

# Einführung in Simulink



## Inhalt


1	Grundlegendes .....	2
2	Simulink starten .....	2
3	Erstellen von Simulinkmodellen .....	3
3.1	Platzieren und Verbinden der Simulinkblöcke .....	3
3.2	Eintragen der Block-Parameter .....	4
3.3	Start und Dauer der Simulation .....	5
3.4	Ergebnisse der Simulation ansehen/auswerten .....	5
3.5	Grundlegende Einstellungen zur Simulation .....	6
3.5.1	Öffnen des Fensters für die Konfiguration .....	6
3.5.2	Der numerische Lösungsalgorithmus .....	6
4	Erweiterung zu komplexeren Modellen .....	8
4.1	Ausbau des Beispiel-Modells zu einem Regelkreis .....	8
4.2	Verwendung von Untersystemen .....	10
4.3	Parameter setzen unter Matlab .....	12
5	Weitere Informationsquellen .....	13

# 1 Grundlegendes

Simulink ist ein grafikorientiertes Softwaretool zur Simulation und Analyse von linearen und nichtlinearen mathematischen Modellen. Simulink ist als Unterprogramm von Matlab implementiert und greift auf dessen numerischen Lösungsalgorithmen zu.

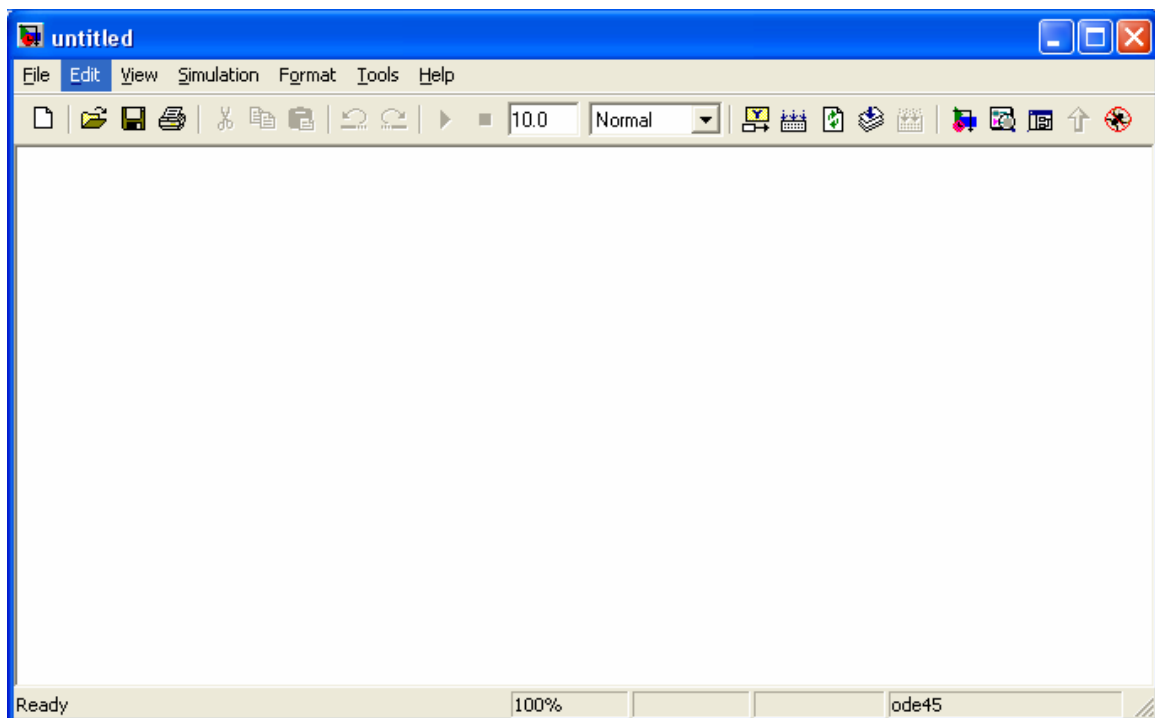
## 2 Simulink starten

Da Simulink unter Matlab läuft, setzt der Aufruf von Simulink den Start von Matlab voraus.

