})$$

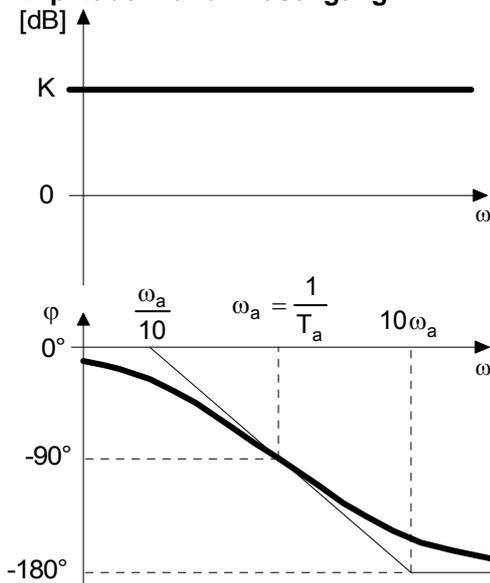
$$G(s) = K \frac{1 - sT_a}{1 + sT_a}$$

Schrittantwort

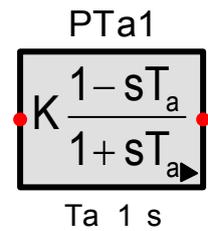
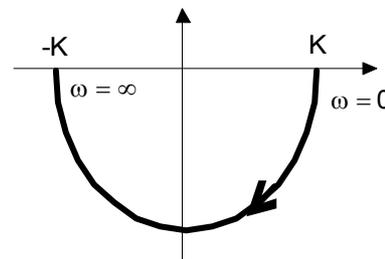
$$h(t) = K(1 - 2e^{-t/T_a})$$



Amplituden- und Phasengang



Ortskurve



13.2.14 Allpassglied 2. Ordnung (PTa2-Glied)

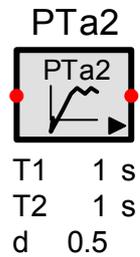
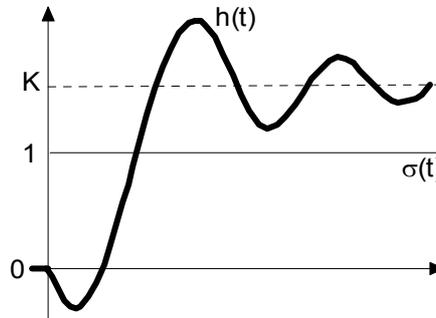
Gegenüber dem Allpass 1. Ordnung, weist dieses Glied ebenfalls eine Schrittantwort auf, die zuerst in die Gegenrichtung weist, wobei aber die Schwingungskomponente im Nenner der Übertragungsfunktion im zweiten Teil der Schrittantwort für eine abklingende Schwingung sorgt (wie beim PT₂-Glied).

Funktionalbeziehungen

$$y + 2dT_2\dot{y} + T_2^2\ddot{y} = K(u - T_1\dot{u})$$

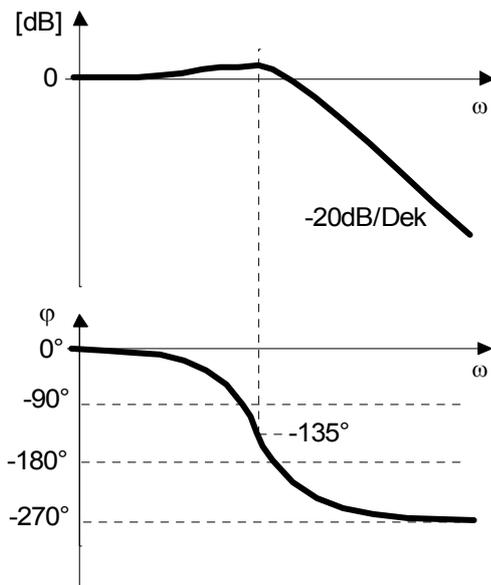
$$G(s) = K \frac{1 - T_1s}{1 + 2dT_2s + T_2^2s^2}$$

Schrittantwort

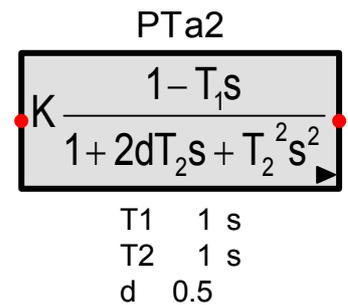
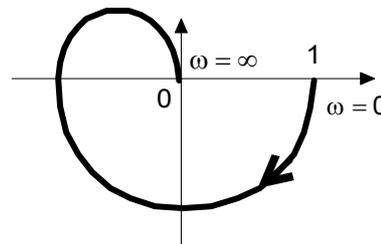


Amplituden- und Phasengang

Bsp. für T1=1, T2=1; d=0.5



Ortskurve



13.2.15 Lineares Differentialgleichungssystem

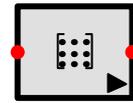
Frei parametrierbares, lineares und zeitinvariantes Differentialgleichungssystem im Zustandsraum.

Vektorform

$$\text{Systemgleichung: } \dot{\underline{x}}(t) = \underline{A}\underline{x}(t) + \underline{B}\underline{u}(t)$$

$$\text{Ausgangsgleichung: } \underline{y}(t) = \underline{C}\underline{x}(t) + \underline{D}\underline{u}(t)$$

LDGS



Übertragungsfunktion

$$G(s) = C(s)[sI - A(s)]^{-1}B(s) + D(s) \quad s = j\omega$$

Das System hat n Zuständen, p Eingangs- und q Ausgangsgrößen

- \underline{x} Zustandsvektor (nx1)
- \underline{u} Eingangsvektor (px1)
- \underline{y} Ausgangsvektor (qx1)
- A Systemmatrix (nxn)
- B Eingangsmatrix (nxp)
- C Ausgangsmatrix (qxn)
- D Durchgangsmatrix (qxp)

LDGS

$$\begin{aligned} \dot{\underline{x}} &= \underline{A}\underline{x} + \underline{B}\underline{u} \\ \underline{y} &= \underline{C}\underline{x} + \underline{D}\underline{u} \end{aligned}$$

Parameter $n \leq 50, p \leq 50, q \leq 50$

Matrizendarstellung:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & \cdot & A_{1n} \\ A_{21} & A_{22} & \cdot & A_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ A_{n1} & A_{n2} & \cdot & A_{nn} \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ x_n \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} & \cdot & B_{1p} \\ B_{21} & B_{22} & \cdot & B_{2p} \\ \cdot & \cdot & \cdot & \cdot \\ B_{n1} & B_{n2} & \cdot & B_{np} \end{bmatrix} * \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ u_p \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ y_q \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & \cdot & C_{1n} \\ C_{21} & C_{22} & \cdot & C_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ C_{q1} & C_{q2} & \cdot & C_{qn} \end{bmatrix} * \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ x_n \end{bmatrix} + \begin{bmatrix} D_{11} & D_{12} & \cdot & D_{1p} \\ D_{21} & D_{22} & \cdot & D_{2p} \\ \cdot & \cdot & \cdot & \cdot \\ D_{q1} & D_{q2} & \cdot & D_{qp} \end{bmatrix} * \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ u_p \end{bmatrix}$$

In realen Systemen ist meistens $D = 0$.

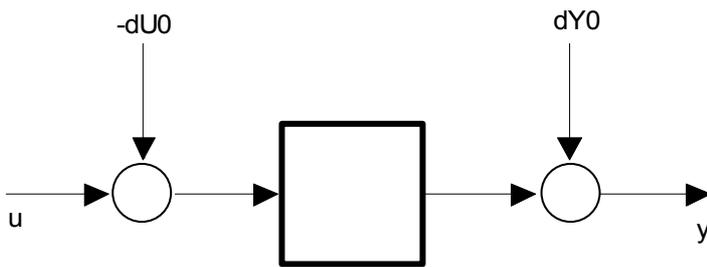
13.3 Nichtlineare Glieder

Für Zeitsimulationen können lineare und nichtlineare, also alle Glieder benutzt werden. Für die Frequenzsimulationen sind jedoch nur die linearen (inkl. zeitdiskreten) Glieder zulässig.

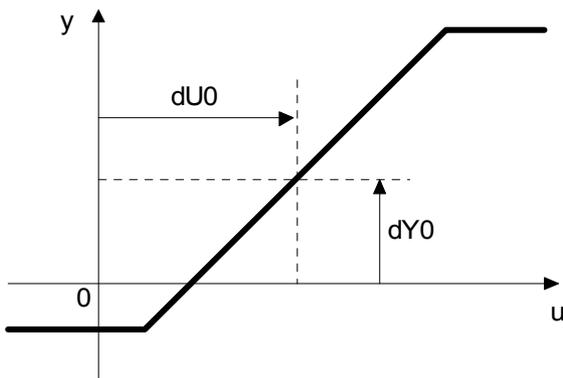
Für die wichtigsten nichtlinearen Funktionen stehen eigenständige Glieder zur Verfügung, andere können mit dem Kennlinienglied oder durch eine Kombination von Standardgliedern realisiert werden.

Eingangs- und Ausgangsverschiebung

Die Kennlinien der meisten nichtlinearen Glieder können innerhalb ihres Kennlinienfeldes verschoben werden. Durch eine konstante Subtraktion des Eingangssignals wird die Kennlinie nach rechts und durch eine Addition des Ausgangssignals nach oben verschoben. Die Werte für die Eingangs- und Ausgangsverschiebung werden in SimApp standardmässig nicht in der Zeichnung angezeigt. Sie werden im Dialogfeld der Simulationseigenschaften der einzelnen Glieder festgelegt.

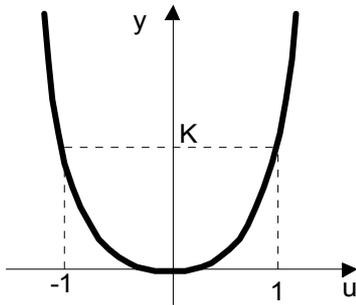


Beispiel: Die Sättigungskennlinie wird um $dU0$ und $dY0$ verschoben:

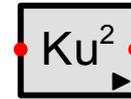


13.3.1 Quadrarglied

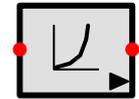
Funktionalbeziehung: $y = Ku^2$



Quadrat

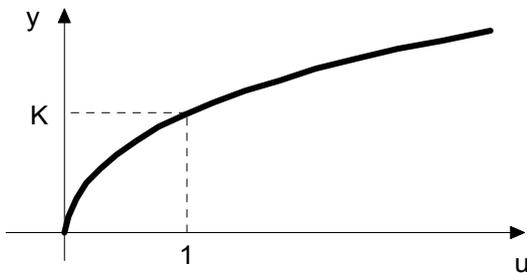


Quadrat

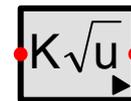


13.3.2 Radizierglied (Wurzel)

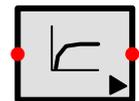
Funktionalbeziehung: $y = K\sqrt{u}$



Wurzel

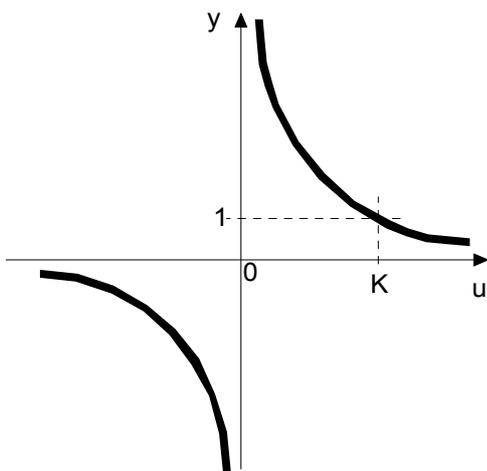


Wurzel

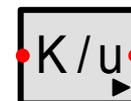


13.3.3 Inverter

Funktionalbeziehung: $y = K/u$



Inverter



Inverter



13.3.4 Multiplizierglied (Produkt)

Das Multiplizierglied bildet das Produkt zweier Eingangsgrößen und der Verstärkung K.

Funktionalbeziehung: $y = Ku_1u_2$

Produkt



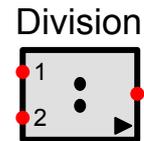
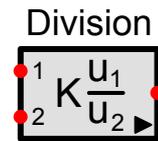
Produkt



13.3.5 Dividierglied

Das Dividierglied dividiert die Eingangsgrösse u_1 durch die Eingangsgrösse u_2 und multipliziert das Resultat mit der Verstärkung K .

Funktionalbeziehung: $y = K \frac{u_1}{u_2}$



13.3.6 Verknüpfungsglied

Dieses Glied ermöglicht das Verknüpfen von mehreren Eingangssignalen mit den vier Grundrechenoperationen: Addition, Subtraktion, Multiplikation und Division.

Die Anzahl der Eingangsknoten ist zwischen 2 und 50 wählbar. Eingänge, die nicht benutzt werden resp. offen sind, werden während der Simulation nicht beachtet.

Das Vorzeichen kann im Kontextmenu der Signalleitung mit dem Befehl *Vorzeichen ändern* oder durch Drücken der Minus- oder Plustaste im numerischen Tastenfeld gekehrt werden.



Addition und Subtraktion

Funktionalbeziehung: $y = \pm u_1 \pm u_2 \pm \dots \pm u_n$

Die Subtraktion einzelner Eingänge erfolgt durch die Inversion der Eingangssignale in den Signalleitungen.

Multiplikation

Funktionalbeziehung: $y = \pm u_1 \times \pm u_2 \dots \times \pm u_n$

Die negative Gewichtung (Minuszeichen) von Faktoren erfolgt durch Inversion der Eingangssignale in den Signalleitungen.

Division

Funktionalbeziehung: $y = \pm u_1 \div \pm u_2 \dots \div \pm u_n$

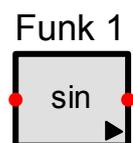
Der Dividend wird durch den ersten (obersten) Eingang gebildet. Alle anderen Eingänge sind Divisoren.

Das Vorzeichen des Dividenden und der Divisoren wird wie oben durch das Vorzeichen der Signalleitungen bestimmt.

13.3.7 Funktionsglied mit einem Eingang

Für nichtlineare Funktionen mit einer Eingangsgrösse existieren nur die wichtigsten Glieder (Quadrierglied, Radizierglied, etc.).

Mit dem Funktionsglied können weitere 29 Funktionen ausgewählt werden, wobei auch einige wenige lineare Funktionen darunter sind.



Trigonometrische und Arcus-Funktionen

sin	cos	tan	cotan	arcsin	arccos	arctan
-----	-----	-----	-------	--------	--------	--------

Hyperbolische und Area-Funktionen

cosh	sinh	tanh	arcosh	arsinh	artanh
------	------	------	--------	--------	--------

Exponential- und Logarithmusfunktionen

exp	2x	10x	x2	x3	ln	Lb	lg	xA	Ax	logA	\sqrt{x}
-----	----	-----	----	----	----	----	----	----	----	------	------------

Diverse Funktionen

x	sign (Vorzeichen)	Deg > Rad	Rad > Deg	Cycle > Rad	Rad > Cycle
---	-------------------	-----------	-----------	-------------	-------------

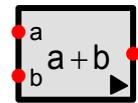
Die linearen Funktionen können auch für Frequenzsimulationen benutzt werden.

13.3.8 Funktionsglied mit zwei Eingängen

Dieses Glied hat zwei Eingangsgrößen mit folgenden Funktionen:

$a + b$	$a - b$	$a \times b$	a / b
a^b	$a^{1/b}$	$\sqrt{a^2 + b^2}$	
$\arctan(a/b)$	$\min(a,b)$	$\max(a,b)$	

Funk 2

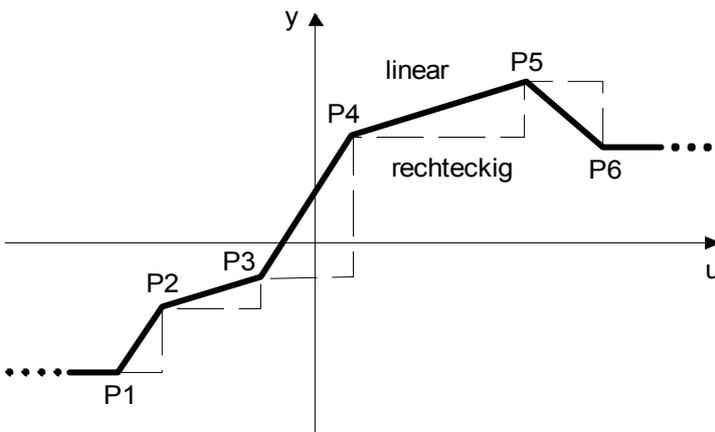


a ist der obere und b der untere Eingangsknoten.

13.3.9 Kennlinienglied (KL-Glied)

Beim Kennlinienglied kann eine beliebige statische Übertragungsfunktion mittels einer begrenzten Anzahl von Stützpunkten vorgegeben werden. Die Punkte werden linear oder rechteckig interpoliert.

Funktionalbeziehung: $y = f(P1, P2, \dots, Pn, t); n = 1..10'000$



Kennlinie



Sprünge werden realisiert, indem für zwei aufeinander folgende Stützstellen der gleiche Eingangswert u zugewiesen wird.

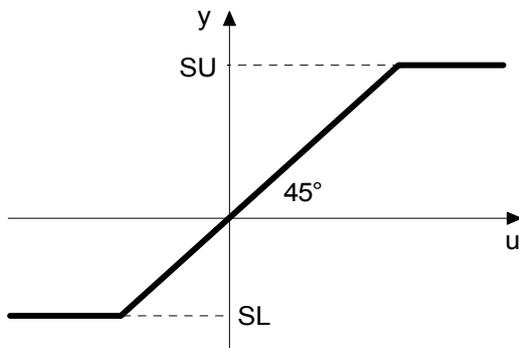
Die y-Werte des ersten und letzten Punktes gelten auch über die Kennlinie hinaus.

Die Datenwerte werden innerhalb des Gliedes gespeichert, können aber auch in eine Textdatei gespeichert oder daraus geladen werden. Dies ermöglicht den Datenaustausch mit anderen Applikationen.

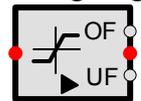
Für das Datenformat in der Textdatei siehe *Programmierbare Quelle* wobei hier aber kein fiktiver Punkt für die Anfangsbedingung zulässig ist.

13.3.10 Sättigungsglied

Das Sättigungsglied begrenzt den Betrag der Eingangsgröße auf die Werte SU und SL.



Sättigung

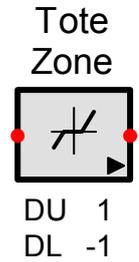
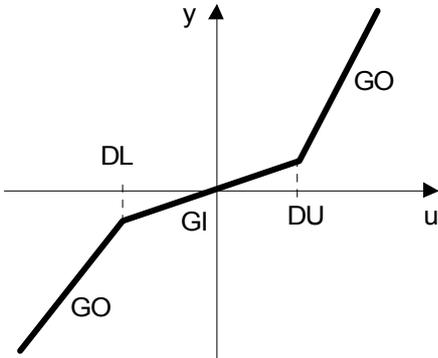


SU 1
SL -1

Wenn das Eingangssignal größer als SU ist, wird der Overflowausgang (OF) auf logisch 1 gesetzt und wenn es kleiner als SL ist, wird der Underflow (UF) Ausgang auf logisch 1 gesetzt. Das analoge Ausgangssignal bleibt jedoch immer zwischen den Begrenzungen.

13.3.11 Ansprechschwelle (Tote Zone)

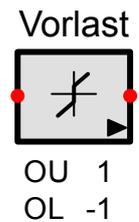
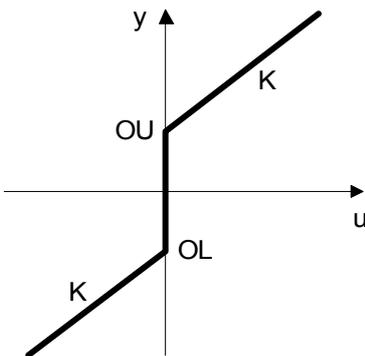
Die Ansprechschwelle unterdrückt innerhalb der toten Zone die kleinen Signale. Innerhalb DU und DL ist die Verstärkung GI, sonst GO. Standardmässig ist GI = 0; der Ausgang bleibt also null, bis das Eingangssignal eine der beiden Schwellen DL oder DU betragsmässig überschritten hat.