In der Matlab-Benutzeroberfläche kann das Fenster für ein neues Simulink-Modell über **File** → **new** → **model** erzeugt werden. Die für die Einrichtung des Modells notwendige Blockbibliothek wird anschließend über den Button  im Modellfenster geöffnet. (Bereits bestehende Simulink-Modelle können mit **File** → **open** geöffnet werden.)

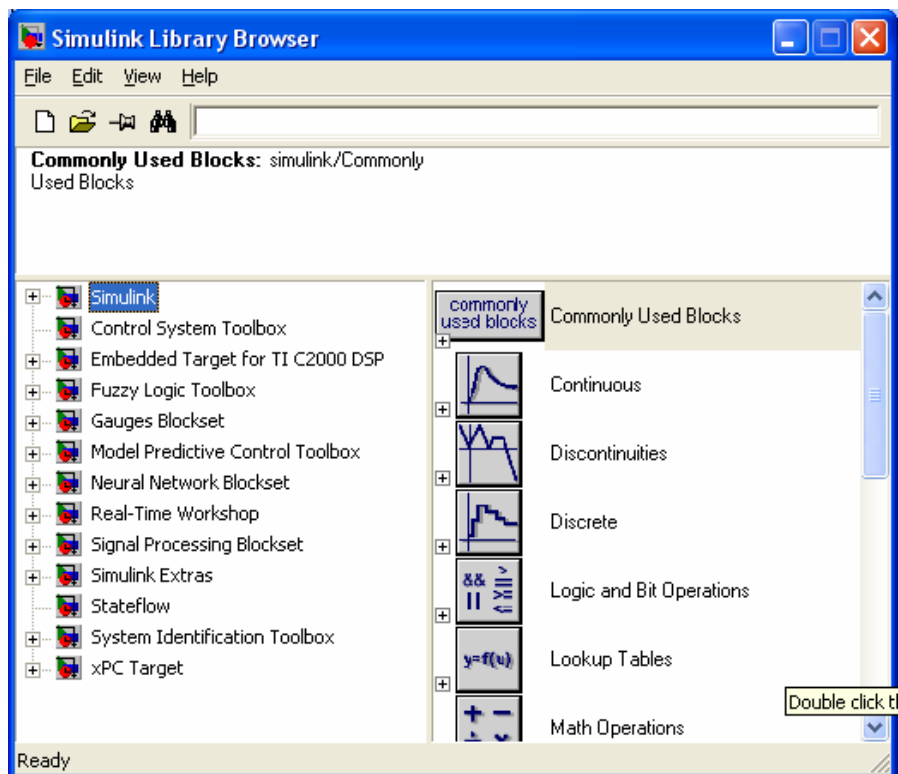
Im Folgenden sind die beiden auf die beschriebene Weise geöffneten Fenster abgebildet:

### Modellfenster:



Der Inhalt des Modellfensters kann auf die allgemein übliche Weise benannt und gespeichert werden (**File** → **save**).

## Blockbibliothek:



## 3 Erstellen von Simulinkmodellen

Im Folgenden soll anhand eines einfachen Beispiels aus der Regelungstechnik die grundlegende Vorgehensweise bei der Erstellung von Simulinkmodellen demonstriert werden.

### 3.1 Platzieren und Verbinden der Simulinkblöcke

Durch Anklicken (linke Maustaste) und Herüberziehen von Blöcken aus der Block-Bibliothek in das Modellfenster kann ein gewünschtes Modell zusammengestellt werden. Für eine erste Minimalversion eines [Beispiel-Modells](#) werden hier die drei Blöcke