13.3.12 Vorlast (Offset)

Die Vorlast addiert einen konstanten positiven oder negativen Offset zur Eingangsgrösse.

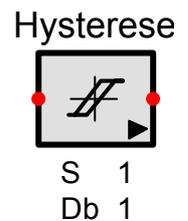
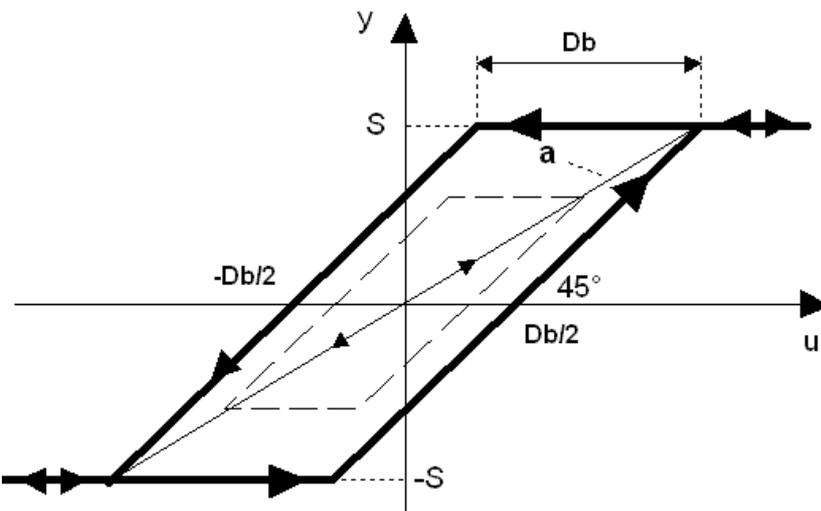
Funktionalbeziehung: $u \geq 0: y = OU + Ku$
 $u < 0: y = OL + Ku$



13.3.13 Hysterese

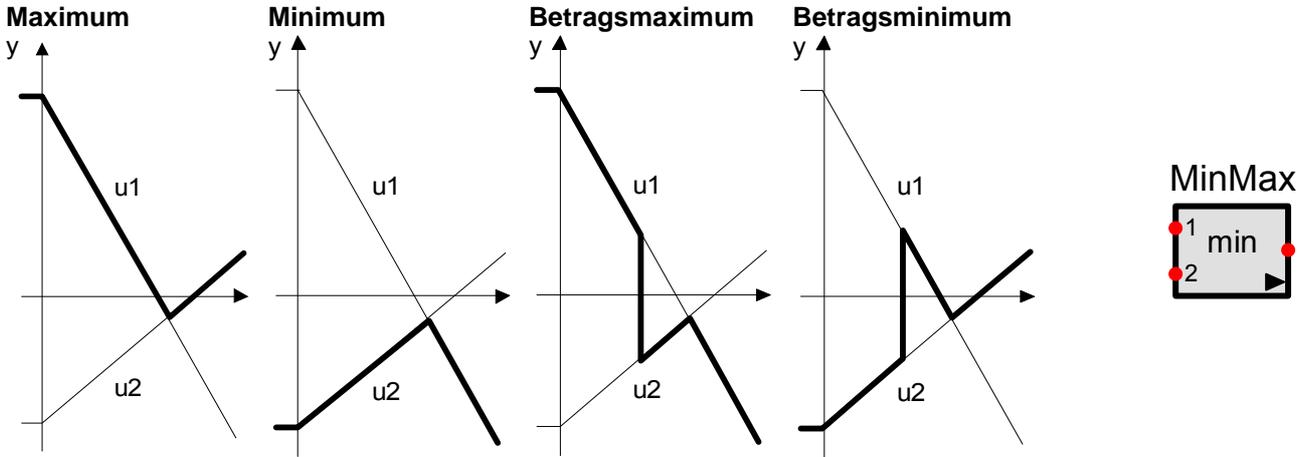
S entspricht dem maximalen Sättigungswert. Ist das Eingangssignal so klein, dass S nicht oder nicht mehr erreicht wird, so wird eine entsprechend kleinere Hystereseschleife gefahren. Wenn die Eingangsgrösse von der Nulllage aus startet, wird die Neukurve a befahren.

Parameter: S, C, dU0, dY0



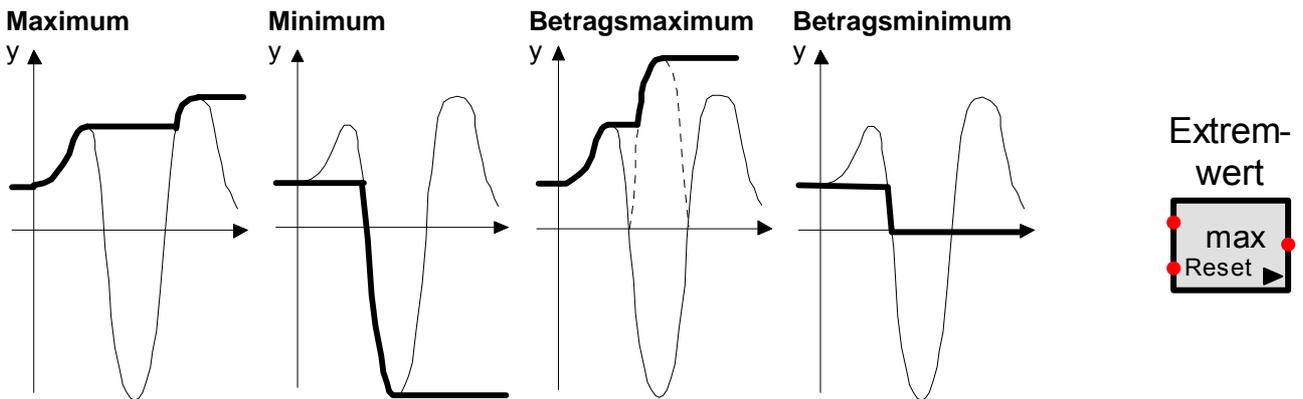
13.3.14 Minimum/Maximum (MinMax)

Dieses Glied mit 2 bis 50 Eingängen schaltet entweder das momentan kleinste oder grösste Eingangssignal zum Ausgang durch. Es können auch die Beträge zur Minimums- und Maximumsbildung herangezogen werden.



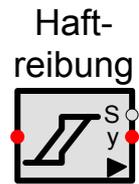
13.3.15 Extremwert

Dieses Glied folgt dem Eingangssignal und hält am Ausgang den grössten oder kleinsten Wert, der seit dem Start der Simulation oder seit dem letzten Zurücksetzen (Reset) aufgetreten ist. Mit der positiven Flanke des Resetsignals wird der Ausgang auf den momentanen Eingangswert zurückgesetzt. Solange der Reseteingang gesetzt ist, folgt der Ausgang exakt dem Eingang.



13.3.16 Haftreibung

Das Auftreten von Haftreibung kann in Regelkreisen leicht zu Schwingungen führen. Dieser Effekt ist zum Beispiel in Ventilen zu beobachten, wenn die Stellgrösse kurzzeitig angehalten oder die Bewegungsrichtung geändert wird. Wenn das Ventil in Ruhe ist, muss für eine weitere Bewegung zuerst die Haftreibung überwunden werden, die grösser als die Gleitreibung ist. Dies führt dazu, dass die Ventilcharakteristik nicht bloss einer einfachen Hysteresekurve entspricht, sondern dass zusätzlich nach jedem Stopp oder nach jeder Richtungsänderung ein Sprung auftritt. Im Englischen spricht man von "valve stiction", wobei stiction aus den beiden Wörtern "stick" (festsitzen, haften) oder "static" (statisch) und "friction" (Reibung) entstanden ist.



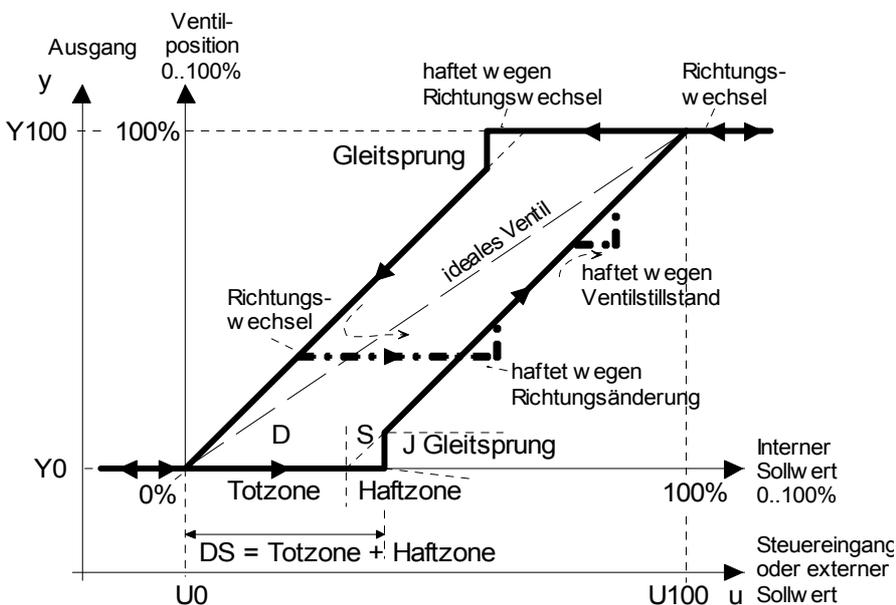
DS	3 %
J	1 %
U0	0
U100	10
Y0	0
Y100	10

Die Haftreibung in SimApp ist jedoch nicht auf Ventile beschränkt. Das Glied kann überall dort als spezielle Stelleigenschaft eingesetzt werden, wo Haftreibung auftritt. Andere geräteabhängige Effekte, die zusätzlich noch auftreten, wie z.B. die Nichtlinearität eines Ventils, können durch nachfolgende Glieder berücksichtigt werden.

SimApp setzt ein empirisches Modell von S. L. Shah (Modelling Valve Stiction) ein, das nicht auf physikalische Grundlagen basiert, aber sehr einfach einzusetzen ist und deren Parameter durch das Bedienungspersonal einfach ermittelt werden können.

Im Modell tritt die Haftreibung auf, wenn die Bewegungsrichtung geändert wird, oder das Ventil in mindestens zwei aufeinander folgenden Integrationsschritten zur Ruhe kommt. In der Praxis wird dieser Effekt in Prozenten des vollständigen Stellbereichs des Ventils ausgedrückt. Im vorliegenden SimApp-Glied existiert jedoch kein expliziter Eingabewert für die Haftreibung. In der Realität jedoch, wo die Totzone (Deadband $\leq 3\%$), Haftzone (Stickband $\leq 1\%$ und der Gleitsprung (Slip jump $= 1\%$) sehr klein sind, sind die Haftzone und der Gleitsprung praktisch identisch (auf ca. 0.02% genau). Wenn sie also einen speziellen Haftreibungswert haben, können Sie diesen auch für den Gleitsprung verwenden.

Für weitere Informationen über die Haftreibung bei Ventilen verweisen wir Sie auf die einschlägige Literatur von Shah, Ruel und anderen (Schlüsselwörter zum Suchen: valve, stiction, Shah, Ruel)



Parameter:

- **DS [%], Totzone(Deadband) + Haftzone(Stickband):** Bereich 0..100%
- **J [%], Gleitsprung(Slip jump):** Die sprunghafte Bewegung in %, wenn die kumulierten Eingangssignaländerungen grösser als die Haftzone (Stickband) werden (0..100).
- **U0:** Eingangssignal, das dem geschlossenen (0%) Ventil entspricht (0%)
- **U100:** Eingangssignal bei offenem (100%) Ventil
- **Y0:** Ausgangssignal, wenn das Ventil geschlossen ist

- **Y100:** Ausgangssignal bei offenem Ventil. Kann die Dimensionen [mm,m], Durchfluss [m^3/s] oder andere haben.

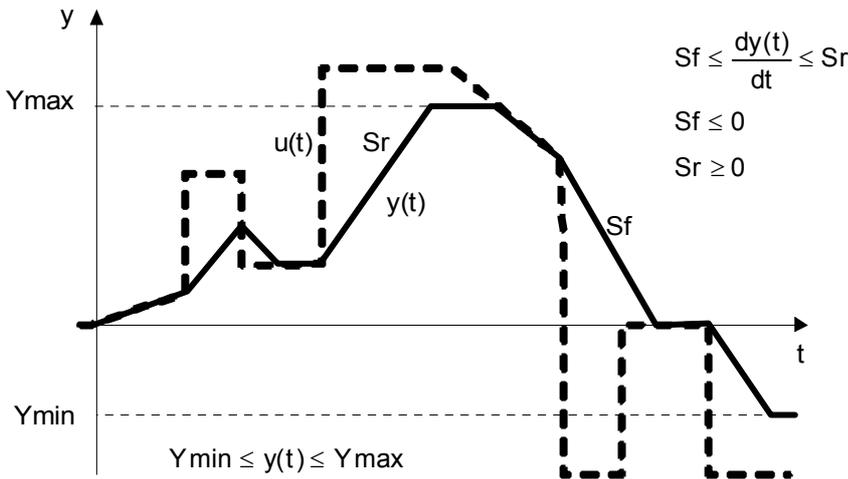
13.4 Stellglieder

13.4.1 Anstiegsbegrenzung

Dieses Verhalten können Stellglieder aufweisen, die dem Steuersignal des Reglers nicht zu folgen vermögen.

Das Glied begrenzt die negativen und positiven Signaländerungen auf einstellbare maximale Werte; ausserdem weist es noch die Eigenschaften eines Sättigungsgliedes auf.

Der Anfangswert Y_0 kann wahlweise vorgegeben werden oder aber gleich dem Eingangswert des Gliedes zum Zeitpunkt $t = 0$ gesetzt werden. Im letzteren Fall ist der für Y_0 eingegebene Wert ohne Bedeutung.



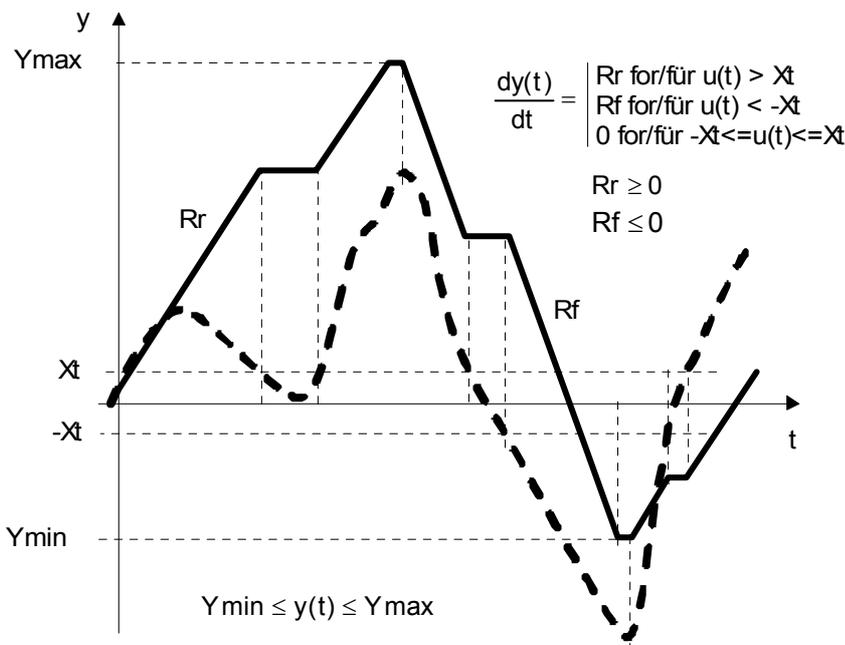
Anstiegsbegrenzung



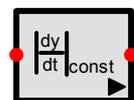
S_r 1 s-1
 S_f -1 s-1

13.4.2 Anstiegskonstante

Die Anstiegskonstante kennt nur drei mögliche Ausgangssignalverläufe. Wenn das Eingangssignal zu klein ist, bleibt der Ausgang konstant (Tote Zone), andernfalls steigt oder fällt er mit konstanter Geschwindigkeit, je nachdem, ob das Eingangssignal positiv oder negativ ist. Schlussendlich existiert auch eine Sättigung, die das Ausgangssignal in seiner Grösse begrenzt.



Anstiegskonstante



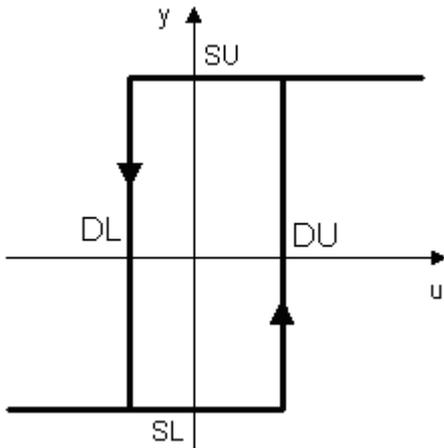
R_r 1 s-1
 R_f -1 s-1

Der Anfangswert Y_0 kann wahlweise vorgegeben werden oder aber gleich dem Eingangswert des Blocks zum Zeitpunkt $t = 0$ gesetzt werden. Im letzteren Fall ist der für Y_0 eingegebene Wert ohne Bedeutung.

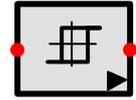
13.5 Regler

13.5.1 Zweipunkt

Der Zweipunktregler schaltet zwischen zwei festgelegten Signalzuständen.



Zweipunkt

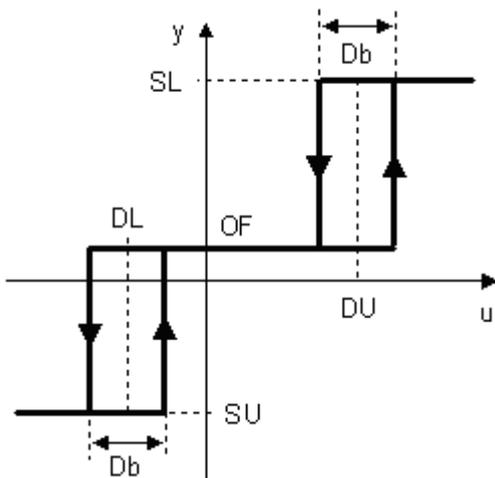


SU	1
SL	-1
DU	0.5
DL	-0.5
Y0	1

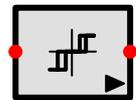
Y0 legt den Anfangswert zum Zeitpunkt $t \leq 0$ fest.

13.5.2 Dreipunkt

Beim Dreipunktregler kann die Ausgangsgrösse drei Zustände einnehmen.



Dreipunkt



SU	1
SL	-1
Db	1
DU	1
DL	-1
Y0	0

Y0 legt den Anfangswert zum Zeitpunkt $t \leq 0$ fest.

13.5.3 Idealer PI-Regler (PI)

Der PI-Regler wird auch als nachziehender Kompensator bezeichnet. Er wird benützt, falls das stationäre Verhalten der Regelstrecke ungenügend ist. Er wird nicht benützt, falls die Strecke selbst schon mindestens einen offenen Integrator besitzt.

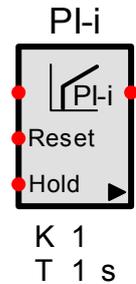
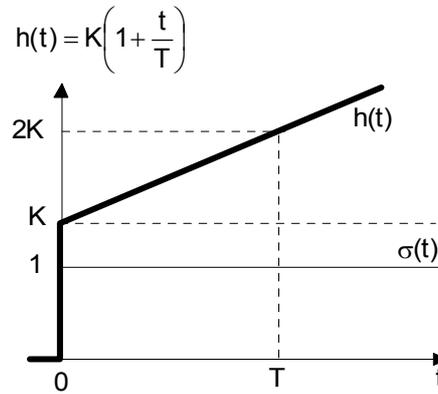
Mit dem idealen PI-Regler kann erreicht werden, dass bei Schrittingang keine stationäre Regeldifferenz mehr auftritt.

Funktionalbeziehungen

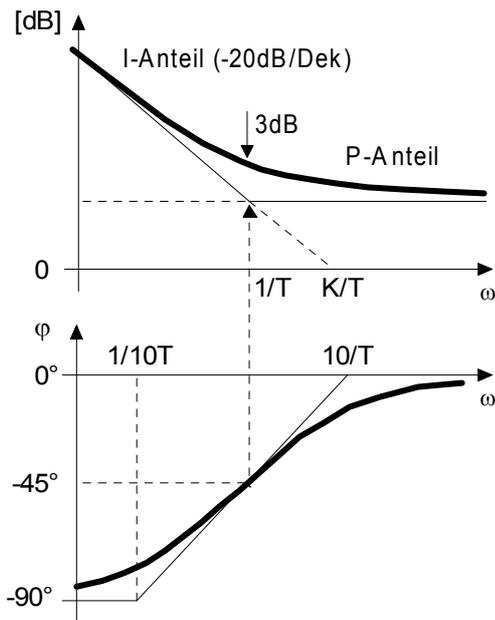
$$y = K \left(u + \frac{1}{T} \int_0^t u(\tau) d\tau \right)$$

$$G(s) = K \frac{1 + sT}{sT}$$

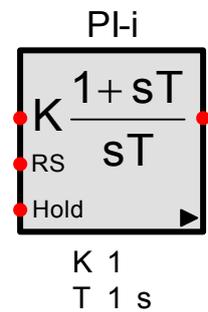
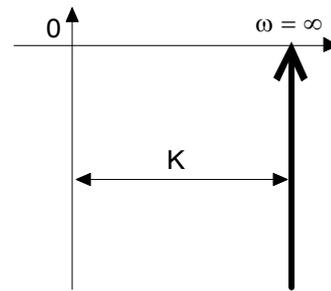
Schrittantwort



Amplituden- und Phasengang



Ortskurve



13.5.4 Modifizierter PI-Regler (PI-m)

Der PI-Regler wird auch als nachheilender Kompensator bezeichnet. Er wird benützt, falls das stationäre Verhalten der Regelstrecke ungenügend ist. Er wird nicht benützt, falls die Strecke selbst schon mindestens einen offenen Integrator besitzt.

Die modifizierte Form eines PI-Reglers erfüllt im Wesentlichen dieselbe Aufgabe, wie der ideale PI-Regler. Die stationäre Genauigkeit des idealen PI-Reglers vermag er aber nicht zu erzeugen.

Funktionalbeziehungen

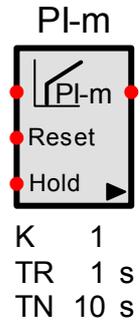
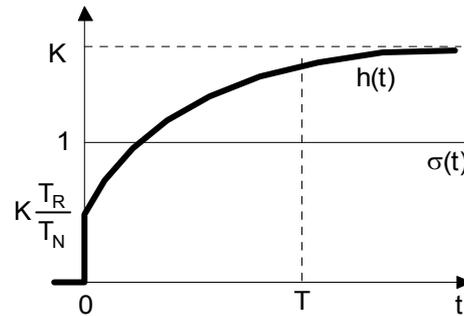
$$y + \dot{y}T_N = K(u + \dot{u}T_R)$$

$$G(s) = K \frac{1 + sT_R}{1 + sT_N}$$

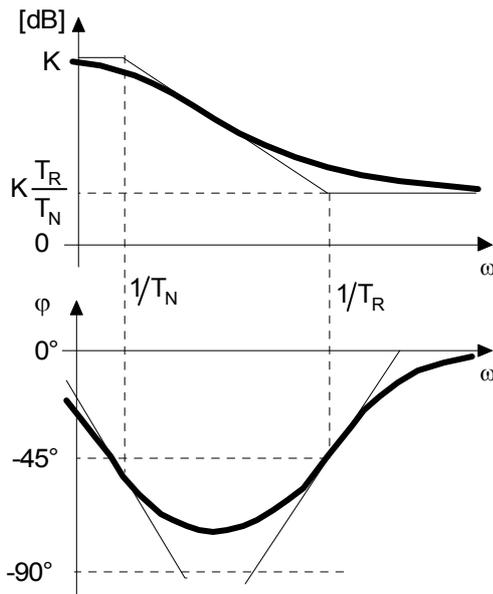
$$T_N \gg T_R$$

Schrittantwort

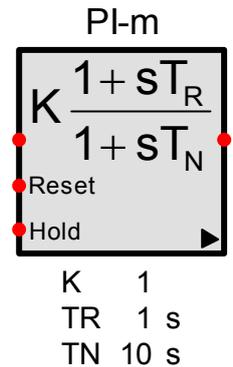
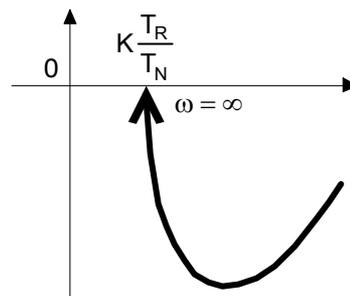
$$h(t) = K \left[1 + \left(\frac{T_R}{T_N} - 1 \right) e^{-\frac{t}{T_N}} \right]$$



Amplituden- und Phasengang



Ortskurve



13.5.5 Idealer PD-Regler (PD-Regler)

PD-Reglerstrukturen werden auch als voreilende Kompensatoren bezeichnet. Sie werden benützt, falls das transiente Verhalten nicht den Anforderungen entspricht oder wenn das gegebene System instabil ist (Phasenhebung von 0° bis 90°).

Sie werden nicht benützt, wenn die Strecke selbst keine offenen Integratoren besitzt.

Der ideale PD-Regler ist vor allem von theoretischem Interesse. Er ist schwierig zu konstruieren und ausserdem wird das Rauschen in der Regelstrecke verstärkt.

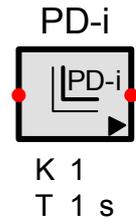
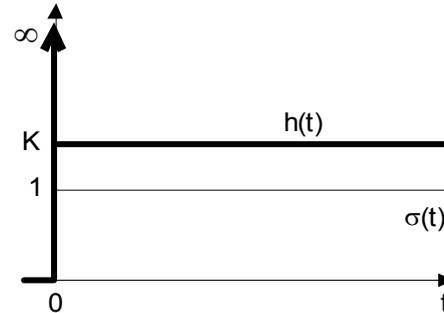
Funktionalbeziehungen

$$y = K(u + T\dot{u})$$

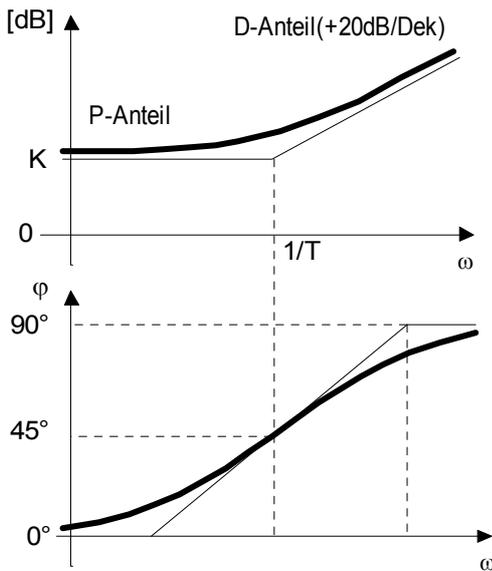
$$G(s) = K(1 + sT)$$

Schrittantwort

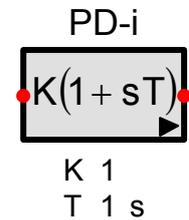
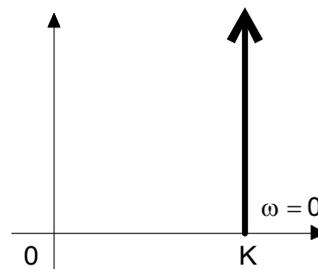
$$h(t) = K[\sigma(t) + T\delta(t)]$$



Amplituden- und Phasengang



Ortskurve



13.5.6 Realer PD-Regler (PD-r)

PD-Reglerstrukturen werden auch als voreilende Kompensatoren bezeichnet. Sie werden benützt, falls das transiente Verhalten nicht den Anforderungen entspricht oder wenn das gegebene System instabil ist (Phasenhebung von 0° bis 90°), aber nicht, wenn die Strecke selbst keine offenen Integratoren besitzt.

Der ideale PD-Regler ist vor allem von theoretischem Interesse. Er ist schwierig zu konstruieren und ausserdem wird das Rauschen in der Regelstrecke verstärkt.

Die reale Ausführung eines PD-Reglers erzeugt eine ähnliche Wirkung wie der ideale PD-Regler. Er entspricht den technischen Realisierungsmöglichkeiten und hält die Verstärkung des Rauschens in Grenzen.

Funktionalbeziehungen

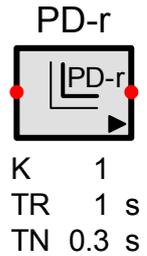
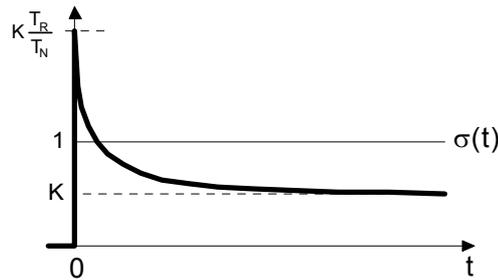
$$y + \dot{y}T_N = K(u + \dot{u}T_R)$$

$$G(s) = K \frac{1 + sT_R}{1 + sT_N}$$

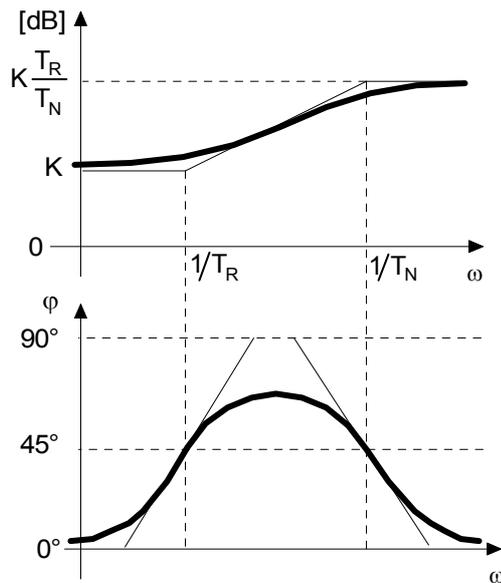
$$T_N < T_R$$

Schrittantwort

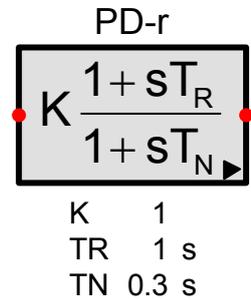
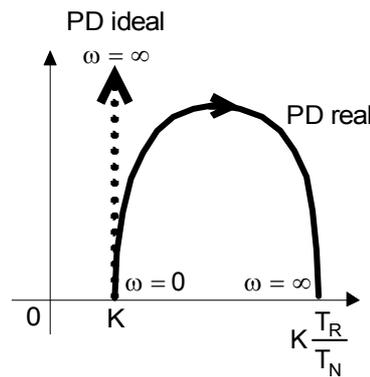
$$h(t) = K \left[1 + \left(\frac{T_R}{T_N} - 1 \right) e^{-\frac{t}{T_N}} \right]$$



Amplituden- und Phasengang



Ortskurve



13.5.7 Idealer PID-Regler, Typ I (PID-I)

PID-Reglerstrukturen werden eingesetzt, wenn sowohl das stationäre als auch das transiente Verhalten den Anforderungen nicht entsprechen.

Die erste Form des idealen PID-Reglers entspricht der Parallelschaltung je eines idealen P-,I- und D-Gliedes. Als Regelparameter treten die Verstärkung K, die Vorhaltezeit T_V und die Nachstellzeit T_N auf. Im Weiteren kann mit T_D die Differentiation abgeschwächt werden (siehe DT1-Glied, wobei dort $T_D = T_1$ entspricht).

Der Anfangswert der Ausgangsgröße wird bestimmt durch die Eingangsgröße, die Verstärkung und den Anfangswert Y_0I des Integrators: $Y_0 := U_0 \cdot K(1 + Y_0I)$;

Die Ausgangsgröße kann durch eine wählbare Anti-Windup-Halt-Funktionalität auf den Wertebereich Y_{min} ... Y_{max} beschränkt werden.

Anti-Windup-Hold

Beim Überschreiten der Grenzwerte wird der Integrator angehalten. Sobald der Ausgang wieder innerhalb der Schranken zu liegen kommt, wird die Integration wieder freigegeben.

Anti-Windup-Reset

Die Ausgangsgröße des Integrators wird nach dem Überschreiten eines Grenzwertes soweit zurückgestellt, bis die Ausgangsgröße des Reglers genau dem überschrittenen Grenzwert entspricht (Y_{max} oder Y_{min}).

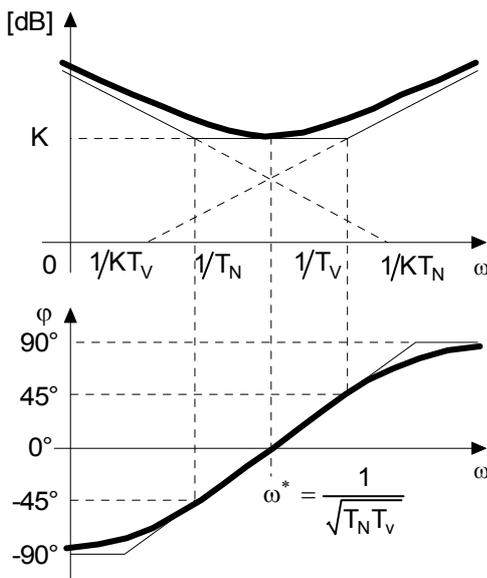
Funktionalbeziehungen

$$y = K \left(u + \frac{1}{T_N} \int_0^t u(\tau) d(\tau) + T_V \dot{u} \right)$$

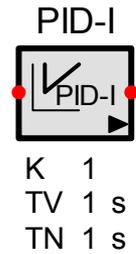
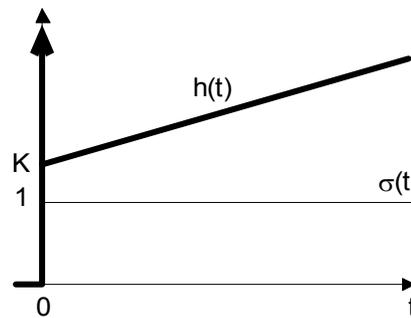
$$G(s) = K \left(1 + \frac{1}{sT_N} + sT_V \right)$$

$$T_N > T_V$$

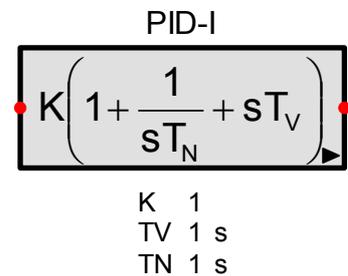
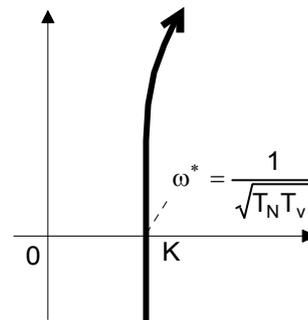
Amplituden- und Phasengang



Schrittantwort



Ortskurve



13.5.8 Adaptiver PID-Regler

Der Adaptive PID-Regler entspricht dem idealen PID-Regler Typ 1, wobei die voreingestellten Parameter K, TV, TN während der Simulation über Steuereingänge modifiziert werden können. Die Parametersteuerung kann multiplikativ oder additiv erfolgen. Die aktuellen Werte für die Parameter ergeben sich somit zu:

Parametersteuerung multiplikativ:

$$K = K0 * dK$$

$$TN = TNO * dTN$$

$$TV = TV0 * dTV$$

Bei offenen Steuereingängen werden die Steuersignale zu 1

Parametersteuerung additiv:

$$K = K0 + dK$$

$$TN = TNO + dTN$$

$$TV = TV0 + dTV$$

Bei offenen Steuereingängen werden die Steuersignale zu 0

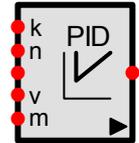
Auch die Begrenzungen können über Steuereingänge multiplikativ modifiziert werden.

$$Ymin = Ymin0 * dYBegr$$

$$Ymax = Ymax0 * dYBegr$$

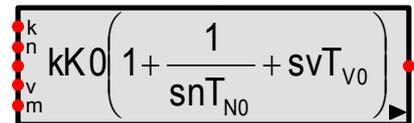
Bei offenen Steuereingängen bleiben die Begrenzungen unverändert.

PID
adaptiv



K0 1
TV0 1 s
TNO 1 s

PID
adaptiv



K0 1
TV0 1 s
TNO 1 s

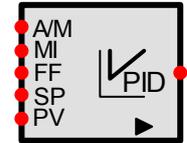
13.5.9 Industrie PID-Regler

Auf dem Markt werden je nach Anwendungsbereich unterschiedliche PID-Strukturen eingesetzt. Somit existiert auch kein standardisierter PID Algorithmus. Selbst in der regeltechnischen Literatur wird keine einheitliche Terminologie verwendet. Man stösst auf Begriffe wie "ideal", "parallel", "seriell", "ISA", "interaktiv", "nicht interaktiv" und mehr. Einige Bezeichnungen werden überall gleich gehandhabt, andere jedoch sehr unterschiedlich.

SimApp benutzt nur zwei Bezeichnungen, die sich durch die Anordnung des Differential- und Integralkanals unterscheiden: "seriell" und "parallel". Zusammen mit zwei weiteren Optionen (D on PV und P on PV) können aber damit alle bekannten Strukturen realisiert werden.

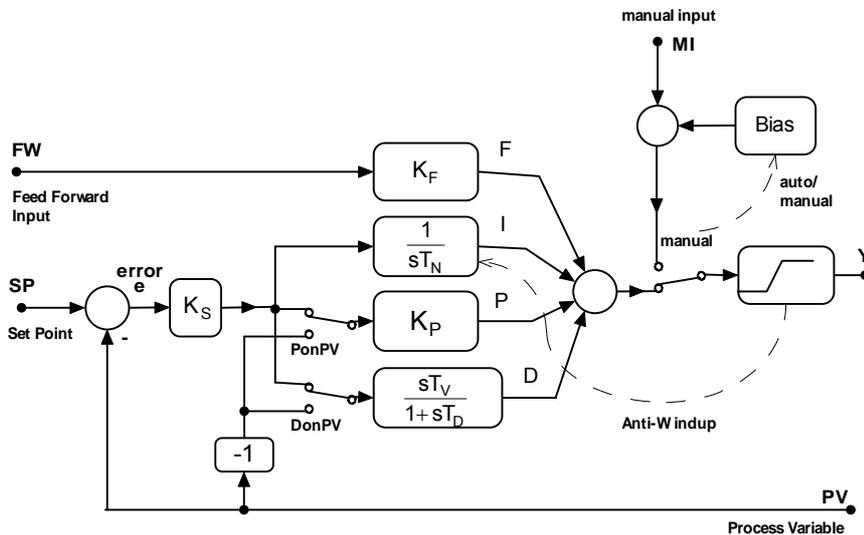
Bei der seriellen Ausführung sind Differentiation und Integration hintereinander geschaltet (interaktiv), wobei sie bei der parallelen Ausführung nebeneinander angeordnet sind und sich gegenseitig nicht beeinflussen (nicht interaktiv). Differentiation und Integration können wahlweise auch direkt auf die Regelgrösse (PV = Process variable) geschaltet werden.

Industrie PID-Regler

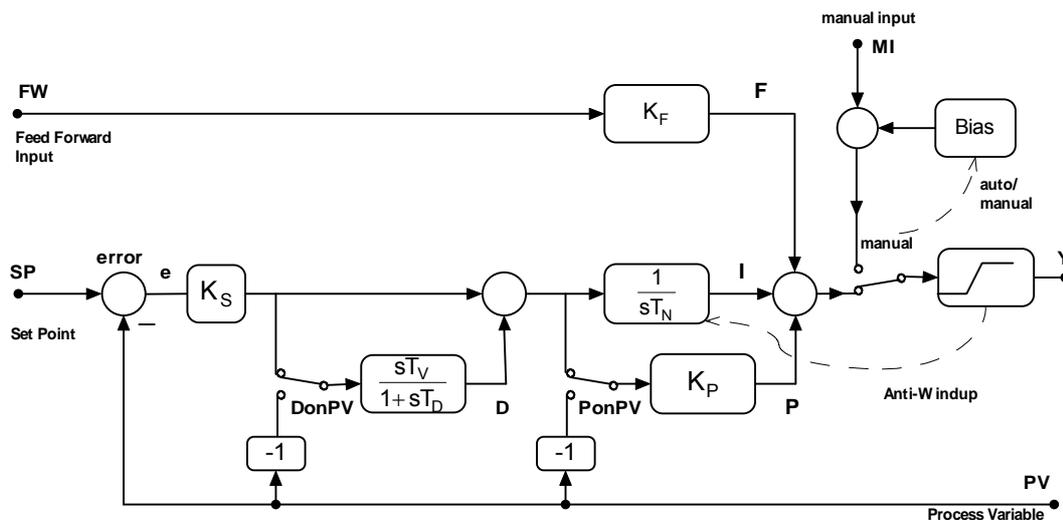


- Ks 1
- Kp 1
- TN 1 s
- TV 1 s

Parallele Form:



Serielle Form:



Eigenschaften und Optionen

PonPV:

Aus: Der P-Kanal reagiert auf die Regeldifferenz (standard)

Ein: Der P-Kanal reagiert nur auf die Regelgrösse (Process variable) und reduziert so den Einfluss von Störungen des Stellsignals auf die Regelung.

DonPV:

Aus: Der Differenzierer reagiert auf die Regeldifferenz (standard).

Ein: Der Differenzierer reagiert nur auf die Regelgrösse und verhindert damit die Erzeugung von Regelpulsen, die durch Störungen auf dem Stellsignal verursacht werden können.

Durchgangskanal

Der Durchgangskanal ermöglicht das direkte und sofortige Reagieren auf Änderungen des Stellsignals, unabhängig von der Regeldifferenz.

Manueller Modus

Die Regelungsaufgabe wird durch das manuelle Eingangssignal wahrgenommen. Die Umschaltung zwischen manuellem und automatischem Betrieb erfolgt absolut stossfrei (bumpless) indem automatisch ein Offset (Bias) zum manuellen Eingangssignal hinzuaddiert ist, so dass der Ausgang des Reglers konstant bleibt. Beim Umschalten vom manuellen in den automatischen Betrieb, wird entsprechend der Integrator gesetzt, so dass keine Stösse auftreten. Im manuellen Modus arbeiten P und D wie gewöhnlich, werden jedoch nicht für den Ausgang verwendet. Der Integrator wird angehalten und hat ebenfalls keinen Einfluss auf den Ausgang.

Filter erster Ordnung im D-Kanal

Um das Rauschen durch den Differenzierer nicht übermässig zu verstärken, wird die reine Ableitung mit einem Tiefpassfilter kombiniert, wobei diese zusammen ein DT_1 -Glieder bilden.

Ausgangsbeschränkung, Sättigung

Begrenzung der Ausgangsgrösse um Überlast zu vermeiden.

Anti-Windup-Hold

Die Integration wird sofort gestoppt, wenn das Ausgangssignal den durch das ausgangsseitige Sättigungsglied bestimmte Wertebereich verlässt. Sobald das Ausgangssignal wieder kleiner wird, wird der Integrator wieder losgelassen.

Anti-Windup-Reset

Wenn das Ausgangssignal den durch das ausgangsseitige Sättigungsglied bestimmte Wertebereich verlässt, wird der Integratorausgang so weit reduziert, dass die Summe aller Kanäle, die auf das Addierglied gehen exakt dem Sättigungswert entspricht (Y_{max} oder Y_{min}).

13.5.10 Idealer PID-Regler, Typ II (PID-II)

PID-Reglerstrukturen werden eingesetzt, wenn sowohl das stationäre als auch das transiente Verhalten den Anforderungen nicht entsprechen.

Die zweite Form des idealen PID-Reglers geht von der Bedeutung der Pol- und Nullstellen aus. Die Zuordnung der Regelparameter (Pol- und Nullstellen) auf die idealen Grundglieder ist nicht mehr so offensichtlich.

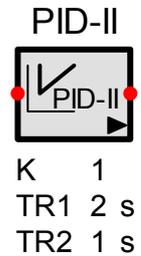
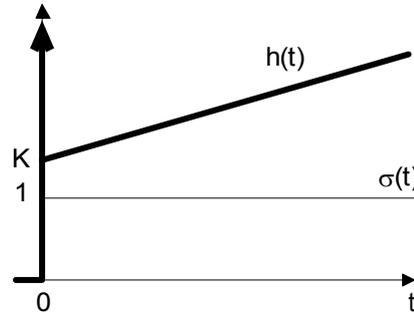
Funktionalbeziehungen

$$y = K \left[(T_{R1} + T_{R2})u + \int_0^t u(\tau)d\tau + \dot{u}T_{R1}T_{R2} \right]$$

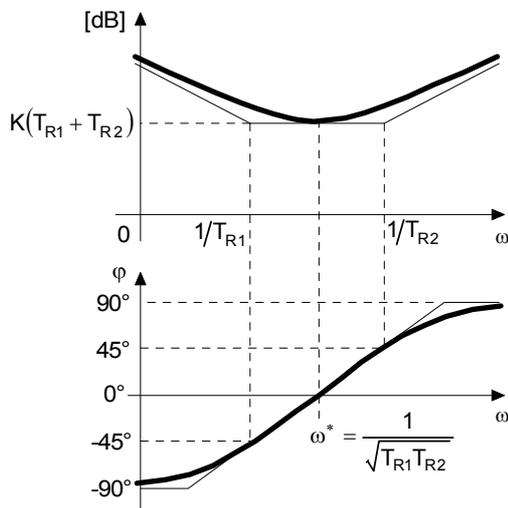
$$G(s) = K \frac{(1 + sT_{R1})(1 + sT_{R2})}{s}$$

$$T_{R1} \geq T_{R2}$$

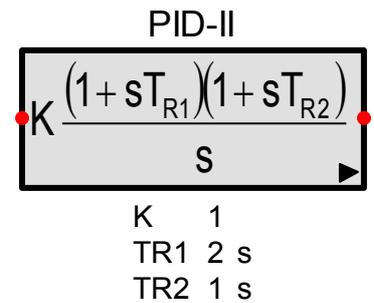
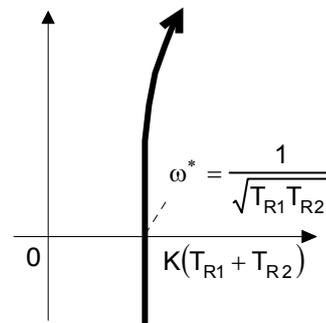
Schrittantwort



Amplituden- und Phasengang



Ortskurve



13.5.11 Realer PID-Regler (PID-r)

Der reale PID-Regler schwächt den Einfluss des D-Anteils im höheren Frequenzbereich (Rauschen) ab und kann einfach realisiert werden.

Funktionalbeziehungen

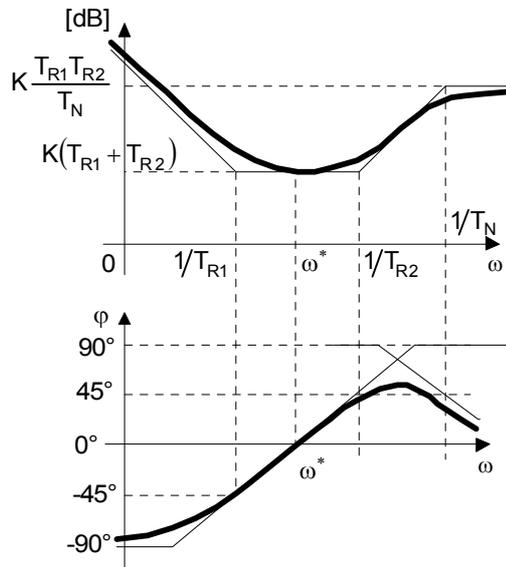
$$y + T_N \dot{y} = K \left[(T_{R1} + T_{R2}) u + \int_0^t u(\tau) d\tau + T_{R1} T_{R2} \dot{u} \right]$$

$$G(s) = K \frac{(1 + sT_{R1})(1 + sT_{R2})}{s(1 + sT_N)}$$

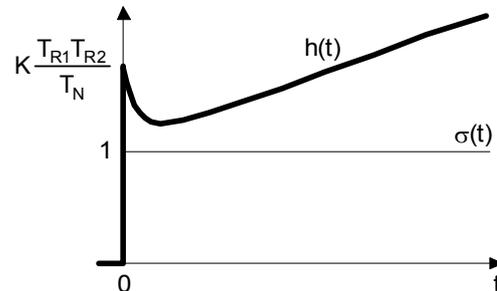
$$T_{R1} \geq T_{R2} > T_N$$

$$\omega^* = \frac{1}{\sqrt{T_{R1}T_{R2} - (T_{R1} + T_{R2})T_N}}$$

Amplituden- und Phasengang

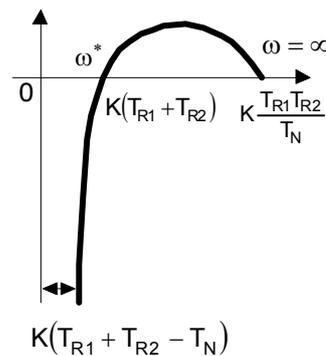


Schrittantwort



PID-r	
K	1
TR1	3 s
TR2	1 s
TN	0.5 s

Ortskurve



PID-r	
$K \frac{(1 + sT_{R1})(1 + sT_{R2})}{s(1 + sT_N)}$	
K	1
TR1	3 s
TR2	1 s
TN	0.5 s

13.5.12 Modifizierter PID-Regler (PIDm-Regler)

Der modifizierte PID-Regler eignet sich für gezielte und flexible Pol- und Nullstellenkompensation in der Strecke.

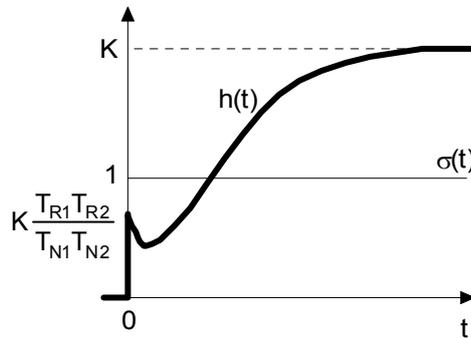
Funktionalbeziehungen

$$y + \dot{y}(T_{N1} + T_{N2}) + \ddot{y}T_{N1}T_{N2} = K[u + \dot{u}(T_{R1} + T_{R2}) + \ddot{u}T_{R1}T_{R2}]$$

$$G(s) = K \frac{(1 + sT_{R1})(1 + sT_{R2})}{(1 + sT_{N1})(1 + sT_{N2})}$$

$$T_{N1} \gg T_{R1} \geq T_{R2} > T_{N2}$$

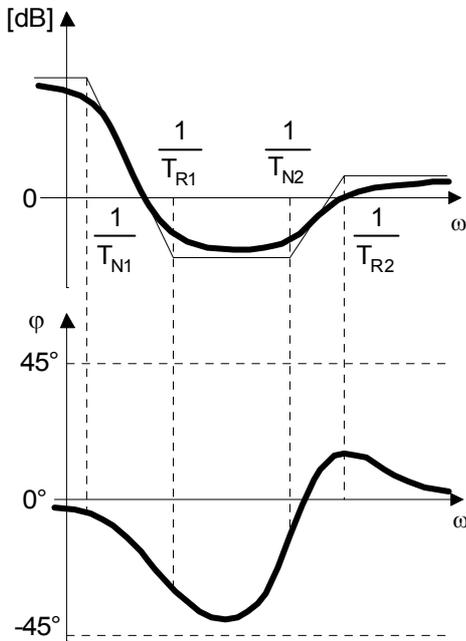
Schrittantwort



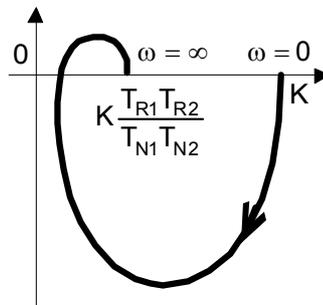
PID

K	1
TR1	2 s
TR2	1 s
TN1	10 s
TN2	0.5 s

Amplituden- und Phasengang



Ortskurve



PID

K	1
TR1	2 s
TR2	1 s
TN1	10 s
TN2	0.5 s

13.5.13 Verallgemeinerter PID-Regler (PID-a)

Mit dem verallgemeinerten PID-Regler sind auch komplexe Wurzeln im Zähler- und Nennerpolynom realisierbar. Alle anderen Strukturen sind jedoch durch geeignete Parameterwahl auch möglich. Eine typische Schrittantwort, Amplituden- und Phasengang und Ortskurve können wegen der möglichen Vielfalt nicht angegeben werden.

Funktionalbeziehungen

$$y + \dot{y}2d_N T_N + \ddot{y}T_N^2 = K[u + \dot{u}2d_Z T_Z + \ddot{u}T_Z^2]$$

$$G(s) = K \frac{1 + 2d_Z T_Z s + T_Z^2 s^2}{1 + 2d_N T_N s + T_N^2 s^2}$$

PID-a

$$K \frac{1 + 2d_Z T_Z s + T_Z^2 s^2}{1 + 2d_N T_N s + T_N^2 s^2}$$

K	1
dZ	1
TZ	1 s
dN	2
TN	2 s

PID-a

K	1
dZ	1
TZ	1 s
dN	2
TN	2 s

13.5.14 Lead/Lag-Regler

PID-Reglerstrukturen werden eingesetzt, wenn sowohl das stationäre als auch das transiente Verhalten den Anforderungen nicht entsprechen. Der Lead/Lag-Regler entspricht dem modifizierten PID-Regler, wobei jedoch die Aufteilung in einen voreilenden (lead) und einen nacheilenden (lag) Anteil besonders übersichtlich ist.

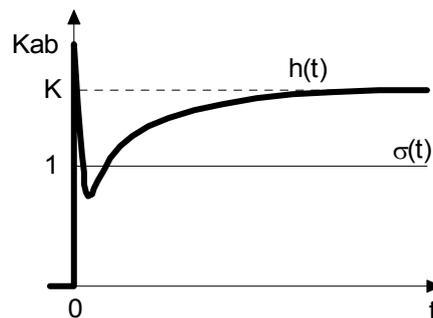
Funktionalbeziehungen

$$y + \dot{y}(T_1 + T_2) + \ddot{y}T_1 T_2 = K[u + \dot{u}(aT_1 + bT_2) + \ddot{u}abT_1 T_2]$$

$$G(s) = K \left(\frac{1 + aT_1 s}{1 + T_1 s} \right) \left(\frac{1 + bT_2 s}{1 + T_2 s} \right)$$

$T_1 < T_2$
 $a > 1$
 $b < 1$

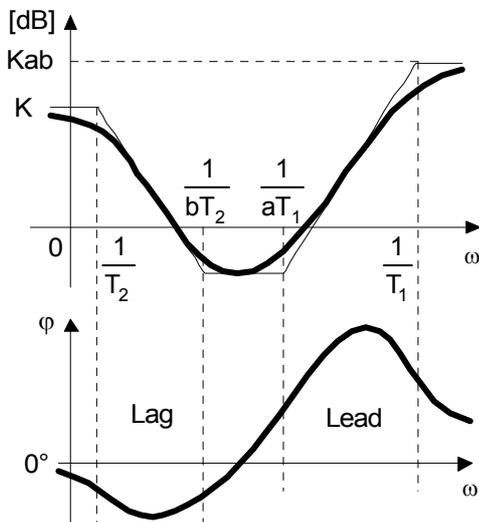
Schrittantwort



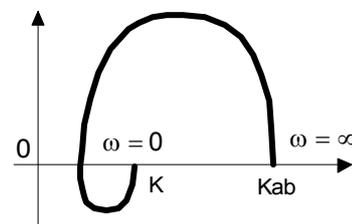
Lead/Lag
Regler

K	1
a	4
T1	0.1 s
b	0.5
T2	1 s

Amplituden- und Phasengang



Ortskurve



Lead/Lag
Regler

$$K \left(\frac{1 + aT_1 s}{1 + T_1 s} \right) \left(\frac{1 + bT_2 s}{1 + T_2 s} \right)$$

K	1
a	4
T1	0.1 s
b	0.5
T2	1 s

13.6 Zeitdiskrete Übertragungsglieder

13.6.1 Einführung

In einem Abtastregelkreis wird durch die Abtastung aus dem Istwertverlauf der Regelgrösse eine Folge von zeitdiskreten Werten gebildet. Diese Werte repräsentieren den Signalverlauf an den diskreten Zeitpunkten $t = kT$ und werden dem zeitdiskreten Regler zugeführt. Der Signalverlauf zwischen den Abtastzeitpunkten spielt keine Rolle, da er nicht mehr betrachtet wird.

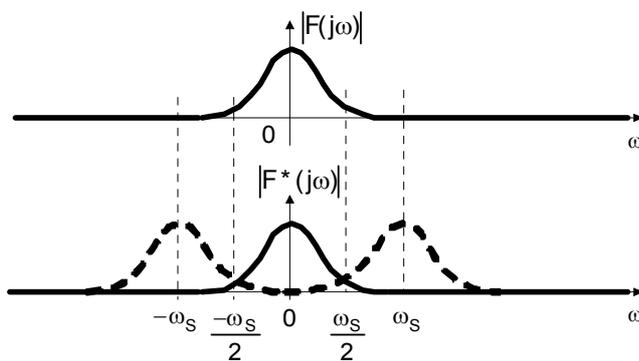
Der zeitdiskrete Regler erzeugt aus dieser Eingangswertefolge eine Ausgangswertefolge, die ebenfalls nur zu den diskreten Abtastzeitpunkten $t = kT$ definiert ist. Weil jedoch das Stellglied bzw. die Regelstrecke nach dem Regler nicht ein pulsartiges Signal, das nur zu den Abtastzeitpunkten den korrekten Wert aufweist, sondern ein zeitkontinuierliches Signal benötigt, das über den ganzen Zeitverlauf definiert ist, werden die Ausgangswerte zwischen den Abtastzeitpunkten konstant gehalten.

SimApp kann zeitdiskrete Glieder sowohl im Zeit- als auch im Frequenzbereich simulieren, wobei jedoch folgende Unterschiede zu beachten sind:

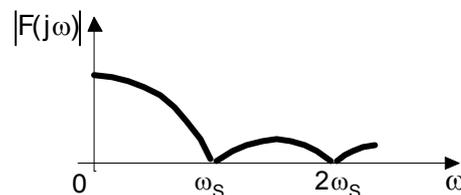
Alle zeitdiskreten Glieder in SimApp besitzen im Zeitbereich am Eingang einen virtuellen Abtaster und am Ausgang ein virtuelles Halteglied. Zeitdiskrete Glieder dürfen für eine Zeitsimulation an beliebigen Orten im System platziert werden, Abtast- und Halteglieder sind erlaubt aber nicht notwendig. Der Signalverlauf bleibt in beiden Fällen gleich.

In Frequenzsimulationen spielt der Übergang zwischen zeitkontinuierlicher und zeitdiskreter Verarbeitung jedoch eine wesentlichere Rolle. Der Abtastvorgang bewirkt eine Entstehung von Seitenbandfrequenzen mit konstanter Gewichtung durch die periodische Repetition des Spektrums des zeitkontinuierlichen Signals. Das Halteglied weist eine Tiefpasscharakteristik mit frequenzabhängiger Gewichtung auf.

Abtasten



Halten



Um einen korrekten Frequenzgang zu erhalten, müssen deshalb die Abtast- und Halteglieder korrekt platziert werden. Sie könnten jedoch weggelassen werden, wenn der untersuchte Frequenzbereich wesentlich kleiner als die Abtastfrequenz ist.

Da es sich bei der Frequenzsimulation nur um lineare Systeme handelt, dürfen Abtast- und Haltglied auch durch ein Abtast/Halteglied ersetzt werden.

Beachte

Die meisten analogen Übertragungsglieder haben die Möglichkeit, sie als zeitdiskrete Glieder zu betreiben. Das Blocksymbol wird in diesem Fall mit blauem Hintergrund versehen. Die zeitdiskrete Übertragungsfunktion wird durch das Ersetzen der komplexen Frequenz $s = j\omega$ durch ein Vorwärts-, Rückwärts- oder Trapez Substitut erreicht:

$$\text{Vorwärtssubstitut: } s = \frac{z-1}{h}$$

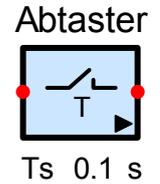
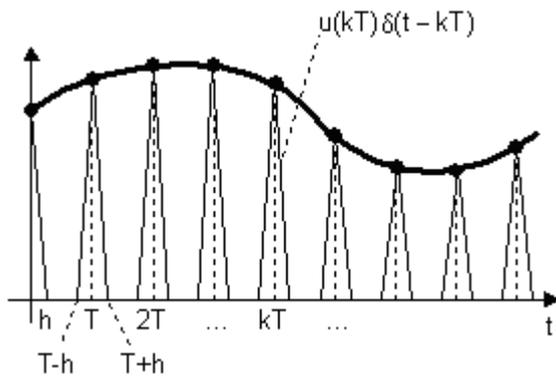
$$\text{Rückwärtssubstitut: } s = \frac{z-1}{hz}$$

$$\text{Trapezsubstitut: } s = \frac{2z-1}{h(z+1)}$$

wobei h die Abtastperiode darstellt.

13.6.2 Abtastglied

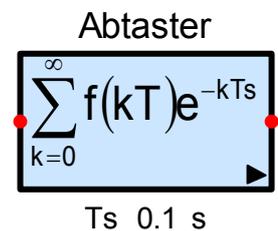
Das Abtastglied entnimmt zu den Zeitpunkten kT ($T =$ Abtastperiode) aus dem Eingangssignal die jeweils aktuellen Signalwerte und erzeugt am Ausgang eine Impulsfolge:



Zeitbereich:

$$y(t) = \sum_{k=0}^{\infty} u(kT)\delta(t - kT)$$

$$s(t) = \begin{cases} 1 & -h/2 \leq t < h/2 \\ 0 & \text{sonst} \end{cases}$$



Frequenzbereich:

$$Y^*(\omega) = \frac{1}{T} \sum_{n=-\infty}^{\infty} U\left(\omega - n \frac{2\pi}{T}\right)$$

wobei $U(\omega)$ das Spektrum des Eingangssignals ist.

Das Spektrum eines ideal abgetasteten Signals $u(t)$ ist gleich dem mit dem Abtastintervall T normierten und mit der Abtastfrequenz $2\pi / T$ periodisch repetiertem Spektrum des ursprünglichen, zeitkontinuierlichen Signals $u(t)$

Die oben dargestellte Form eines Impulsrechens entspricht der Wirklichkeit nur bedingt. Nach der Abtastung ist das Signal nur zu den Zeitpunkten kT definiert. Alle nachfolgenden zeitdiskreten Glieder benutzen nur diesen Wert. Die Form der Impulsflanken spielt demnach keine Rolle, da sie in der weiteren Verarbeitung nicht beachtet werden. Es ist deshalb nicht zulässig, hinter einem Abtaster zeitkontinuierliche Glieder zu schalten. Die resultierenden Signale würden nicht der Wirklichkeit entsprechen.

Gestattet sind demnach nur zeitdiskrete Glieder, also Glieder, die das Signal ebenfalls nur zu den Zeitpunkten kT betrachten und sich nicht um die Zwischenwerte kümmern.

Um ein abgetastetes und verarbeitetes Signal wieder einem zeitkontinuierlichen System zuzuführen, muss es mittels eines Haltegliedes jeweils zwischen zwei benachbarten Abtastzeitpunkten konstant gehalten werden. Dies erfolgt gewöhnlich mit einem Halteglied. Viele reale zeitdiskrete Glieder erfüllen diese Bedingung bereits von sich aus und besitzen an ihrem Ausgang quasi ein virtuelles Halteglied, so dass im Zeitbereich kein externes Halteglied benötigt wird. Dies gilt auch für alle zeitdiskreten Glieder in SimApp während einer Zeitsimulation. Bei Frequenzsimulationen müssen jedoch Abtast- und Halteglied paarweise existieren (siehe Einleitung zu den zeitdiskreten Gliedern)

13.6.3 Halteglied

Das Halteglied erzeugt aus einer Wertfolge $u(kT)$ ein treppenförmiges zeitkontinuierliches Signal, indem jeweils der letzte Wert über die ganze Abtastperiode T konstant gehalten wird. Dies ist ein Halteglied 0-ter Ordnung (zero order hold).

Im Zeitbereich können auch aus zeitkontinuierlichen Eingangssignalen zeitkontinuierliche, treppenförmige Ausgangssignale erzeugt werden, da das Halteglied in SimApp auch eine inhärente Signalabtastung vornimmt.

Im Frequenzbereich muss jedoch beachtet werden, dass eine frequenzabhängige Signalgewichtung entsteht, und das Übertragungsverhalten nur in Kombination mit einem Abtaster korrekt ist.

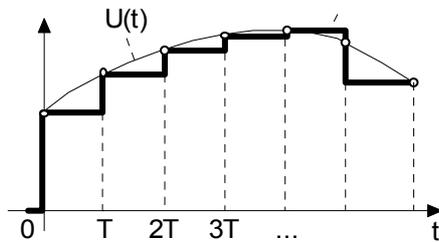
Funktionalbeziehungen

$$y(t) = \sum_{k=0}^{\infty} u(kT) [\sigma(t - kT) - \sigma(t - kT - T)]$$

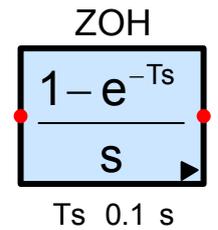
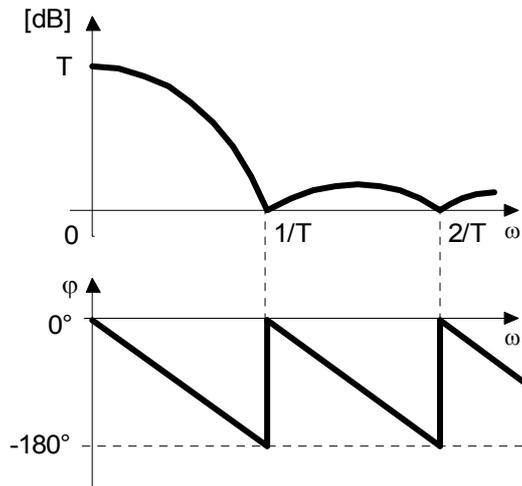
$$G(s) = \frac{1 - e^{-Ts}}{s} = T \frac{\sin \omega T/2}{\omega T/2} e^{-j\omega T/2}$$



Zeitantwort

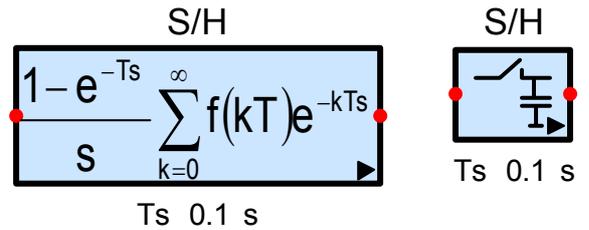


Amplituden- und Phasengang



13.6.4 Abtast-/Halteglied

Das Abtast-/Halteglied entnimmt zu den Zeitpunkten kT ($T =$ Abtastperiode) aus dem Eingangssignal die jeweils aktuellen Signalwerte und hält Sie als Ausgangswerte über die ganze Abtastperiode konstant.

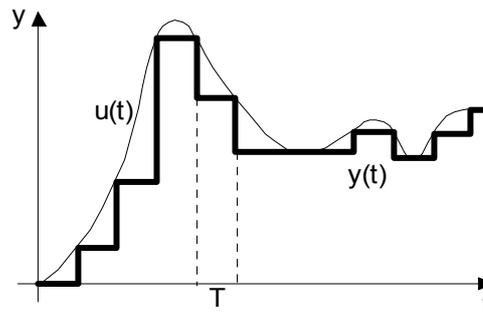


Funktionalbeziehungen

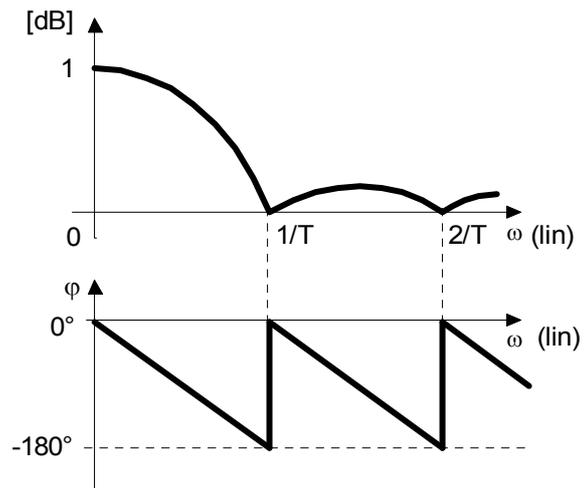
$$y(t) = \sum_{k=0}^{\infty} u(kT) [\sigma(t - kT) - \sigma(t - kT - T)]$$

$$Y(s) = \frac{1 - e^{-Ts}}{s} \sum_{k=0}^{\infty} u(kT) e^{-kTs}$$

Zeitantwort



Amplituden- und Phasengang



13.6.5 Zeitdiskreter Integrator (Iz)

Es gibt mehrere Möglichkeiten einen zeitdiskreten Integrator zu realisieren. In SimApp entspricht er der Approximation eines idealen analogen Integrators nach der Trapezregel.

Im Bereich der halben Abtastfrequenz und darüber treten starke Verzerrungen (Pole und Frequenzspiegelungen) auf. In abgetasteten Systemen muss das Abtastintervall deshalb genügend kurz sein und unerwünschte hohe Frequenzanteile müssen vor der Abtastung mit einem Tiefpassfilter unterdrückt werden

Funktionalbeziehungen

Trapezregel:

$$y_k = y_{k-1} + \frac{1}{T_i} \frac{u_k + u_{k-1}}{2} T$$

Zustandsgleichung:

$$x_{k+1} = x_k + u_k$$

$$y_k = \frac{T}{2T_i} (2x_k + u_k)$$

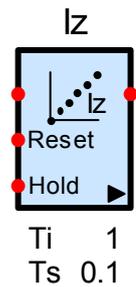
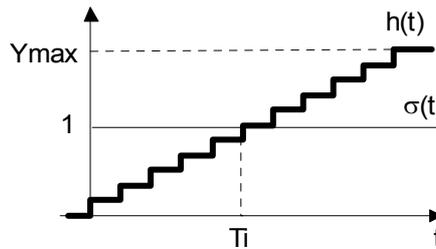
$$\underline{x}_k = \underline{x}(kT)$$

Übertragungsfunktion:

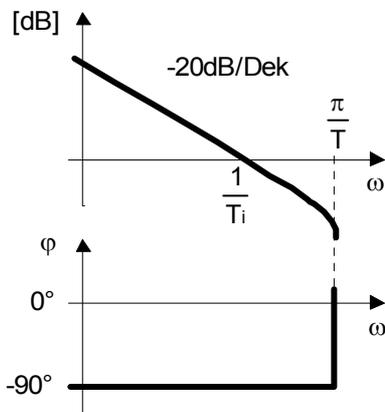
$$G(z) = \frac{T}{2T_i} \frac{z+1}{z-1}$$

$$z = e^{sT} = e^{j\omega T}$$

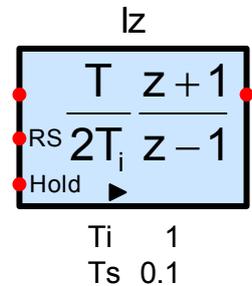
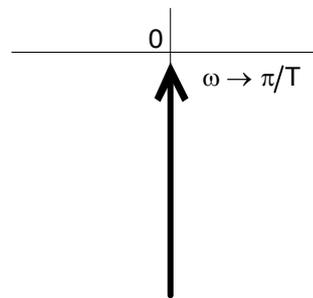
Schrittantwort



Amplituden- und Phasengang



Ortskurve



Der vorliegende zeitdiskrete Integrator kann auch einfach aus der analogen Form durch Anwendung der Bilinearen Transformation hergeleitet werden:

Übertragungsfunktion des analogen Integrators: $G(s) = \frac{1}{sT_i}$

Bilineare Transformation: $s = \frac{2}{T} \frac{z-1}{z+1}$

Durch Einsetzen von s in G(s) erhält man G(z)

13.6.6 Zeitdiskreter Differenzierer (Dz)

Der zeitdiskrete Differenzierer entsteht durch die Approximation des idealen analogen Differenzierers mittels Differenzenbildung.

Im Bereich der halben Abtastfrequenz und darüber treten starke Verzerrungen (Pole und Frequenzspiegelungen) auf. In abgetasteten Systemen muss das Abtastintervall deshalb genügend kurz sein und unerwünschte hohe Frequenzanteile müssen vor der Abtastung mit einem Tiefpassfilter unterdrückt werden

Funktionalbeziehungen

Differenzenbildung:

$$y_k = T_D \frac{u_k - u_{k-1}}{T}$$

Zustandsgleichung:

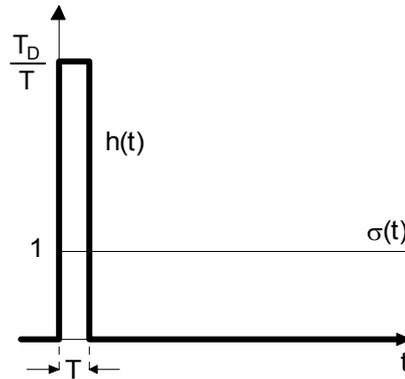
$$x_{k+1} = u_k$$

$$y_k = \frac{T_D}{T} (-x_k + u_k)$$

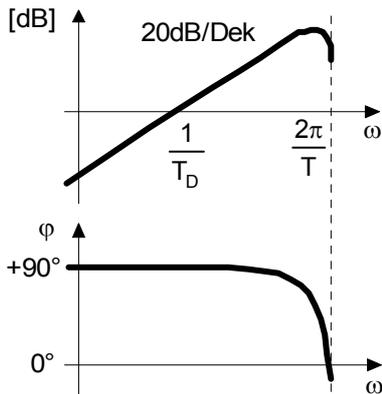
Übertragungsfunktion:

$$G(z) = \frac{T_D}{T} \frac{z-1}{z}$$

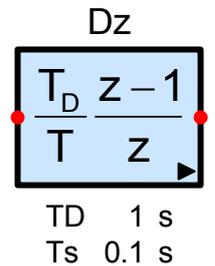
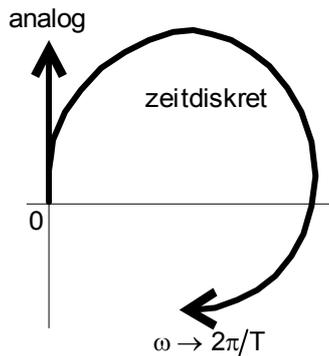
Schrittantwort



Amplituden- und Phasengang



Ortskurve



13.6.7 Einheitsverzögerung

Die Einheitsverzögerung verzögert ein zeitdiskretes oder analoges Signal um das Abtastintervall T. Das Symbol kann in 90°-Schritten gedreht werden.

Funktionalbeziehungen

$$y(t) = u(t-T)$$

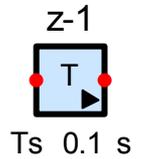
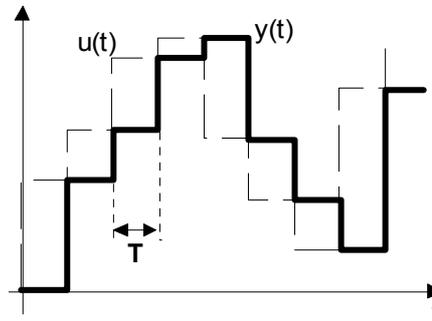
Schrittantwort:

$$h(t) = \sigma(t-T)$$

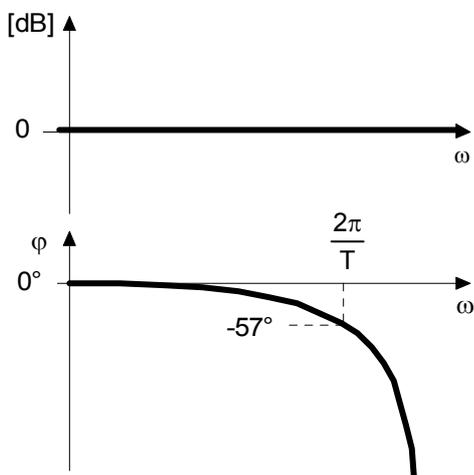
Übertragungsfunktion:

$$G(z) = z^{-1}; \quad z := e^{sT}$$

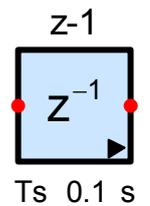
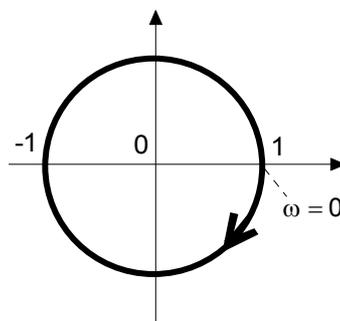
Zeitantwort



Amplituden- und Phasengang



Ortskurve



13.6.8 Zeitdiskreter PID-Regler (PIDz)

Der zeitdiskrete PID-Regler ist das Pendant zum idealen analogen PID-Regler, Typ 1. Es besteht aus der Parallelschaltung eines P-, I- und D-Gliedes, die einzeln ausgeschaltet werden können.

Als Regelparameter treten wiederum die Verstärkung K, die Vorhaltezeit TV und die Nachstellzeit TN auf. Dies erlaubt einen direkten Vergleich zwischen der analogen und der zeitdiskreten Ausführung.

Die Ausgangsgröße kann durch zwei verschiedene Anti-Windup-Halt-Massnahmen auf den Wertebereich Ymin ... Ymax beschränkt werden.

Anti-Windup-Hold

Beim Überschreiten der Grenzwerte wird der Integrator angehalten. Sobald der Ausgang wieder innerhalb der Schranken zu liegen kommt, wird die Integration wieder freigegeben.

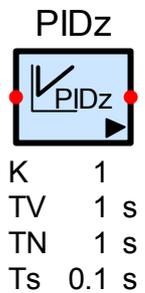
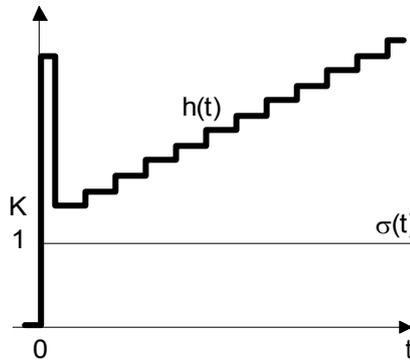
Anti-Windup-Reset

Die Ausgangsgröße des Integrators nach dem Überschreiten eines Grenzwertes soweit zurückgestellt, dass die Ausgangsgröße des Reglers genau dem überschrittenen Grenzwert entspricht (Ymax oder Ymin).

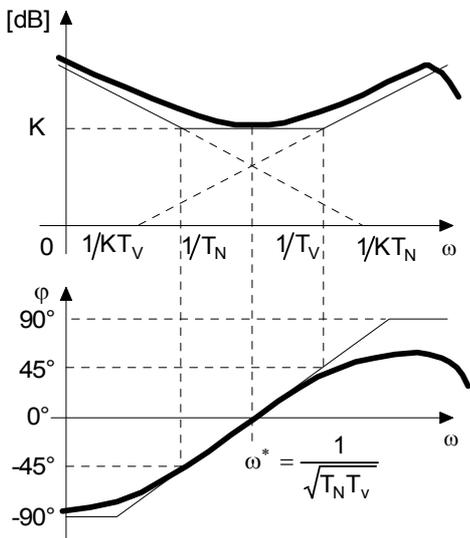
Funktionalbeziehungen

$$G(z) = K \left(1 + \frac{T/T_N (z+1)}{2(z-1)} + \frac{(z-1)}{T/T_V z} \right)$$

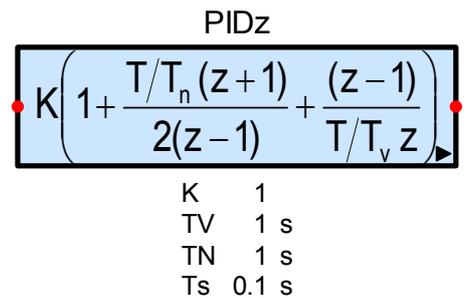
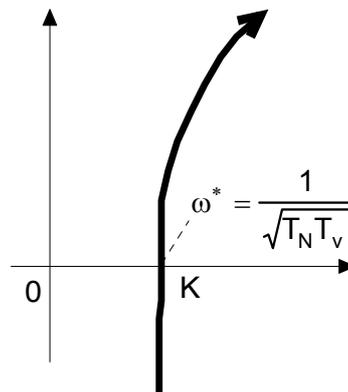
Schrittantwort



Amplituden- und Phasengang



Ortskurve



13.6.9 z-Übertragungsfunktion (G(z)-Glied)

Das G(z)-Glied ist die allgemeine Form eines linearen zeitdiskreten Übertragungsgliedes. Es ist das Pendant zum analogen G(s)-Glied und wird auch als digitaler Regler bezeichnet.

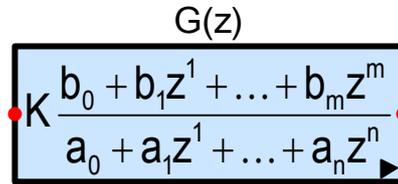
Ein Digitalrechnerprogramm muss kausal sein, d.h. zur Berechnung eines neuen Wertes y(kT) der Ausgangsgröße können nur die Eingangswerte u(kT), u(kT-T), ... benutzt werden. Daraus folgt, dass der Nennergrad grösser oder gleich dem Zählergrad sein muss.

Funktionalbeziehungen

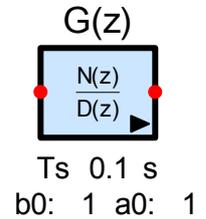
$$G(z) = K \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \dots + a_n z^{-n}}$$

$$m \leq n, a_n \neq 0$$

$$m \leq 31 \quad n \leq 31$$



Ts 0.1 s
b0: 1 a0: 1



Ts 0.1 s
b0: 1 a0: 1

Die Bedeutung der Anfangsbedingungen wird aus der Zustandsdarstellung ersichtlich:

$$\underline{x}_{k+1} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ & & & & 1 \\ -a_0^* & -a_1^* & -a_2^* & \dots & -a_n^* \end{bmatrix} \underline{x}_k + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1/a_n \end{bmatrix} u_k$$

$$\frac{y_k}{K} = [b_0 - a_0^* b_n \quad b_1 - a_1^* b_n \quad \dots \quad b_{n-1} - a_{n-1}^* b_n] \underline{x}_k + b_n^* u_k$$

$$a_i^* = \frac{a_i}{a_n} \quad b_i^* = \frac{b_i}{a_n}$$

13.6.10 Zeitdiskretes Filter (z-Filter)

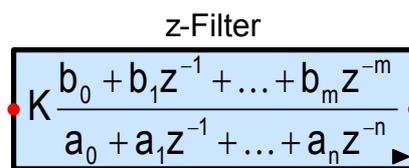
Bei diesem Glied entfallen die Einschränkungen des G(z)-Gliedes. Nennergrad und Zählergrad können frei gewählt werden. Die Kausalitätsbedingung gilt hier nicht mehr.

Funktionalbeziehungen

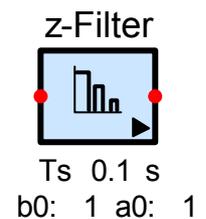
$$G(z) = K \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \dots + a_n z^{-n}}$$

$$a_0 \neq 0$$

$$m \leq 31 \quad n \leq 31$$



Ts 0.1 s
b0: 1 a0: 1



Ts 0.1 s
b0: 1 a0: 1

13.6.11 Lineares Differenzgleichungssystem

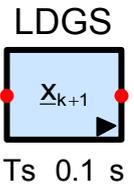
Frei parametrierbares, lineares und zeitinvariantes Differenzgleichungssystem. Dieses Glied ist das zeitdiskrete Pendant zum linearen, zeitinvarianten Differentialgleichungssystem.

Vektorform

$$\underline{x}_{k+1} = A\underline{x}_k + B\underline{u}_k$$

$$\underline{y}_k = C\underline{x}_k + D\underline{u}_k$$

wobei $\underline{x}_k = \underline{x}(kT)$



Das System hat n Zuständen, p Eingangs- und q Ausgangsgrößen

\underline{x}_k Zustandsvektor (nx1)

\underline{u}_k Eingangsvektor (px1)

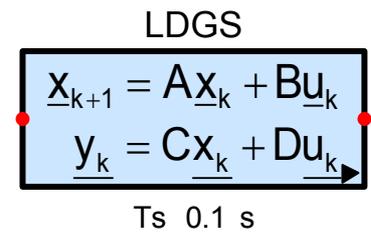
\underline{y}_k Ausgangsvektor (qx1)

A Systemmatrix (nxn)

B Eingangsmatrix (n x p)

C Ausgangsmatrix (q x n)

D Durchgangsmatrix (q x p)



Parameter: n<=50, p<=50, q<= 50, Koeffizienten A_{ik}, B_{ik}, C_{ik}, D_{ik}, Anfangswerte x_{nk}(i)

Matrizendarstellung:

$$\begin{bmatrix} \dot{x}_{k+1,1} \\ \dot{x}_{k+1,2} \\ \cdot \\ \dot{x}_{k+1,n} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & \cdot & A_{1n} \\ A_{21} & A_{22} & \cdot & A_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ A_{n1} & A_{n2} & \cdot & A_{nn} \end{bmatrix} * \begin{bmatrix} x_{k,1} \\ x_{k,2} \\ \cdot \\ x_{k,n} \end{bmatrix} + \begin{bmatrix} B_{11} & B_{12} & \cdot & B_{1p} \\ B_{21} & B_{22} & \cdot & B_{2p} \\ \cdot & \cdot & \cdot & \cdot \\ B_{n1} & B_{n2} & \cdot & B_{np} \end{bmatrix} * \begin{bmatrix} u_{k,1} \\ u_{k,2} \\ \cdot \\ u_{k,p} \end{bmatrix}$$

$$\begin{bmatrix} y_{k,1} \\ y_{k,2} \\ \cdot \\ y_{k,q} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & \cdot & C_{1n} \\ C_{21} & C_{22} & \cdot & C_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ C_{q1} & C_{q2} & \cdot & C_{qn} \end{bmatrix} * \begin{bmatrix} x_{k,1} \\ x_{k,2} \\ \cdot \\ x_{k,n} \end{bmatrix} + \begin{bmatrix} D_{11} & D_{12} & \cdot & D_{1p} \\ D_{21} & D_{22} & \cdot & D_{2p} \\ \cdot & \cdot & \cdot & \cdot \\ D_{q1} & D_{q2} & \cdot & D_{qp} \end{bmatrix} * \begin{bmatrix} u_{k,1} \\ u_{k,2} \\ \cdot \\ u_{k,p} \end{bmatrix}$$

In realen Systemen ist meistens D = 0.

13.7 Wandler

In SimApp wird zwischen Binär- und Digitalumsetzer unterschieden. Diese Unterscheidung, die in der Praxis nicht üblich ist, wird benutzt, um die unterschiedliche Art der Signalführung zu berücksichtigen.

Ein digitaler Ein- oder Ausgang ist in SimApp ein einzelner Knoten, der alle Bits auf einmal übertragen kann. Der digitale Zahlenwert wird als analoges Signal übertragen, das nur positive und ganzzahlige Werte annehmen kann. Z. B. binär 0 entspricht 0, binär 1101 (13 dezimal) entspricht dem Signalwert 13. Es macht keinen Sinn, digitale Signale mit analogen Signalen zu vermischen, deshalb unterscheiden sich Signalleitungen, die digitale Signale führen, farblich von den andern. Der Vorteil dieser analogen Codierung digitaler Werte liegt im geringen Platzbedarf der Glieder und der Übertragung (nur eine Leitung).

Ein binärer Ein- oder Ausgang in SimApp entspricht eher den gängigen Vorstellungen. Für jedes Bit wird eine eigene Signalleitung zur Übertragung benötigt. Der Nachteil dieser Lösung ist der grosse Platzbedarf für grössere Zahlen (grosse Glieder und mehrere Leitungen).

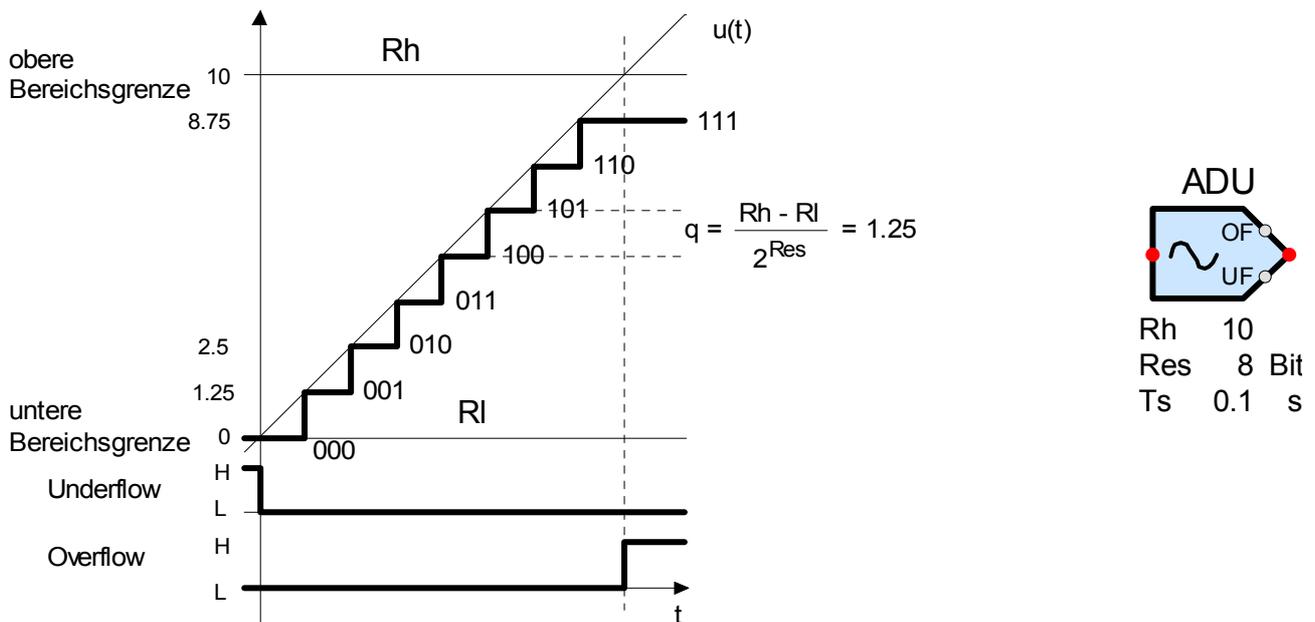
13.7.1 Analog/Digital Umsetzer (ADU, ADC)

Der ADC tastet das Signal in konstanten Zeitintervallen am Eingang ab und wandelt es in ein digitales Signal um. Das Glied besteht aus einem Abtaster, der das Eingangssignal über ein Abtastintervall konstant hält, und einem Quantisierer, der die Signalwerte auf ganzzahlige positive Werte rundet.

Das Abtastintervall, die Auflösung sowie die untere und obere Bereichsgrenze müssen vorgegeben werden. Standardmässig ist die untere Bereichsgrenze auf 0 gesetzt.

Wenn das Eingangssignal den gültigen Bereich verlässt, wird der Overflow oder Underflow Ausgang auf logisch 1 gesetzt.

Das folgende Beispiel zeigt einen ADC mit 3 Bit Auflösung im Bereich 0..10.



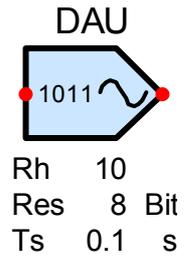
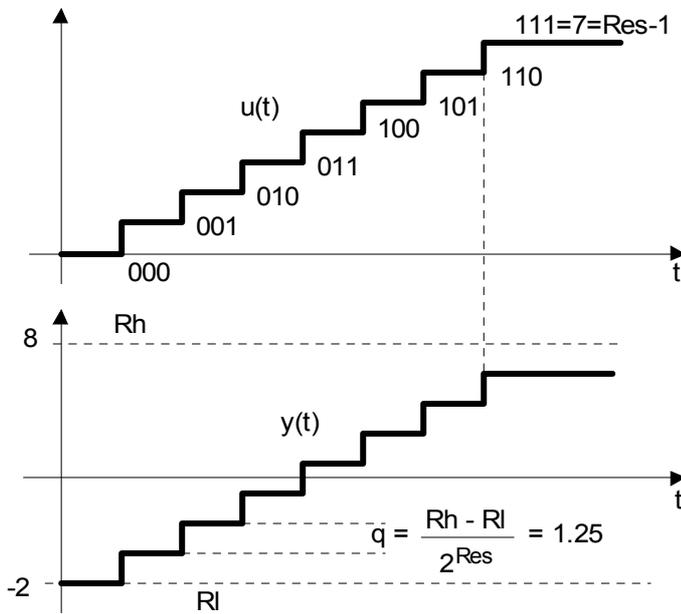
Bemerkung: Rh entspricht exakt dem binären Wert 1000, der jedoch nicht mehr in der vorgegebenen Wortbreite (3 Bit) Platz hat.

13.7.2 Digital/Analog Umsetzer (DAU, DAC)

Der DAC setzt den digitalen Eingangswert in ein Analogsignal um. Das Abtastintervall, die Auflösung sowie die untere und obere Bereichsgrenze müssen vorgegeben werden. Standardmässig ist die untere Bereichsgrenze auf 0 gesetzt.

Das Abtastintervall T muss gleich gross wie jenes des ADCs sein.

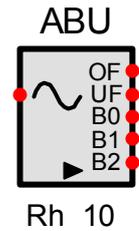
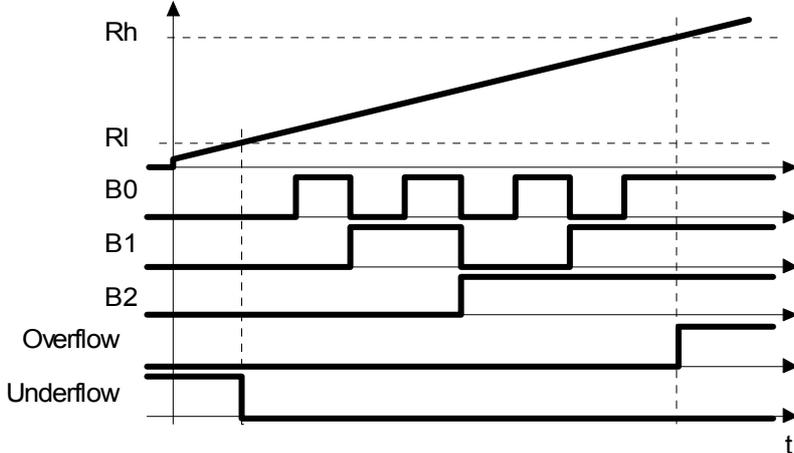
Das folgende Beispiel zeigt einen DAC mit 3 Bit Auflösung im Bereich -2 bis 8.



13.7.3 Analog/Binär Umsetzer (ABU, ABC)

Dieses Glied setzt ein analoges Eingangssignal in binär gewichtete Ausgangszustände um. Jeder binären Stelle ist ein Ausgang zugeordnet. Der zulässige Eingangsbereich und die darin gewünschte Auflösung in Bits (= Anzahl der Ausgänge) ist wählbar. Zulässig sind Auflösungen von 1 bis 31 Bit. Wenn das Eingangssignal den zulässigen Bereich verlässt, wird der Underflow (UF) oder Overflow (OF) Ausgang gesetzt. Im Underflow-Fall sind alle binären Ausgänge gelöscht und im Overflow-Fall gesetzt.

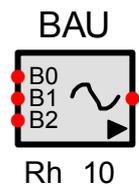
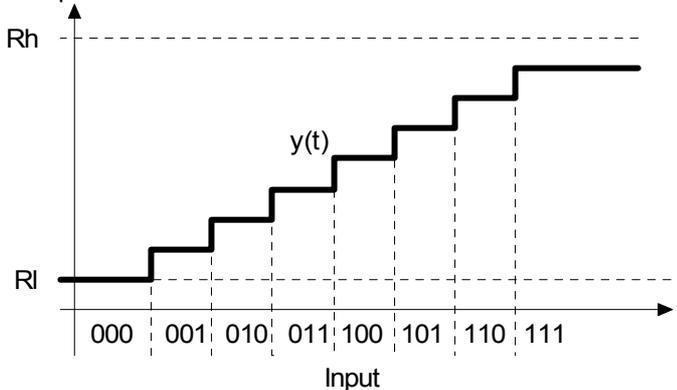
Beispiel: 3-Bit ABU



13.7.4 Binär/Analog Umsetzer (BAU, BAC)

Dieses Glied ist das Gegenstück zum Analog/Binär Umsetzer. Es setzt das binäre Eingangswort, repräsentiert durch einen oder mehrere binär gewichtete Eingänge, in ein analoges Ausgangssignal um. Die Auflösung (Anzahl Bits = Anzahl der binären Eingänge) kann zwischen 1 und 31 frei gewählt werden. Die beiden festzulegenden Bereichsgrenzen bestimmen den Ausgangswert für binär 0 und binär 11111..1 +1. Die obere Grenze des Bereichs kann nie ganz erreicht werden, weil sie den Ausgangswert für 2^n repräsentiert, der maximale binäre Eingangswert aber $2^n - 1$ (n = Anzahl Bits, Auflösung) ist.

Beispiel: 3 Bit BAU



13.7.5 Quantisierer

Das Ausgangssignal des Quantisierers kann nur diskrete Werte annehmen, die ein ganzzahliges Mehrfaches eines Grundwertes sind. Er folgt also einem rampenförmigen Eingangssignal in Form einer Treppenfunktion.

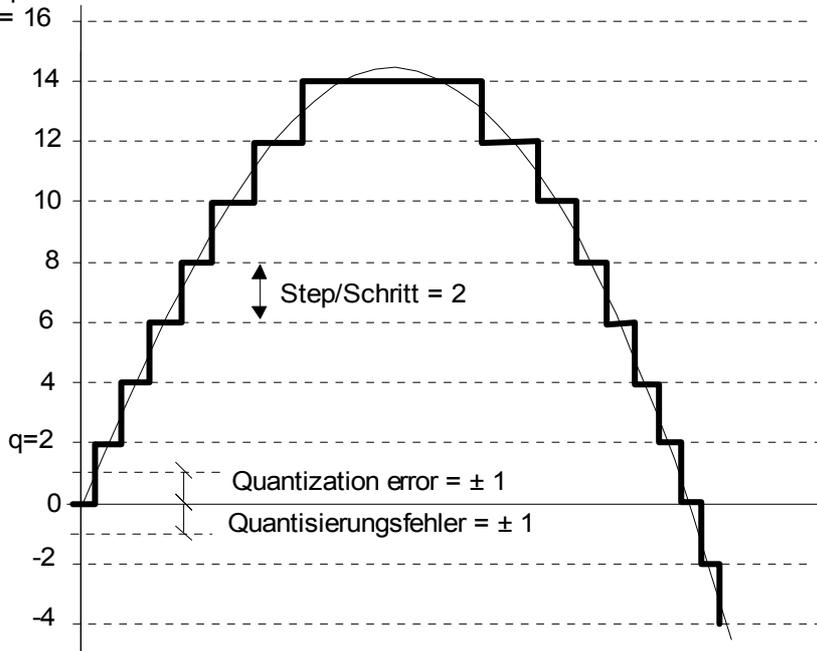
Die Schritthöhe wird durch die Wahl eines Signalbereichs und der darin gültigen Auflösung bestimmt. Bei einem Bereich von 16 und einer Auflösung von 3 Bit beträgt die Schritthöhe zum Beispiel $q = 16 / 2^3 = 2$.

Das Ausgangssignal wird jedoch durch den spezifizierten Bereich nicht begrenzt. Dessen Eingabe wird nur zur Berechnung der Schritthöhe benötigt.

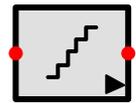
Der Quantisierer kann zur Simulation von digitalen Systemen eingesetzt werden, wo, bedingt durch die begrenzte Auflösung, nur diskrete Signalwerte vorkommen können. Die zeitdiskreten Glieder führen selber keine Quantisierung durch; sie übernehmen zu den diskreten Abtastzeitpunkten bloss die momentanen Eingangssignalwerte.

Beispiel: 3-Bit Quantisierer

Ra = 16



Quanti-
sierer



Res 8 Bit
Ra 10

13.8 Logik

Logikglieder führen bool'sche Operationen mit binären oder logischen Signalen aus. In SimApp besteht kein Unterschied zwischen einem logischen und einem analogen Signal. Logikelemente können durch Ausgangssignale von analogen Gliedern und umgekehrt angesteuert werden.

Eingangssignale, die kleiner als die logische Schaltschwelle sind, werden als 0 (tief, low, inaktiv, falsch) interpretiert. Im Gegensatz dazu werden Signale, die gleich oder grösser als die logische Schaltschwelle sind, als 1 (hoch, high, aktiv, wahr) interpretiert.

Der Standardwert für logisch 1 ist 1 mit einer Schaltschwelle bei 0.5. Diese Werte können für jedes Glied individuell eingestellt werden. Der logisch 0 Wert dagegen ist unveränderlich immer gleich null.

Rückkopplung mit logischen Gliedern und Bedeutung der Schaltverzögerung

Standardmässig besitzen Logikglieder in SimApp keine Laufzeiten. Wenn sich zum Beispiel bei einem Flip-Flop die Eingangssignale ändern und gleichzeitig der Clockeingang gesetzt wird, ändert der Ausgang noch im gleichen Integrationsschritt. Dadurch werden jedoch Signalarückkopplungen verunmöglicht, was aber in logischen Schaltkreisen durchaus normal ist. Viele Funktionen können ohne Rückkopplung nicht realisiert werden (Beispiel: Set/Reset Flip-Flop mit NAND Gattern).

Im Weiteren werden minimale Signallaufzeiten auch in diversen synchronen Schaltungen, die keine Rückkopplungen besitzen, vorausgesetzt. Beispiel: Bei einem synchronen Zähler der aus einer Kette von JK Flip-Flops aufgebaut wird, haben alle Flip-Flops das gleiche Takt-Eingangssignal. Zur korrekten Funktion der Schaltung ist es nun entscheidend, dass beim Auftreten der positiven Taktflanke der Zustand der Eingangssignale, der unmittelbar vor der Taktflanke geherrscht hat, eingelesen und verarbeitet wird. Wenn jedoch die Ausgänge exakt mit der Taktflanke wechseln, ist dies nicht möglich, da bereits die neuen Zustände berücksichtigt werden.

Um die oben genannten Fälle zu berücksichtigen, haben alle Logikglieder eine *Laufzeitverzögerungs*-Option. Wenn diese Option eingeschaltet ist, wechselt der Ausgang erst im nachfolgenden Integrations-schritt. Wenn der Integrationsschritt kurz ist (wählbar in den Einstellungen zur Zeitsimulation) und nur wenige Logikglieder hintereinander geschaltet sind, spielt die resultierende Gesamtverzögerung keine Rolle. Aber bei grossen Logikschaltkreisen oder langen Ketten von Logikelementen könnte sich dies zu inakzeptablen Verzögerungen aufsummieren, wenn jedes Glied unnötigerweise mit einer Laufzeitverzögerung behaftet wäre.

Es sollte deshalb folgende Regel beachtet werden:

1. Verzögerungsoption im Allgemeinen nur für ein einziges Glied in einer Signalarückkopplung einschalten.
2. Falls nötig, Integrationsschrittweite um den Faktor 10 reduzieren.

13.8.1 GND (Ground, Logisch 0, Masse)

Dieses Glied verbindet einen Eingang mit Masse (0, Ground, Erde, logisch 0) und arbeitet somit als Quelle, die andauernd den Wert 0 ausgibt. Normalerweise werden offene Eingänge in SimApp bereits als Masseverbindung interpretiert.

Diese logische Quelle kann auch in analogen Schaltkreisen verwendet werden.

Zum korrekten Positionieren kann das Symbol in 90 Grad Schritten gedreht werden.



13.8.2 V+ (Logisch 1)

Dieses Glied verbindet einen Eingang mit Speisung (logisch 1) und arbeitet somit als Quelle, die andauernd den Wert für logisch 1 ausgibt.

Diese logische Quelle kann auch in analogen Schaltkreisen verwendet werden.

Zum korrekten Positionieren kann das Symbol in 90 Grad Schritten gedreht werden.



13.8.3 UND (AND)

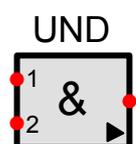
Dieses Logikglied kann mehrere Eingänge aufweisen und führt folgende logische Operation aus:

$$Y = I_1 \cdot I_2 \cdot \dots \cdot I_n$$

Die Anzahl der Eingänge ist zwischen 2 und 50 frei wählbar.

Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Die NAND-Funktion wird mit einem nachgeschalteten Inverter realisiert.



13.8.4 ODER (OR)

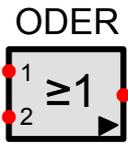
Dieses Logikglied kann mehrere Eingängen aufweisen und führt folgende logische Operation aus:

$$Y = I_1 + I_2 + \dots + I_n$$

Die Anzahl der Eingänge ist zwischen 2 und 50 frei wählbar.

Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Die NOR-Funktion wird mit einem nachgeschalteten Inverter realisiert.



13.8.5 Exklusiv-ODER (XOR, Antivalenz)

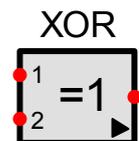
Diese Logikglied führt folgende logische Operation aus:

$$Y = I_1 \oplus I_2 \text{ resp. } Y = I_1 \neq I_2 \text{ resp. } Y = \bar{I}_1 \cdot I_2 + I_1 \cdot \bar{I}_2$$

Die Anzahl der Eingänge ist unveränderlich und beträgt zwei.

Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Die EXNOR Funktion $Y = I_1 = I_2$ (Äquivalenz) wird mit einem nachgeschalteten Inverter realisiert.



13.8.6 Inverter (NOT, Negation)

Dieses Logikglied führt eine logische Inversion aus:

$$Y = \bar{I} \text{ oder } Y = \text{not } I$$

Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrations-schritt verzögert.

Dieses Gatter besitzt zwei Symbole:

Das erste Symbol ist ein kleiner Kreis, so dass in Kombination mit einem anderen Logikglied ohne grossen Platzaufwand invertierende Ein- und Ausgänge realisiert werden können.

Das zweite Symbol ist ein Dreieck und eignet sich zum Einsatz als eigenständiges Glied.



13.8.7 SR Flip-Flop

Eine 1 am Set-Eingang setzt den Ausgang auf 1 (Q auf 1, Q- auf 0) und eine 1 am Reset-Eingang auf 0. Wenn beide Eingänge auf null wechseln, bleibt der Ausgang konstant.

Wenn Set und Reset beide auf 1 wechseln, ist der Zustand des Ausganges von der *Set Dominant*-Option abhängig.

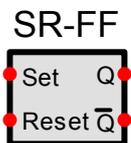
Wenn *Set Dominant* aktiviert ist, wird er Ausgang auch gesetzt, wenn neben Set auch Reset gesetzt ist. Wenn *Set Dominant* deaktiviert ist, ist Reset dominant, d.h. der Ausgang ist 0, wenn beide Eingänge 1 sind.

Wenn *Positiv flankengetriggert* aktiviert ist, finden Zustandsänderungen nur bei positiven Flanken statt, ansonsten sind die statischen Eingangspegel massgebend. Bei positiver Flankentriggerung entstehen keine Ausgangsänderungen, wenn die Eingänge von 1 auf 0 wechseln.

Der logische Anfangspegel zu Beginn der Simulation ist wählbar
Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Inputs		Outputs	
Set	Reset	Q	\bar{Q}
L	H	L	H
H	L	H	L
L	L	Q ₀	\bar{Q}_0
H	H	H	L
H	H	L	H

Set dominant
Reset dominant



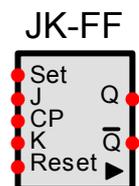
Q0 ist der Zustand von Q vor dem letzten Wechsel von Set und Reset

13.8.8 JK Flip-Flop

Ein 1 Pegel am Set- oder Reseteingang setzt den Eingang auf 1 oder 0, unabhängig vom Zustand der anderen Eingänge. Wenn Set und Reset inaktiv sind (0), werden die Daten am J- und K-Eingang durch die positive Taktflanke zum Ausgang transferiert. Die Daten am J- und K-Eingang haben zu anderen Zeitpunkten keine Auswirkungen auf den Ausgang. Wenn J und K auf 1 gehalten werden, arbeitet dieses Flip-Flop als Toggleschalter.

Der logische Anfangspegel zu Beginn der Simulation ist wählbar
Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Inputs					Outputs	
Set	Reset	CP	J	K	Q	\bar{Q}
H	L	X	X	X	H	L
L	H	X	X	X	L	H
H	H	X	X	X	H	H
L	L		L	L	Q ₀	\bar{Q}_0
L	L		H	L	H	L
L	L		L	H	L	H
L	L		H	H	Toggle	
L	L	L	X	X	Q ₀	\bar{Q}_0



- X = gleichgültig, don't care
- Q0 = Zustand von Q vor dem letzten Eingangswechsel oder Clock
- = 0-1 CP-Übergang

13.8.9 D Flip-Flop

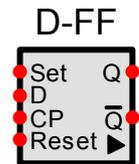
Die Information am D-Eingang wird zum Zeitpunkt der positiven Taktflanke verarbeitet und zum Ausgang weitergeleitet.

Vor und nach der Taktflanke hat der D-Eingang keinen Einfluss auf den Ausgang.

Der logische Anfangspegel zu Beginn der Simulation ist wählbar

Mit der Laufzeitverzögerung schaltet der Ausgang um einen Integrationsschritt verzögert.

Inputs				Outputs	
Set	Reset	CP	D	Q	\bar{Q}
H	L	X	X	H	L
L	H	X	X	L	H
H	H	X	X	H	H
L	L		H	H	L
L	L		L	L	H
L	L	L	X	Q_0	\bar{Q}_0



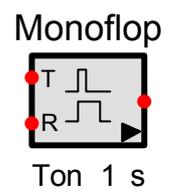
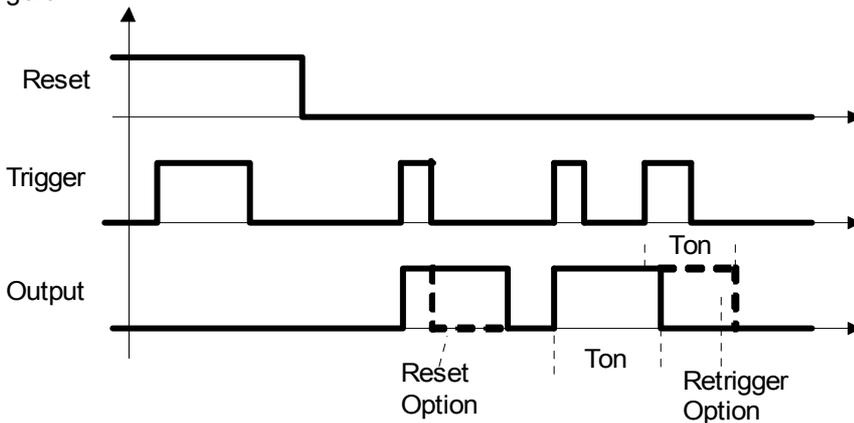
13.8.10 Monoflop

Der Monoflop weist einen flankengesteuerten Triggereingang und einen asynchronen Reseteingang auf. Eine positive Flanke am Trigger-Eingang setzt den Ausgang auf 1. Der Ausgang bleibt während der spezifizierten Ein-Zeit (Ton) auf 1 und ist unabhängig vom weiteren Signalverlauf am Trigger-Eingang, falls die Retrigger-Option nicht gesetzt ist (siehe Optionen). Der Ausgang wird nach der abgelaufenen Ein-Zeit (Ton) automatisch oder nach dem Setzen des Reset-Einganges vorzeitig wieder auf 0 gesetzt. Solange Reset auf 1 ist, kann der Ausgang nicht gesetzt werden.

Optionen

Wenn die *Reset*-Option eingeschaltet ist, wird der Ausgang sofort zurückgesetzt, wenn der Eingang auf 0 geht, unabhängig, ob die spezifizierte Ein-Zeit abgelaufen ist oder nicht.

Wenn die *Retrigger*-Option eingeschaltet ist, wird der Ausgang nochmals gesetzt, unabhängig vom momentanen Zustand. Wenn der Ausgang bereits gesetzt ist, wird der 1-Zustand um eine weitere Ton-Zeit verlängert.



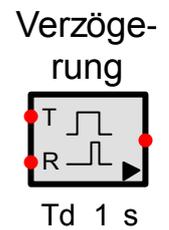
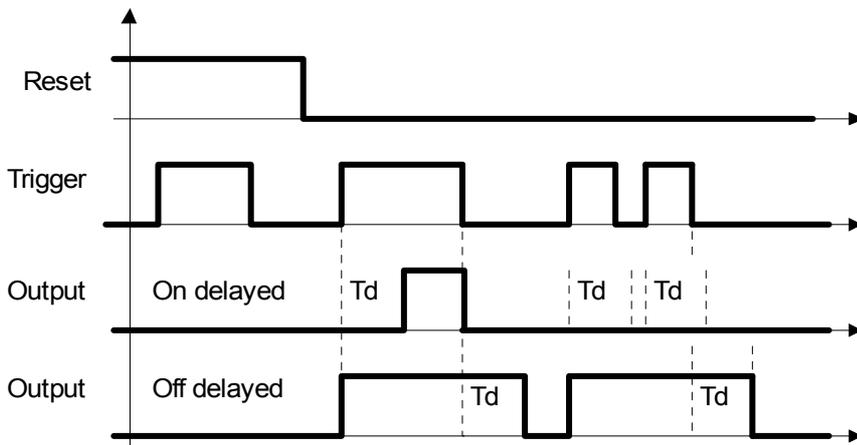
13.8.11 Verzögerter Monoflop

Dieser Monoflop weist einen flankengetriggerten Triggereingang und einen asynchronen Reseteingang auf. Im Gegensatz zum nicht verzögerten Monoflop wird keine absolute Ein-Zeit spezifiziert. Die Ein-Zeit wird bestimmt durch die Länge des Tiggerimpulses und einer zusätzlichen Verzögerungszeit T_d . Die genauen Zeitpunkte des Setzens und Zurücksetzens des Ausganges hängen von der *Aus-Verzögert*-Option ab:

Optionen:

Wenn *Aus-Verzögert* ausgeschaltet ist, wird der Ausgang um die Zeit T_d nach der positiven Triggerflanke gesetzt und bei der negativen Triggerflanke wieder zurückgesetzt. Wenn der Triggerimpuls kürzer als die spezifizierte Verzögerungszeit T_d ist, bleibt der Ausgang auf 0.

Wenn *Aus-Verzögert* eingeschaltet ist, wird der Ausgang sofort bei der positiven Triggerflanke gesetzt und um die Verzögerungszeit T_d nach der negativen Triggerflanke wieder zurückgesetzt. Wenn der Triggereingang vor Ende der Verzögerungszeit wieder gesetzt wird, wird der 1 Zustand am Ausgang erst T_d nach dem nächsten negativen Triggerübergang zurückgesetzt (Retriggierung).



13.9 Verschiedenes

13.9.1 1:2 Schalter

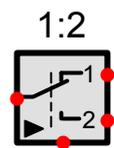
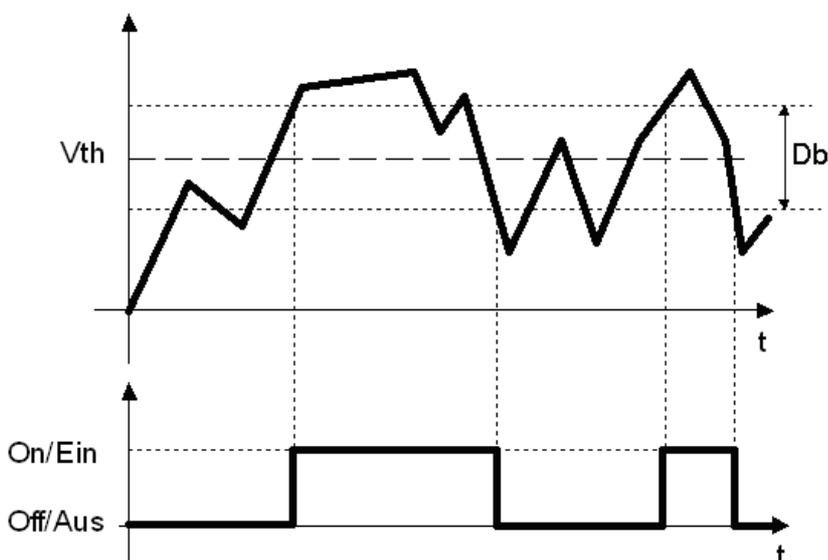
Dieses Glied ist ein 1:2 Demultiplexer. Der Steuereingang bestimmt zu welchem Ausgang der Eingang durchgeschaltet wird.

Wenn der Steuereingang tief ist (Aus-Zustand), wird der Eingang zum oberen Ausgang durchgeschaltet. Der andere Ausgang gibt 0 aus.

Wenn der Steuereingang 1 ist (Ein-Zustand), wird der Eingang zum unteren Ausgang durchgeschaltet. Der obere Ausgang gibt 0 aus.

Der Schmitt-Trigger am Steuereingang verhindert ein unkontrolliertes Hin- und Herschalten (Jitter), wenn das Steuersignal nur langsam ändert und mit Rauschen oder anderen Störungen überlagert ist. Die Schaltschwelle und die Hysterese sind einstellbar.

Der Schalter kann sowohl für analoge als auch für rein logische Signale eingesetzt werden.

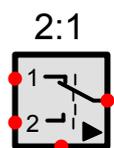


13.9.2 2:1 Schalter

Dieses Glied ist ein 2:1 Multiplexer. Der Steuereingang bestimmt, welcher Eingang auf den Ausgang durchgeschaltet wird.

Wenn der Steuereingang tief ist (Aus-Zustand), wird der obere Eingang zum Ausgang durchgeschaltet.

Wenn der Steuereingang 1 ist (Ein-Zustand), wird der untere Eingang zum Ausgang durchgeschaltet.

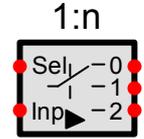


Weitere Information zum Schaltverhalten siehe 1:2 Schalter.

13.9.3 1:n Ausgangsschalter (Demultiplexer)

Der Eingang kann wahlweise auf 2 bis 50 Ausgänge geschaltet werden. Die anderen Ausgänge geben 0 aus. Die Position wird durch ein digitales Signal am Steuereingang festgelegt.

S	O0	O1	...	On-1
<0	0	0	...	0
0	1	0	...	0
1	0	1	...	0
⋮	⋮	⋮	⋮	⋮
n-1	0	0	...	1
≥n	0	0	...	0

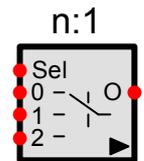


Wenn das Steuersignal < 0 oder grösser als die Anzahl Ausgänge -1 ist, werden alle Ausgänge auf 0 gesetzt.

13.9.4 n:1 Eingangsschalter (Multiplexer)

Dieser Multiplexer kann sowohl analoge als auch rein logische Signale umschalten. Er gestattet, aus 2 bis 50 Eingängen einen einzelnen auszuwählen und dem Ausgang zuzuführen. Dies ist die Ausführung eines einpoligen Schalters mit n Positionen, die über ein digitales Eingangssignal ausgewählt werden.

S	Out
<0	0
0	I0
1	I1
⋮	⋮
n-1	In-1
≥n	0



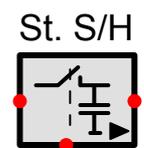
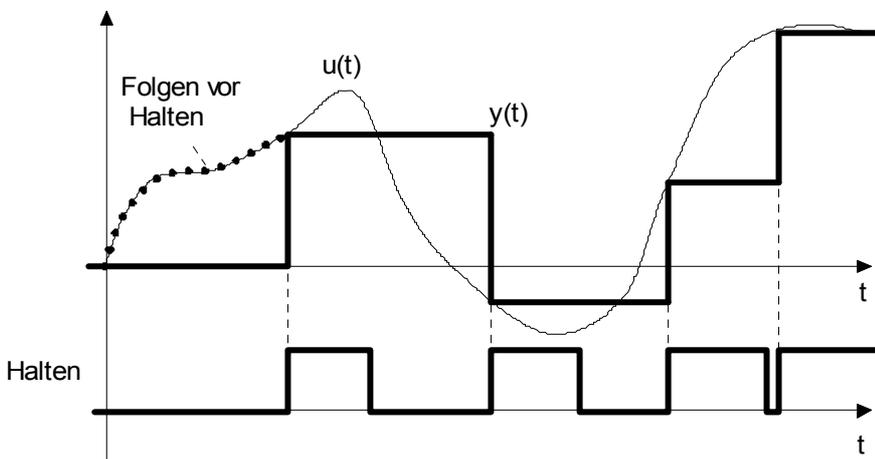
Wenn das Steuersignal < 0 oder grösser als die Anzahl Eingänge -1 ist, wird der Ausgang auf 0 gesetzt.

13.9.5 Steuerbares Abtast-/Halteglied

Dieses Glied weist einen analogen und einen logischen Eingang auf. Auf eine positive Flanke am Steuereingang (Halten) wird der momentane Eingangswert erfasst und am Ausgang zur Verfügung gestellt. Der Ausgang bleibt bis zum Ende der Simulation oder bis zum Auftreten des nächsten Haltesignals konstant.

Bis zum ersten Halten wird der Ausgang durch die *Folgen vor Halten* Option bestimmt:

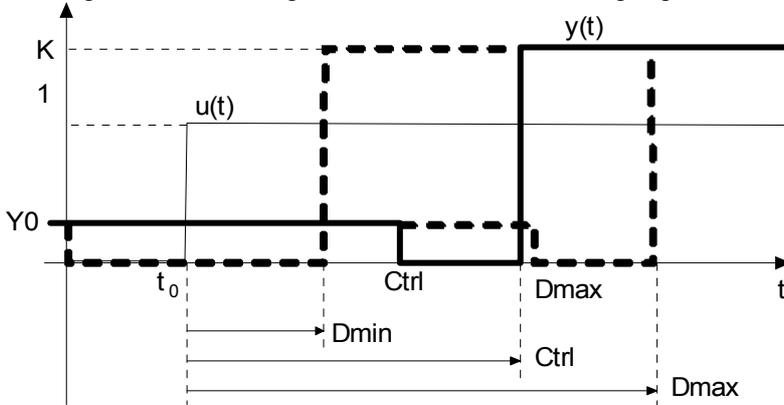
Wenn diese Option aktiviert ist, folgt der Ausgang bis zum Eintreffen des ersten Haltesignals exakt dem Eingang, andernfalls bleibt er auf 0. Nach dem ersten Halten hat diese Option keine Auswirkung mehr.



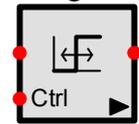
13.9.6 Steuerbares Verzögerungsglied

Die aktuelle Verzögerungszeit in [s] wird durch das analoge Steuersignal am Ctrl-Eingang bestimmt. Der gültige Verzögerungsbereich kann mit D_{min} und D_{max} eingeschränkt werden. Wenn D_{min} oder D_{max} kleiner als null sind, werden die Schranken aufgehoben. D.h. wenn $D_{min} < 0$, dann ist D_{min} effektiv = 0 und wenn $D_{max} < 0$, dann ist D_{max} effektiv = ∞ .

Die folgende Skizze zeigt die Antwort auf einen Eingangsschritt zur Zeit $t = t_0$:



Verzögerung

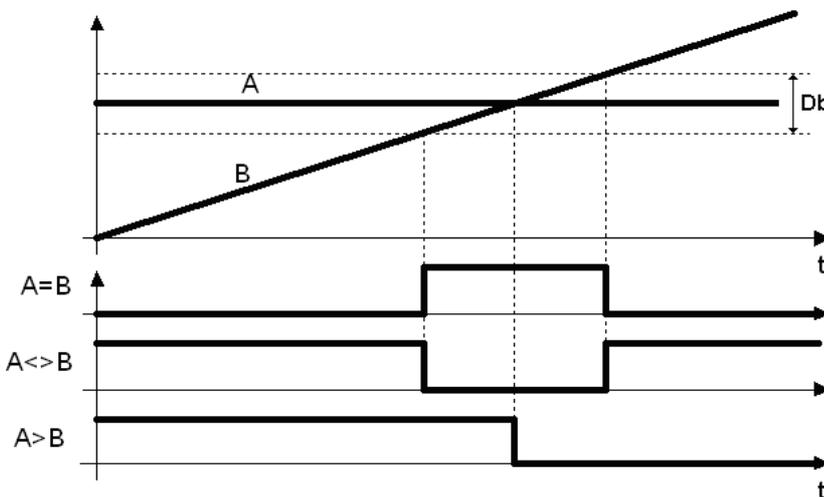


13.9.7 Vergleicher

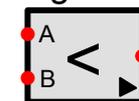
Der Vergleicher offeriert 6 verschiedene Vergleichsoperationen:

$A < B$ $A \leq B$ $A = B$ $A \geq B$ $A > B$ $A \neq B$

Für die Gleich- ($A=B$) und Ungleichoperation ($A \neq B$) wird eine Hysterese benötigt, da das exakte Vergleichen von analogen Grössen sehr kritisch ist. Sind beide Eingänge innerhalb dieses Bandes, werden sie als gleich resp. ungleich interpretiert.



Vergleicher



13.9.8 Fensterkomparator

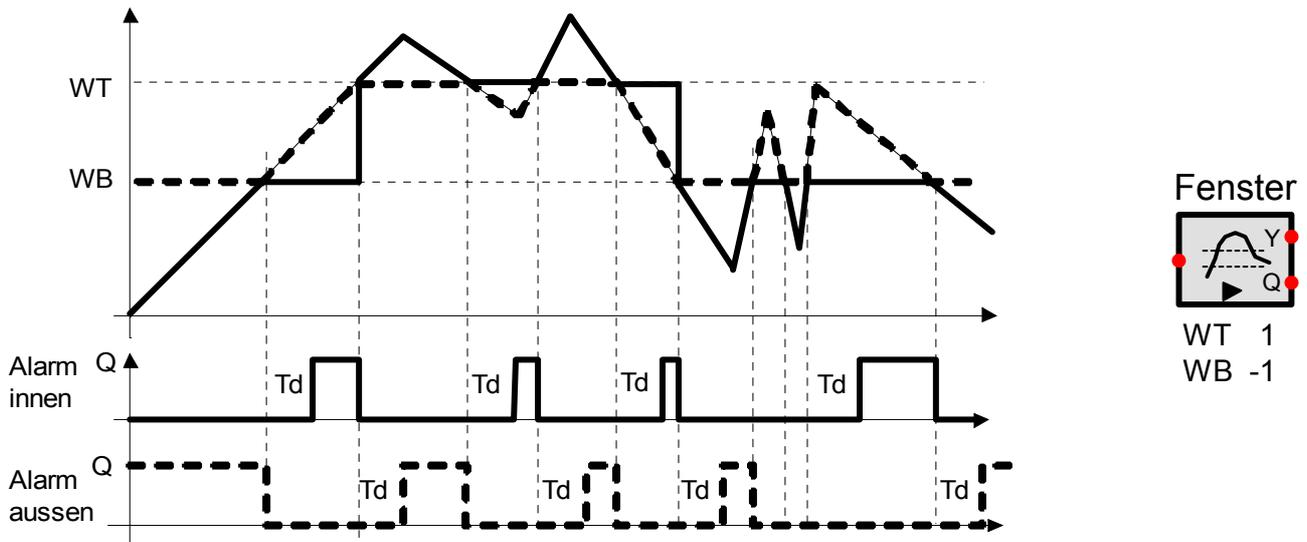
Dieses Glied testet, ob das Eingangssignal innerhalb oder ausserhalb des Fensters WB --- WT liegt. Der Zustand (innerhalb oder ausserhalb) wird durch den Alarm-Ausgang angezeigt. Der Y-Ausgang gibt das Eingangssignal unverändert aus, solange es im gültigen Bereich ist, andernfalls wird WB oder WT ausgegeben.

Die Bezeichnung des gültigen Bereiches erfolgt durch die *Alarm-Ausserhalb*-Option.

Wenn *Alarm-Ausserhalb* aktiviert ist, liegt der Alarmbereich ausserhalb WB...WT. Wenn die Eingangsgrösse zwischen WB...WT liegt, wird der Alarm-Ausgang nicht gesetzt.

Wenn *Alarm-Ausserhalb* deaktiviert ist, liegt der Alarmbereich innerhalb WB...WT. Wenn die Eingangsgrösse ausserhalb WB...WT liegt, wird der Alarm-Ausgang nicht gesetzt.

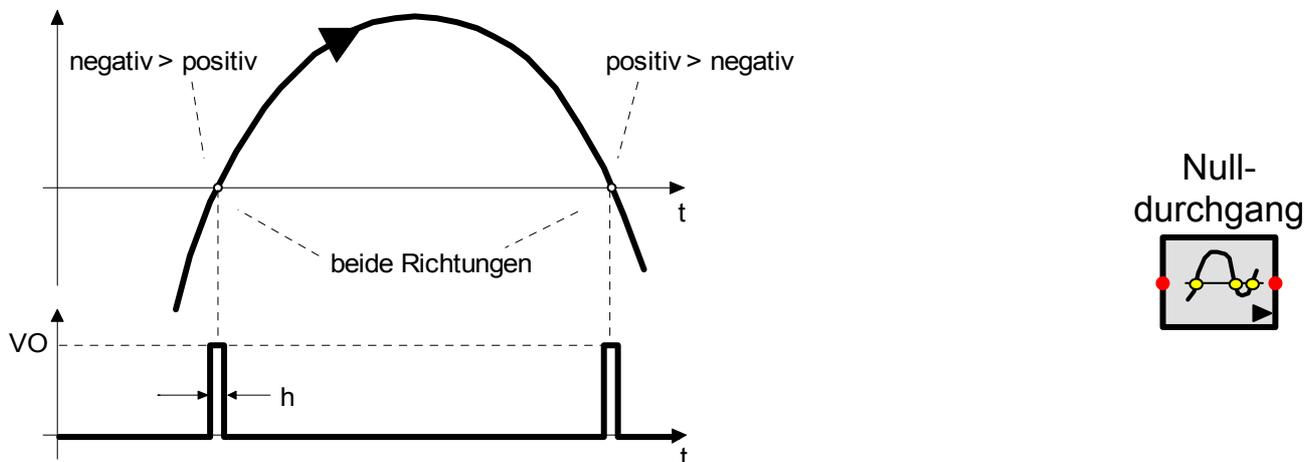
Im Weiteren kann das Aktivieren des Alarmausganges mit einer Alarmverzögerung zeitlich verzögert werden, so dass kurzzeitige Bereichsverletzungen ignoriert werden können.



13.9.9 Nulldurchgangsdetektor

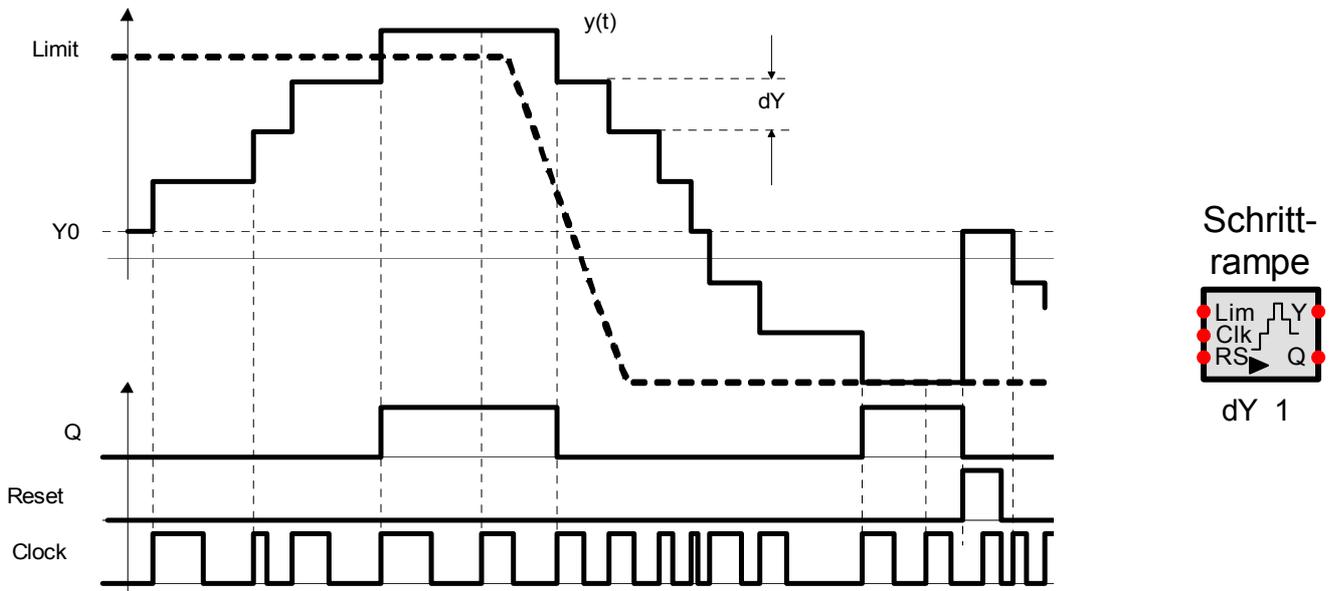
Dieses Glied gibt einen kurzen Impuls aus, wenn das Eingangssignal die Nulllinie überschreitet. Die Impulsbreite entspricht genau einem Integrationsschritt und ist nicht sichtbar im Y/t-Diagramm, wenn die Zeitauflösung zum Darstellen und Speichern der Signale (Optionen zur Zeitsimulation) länger als die Integrations-schrittweite ist.

Es kann gewählt werden, ob das Glied bei jedem Nulldurchgang, nur von der positiven zur negativen oder nur von der negativen zur positiven Seite anspricht.



13.9.10 Schrittrampe

Die Schrittrampe erzeugt am Ausgang eine treppenförmige Rampe mit fest vorgegebenem Schrittwert. Mit jeder positiven Flanke am Clockeingang wird die Rampe um eine weitere Stufe erhöht oder erniedrigt. Die Rampe startet bei einem vorgegebenen Startwert und wächst in Richtung des Grenzwerteingangssignals (Limit). Ein Resetsignal ermöglicht das asynchrone Zurücksetzen der Rampe auf den Startwert. Wenn die Rampe den Grenzwert erreicht hat, wird das Grenzsinal (Q) gesetzt und der Ausgang bleibt auf seinem letzten Wert stehen. Wird der Grenzwert verschoben, wird das Grenzsinal wieder gelöscht und die Rampe nähert sich bei jeder Clockflanke nun dem neuen Grenzwert. Durch die Umschaltung des Grenzwertes kann die Richtung der Rampe jederzeit im vollen Lauf geändert werden.



13.10 Spezialglieder

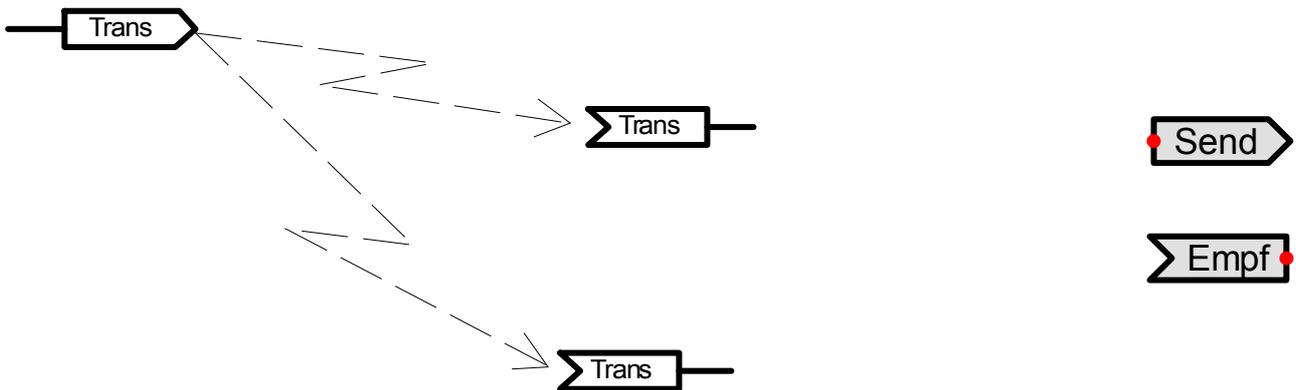
13.10.1 Sende- und Empfangsglied

Mit Sende- und Empfangsgliedern können grosse Blockdiagramme übersichtlicher gestaltet werden. Das Signal wird von einem Sender auf einen oder mehrere Empfänger übertragen, ohne dass eine Signalleitung gezogen werden muss. Die Zuordnung zwischen Sender und Empfänger erfolgt über den gemeinsamen Namen.

Es dürfen nicht zwei Sender den gleichen Namen aufweisen, weil dies einem Kurzschluss zwischen zwei Ausgängen gleichkäme.

Es ist auch möglich, Daten aus einem Anwenderblock hinaus zu senden um so auf einen Knoten zu verzichten. Dies ist jedoch nur für Testzwecke sinnvoll, weil der Sender von aussen nicht sichtbar ist.

Beispiel: 1 Sender und 2 Empfänger mit Erkennungsname Trans



14 Literaturhinweise

Deutsch

- [1] Föllinger Otto: Regelungstechnik, Hüthig-Verlag Heidelberg, 1994
- [2] Orłowski Peter: Praktische Regeltechnik, Springer-Verlag Berlin, 1994
- [3] Mann Heinz, Schiffelgen Horst, Froriep Rainer: Einführung in die Regelungstechnik, Carl Hanser-Verlag München, 1997
- [4] Wegener Adolf: Analoge Regelungstechnik, Carl Hanser-Verlag München, 1995
- [5] Bossel Hartmut: Modellbildung und Simulation, Vieweg-Verlag Wiesbaden, 1994
- [6] Schwarz Hans Rudolf: Numerische Mathematik, B. G. Teubner Stuttgart, 1993
- [7] Föllinger Otto: Lineare Abtastsysteme, Oldenburg Verlag München, 1993
- [8] Föllinger Otto: Nichtlineare Regelungen I, Oldenburg Verlag München, 1993
- [9] Föllinger Otto: Nichtlineare Regelungen II, Oldenburg Verlag München, 1993
- [10] Föllinger Otto: Laplace- und Fourier-Transformation, AEG-Telefunken, 1980
- [11] Lutz, Wendt: Taschenbuch der Regelungstechnik, Verlag Harri Deutsch
- [12] Norbert Grosse, Wolfgang Schorn, Taschenbuch der praktischen Regelungstechnik, Hanser, 2006

English

- [21] Kuo Benjamin C.: Automatic Control Systems, Prentice Hall, 1995
- [22] Kuo Benjamin C.: Digital Control Systems, Holt, Rinehart and Winston, Inc, 1980
- [23] Kailath Thomas: Linear Systems, Prentice Hall, 1980
- [24] Gibson John E.: Nonlinear Automatic Control, McGraw-Hill, 1963
- [25] Atherton D. P.: Nonlinear Control Engineering, Van Nostrand Reinhold Company London, 1975
- [26] Slotine, Jean-Jacques E. and Weiping Li: Applied Nonlinear Control, Prentice-Hall, 1991
- [27] Ogata, Katsuhiko: Modern Control Engineering, Prentice-Hall, 2002

Français

- [41] BHALY: Boucles de Régulation, études et mise au point, Kirk Editions, 1990
- [42] C.Sueur, P.VanHeeghe, P.Borne: Automatique des Systèmes Continus, Edition Technip, 1997
- [43] Yves Granjon: Automatique, Dunod, 2003