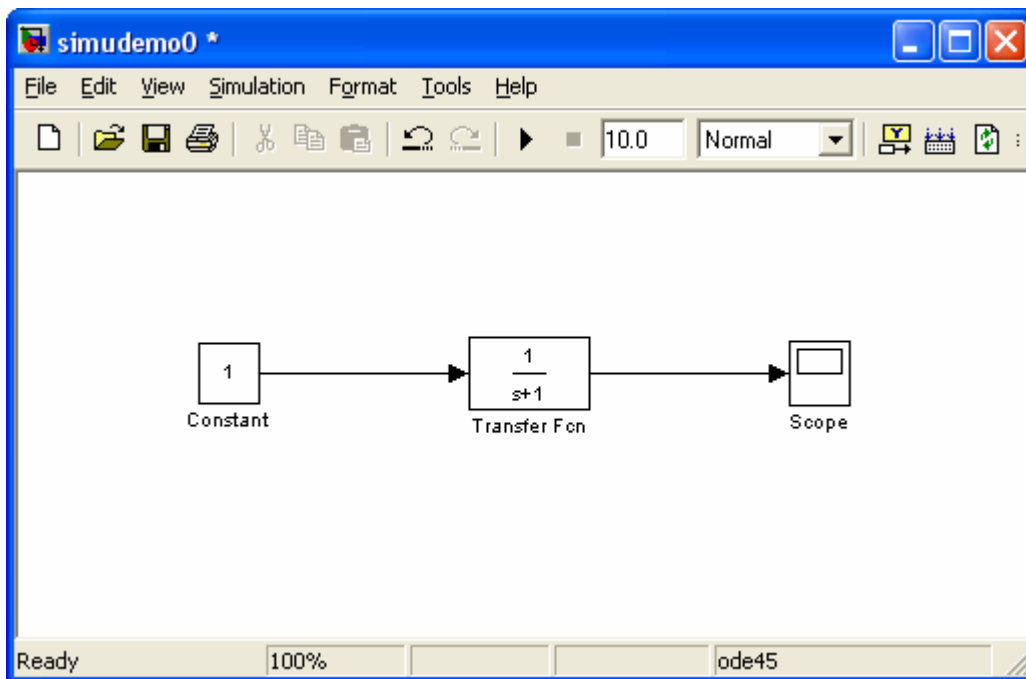
Simulink → Sources → Constant

Simulink → Continuous → Transfer Fcn („Übertragungsfunktion“)

Simulink → Sinks → Scope

in das Modellfenster eingefügt.

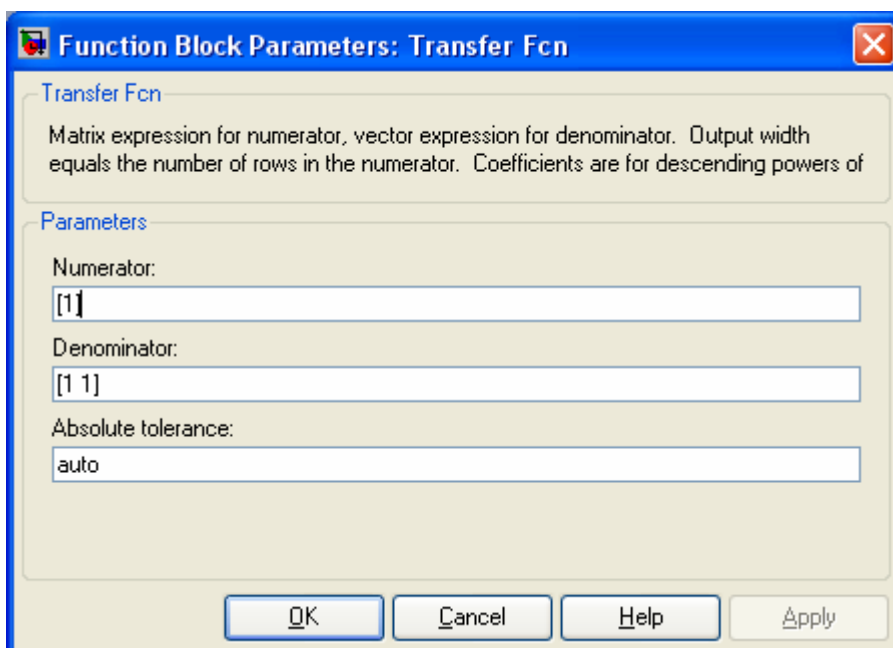
Die einzelnen Blöcke müssen anschließend miteinander verbunden werden, so dass sich der gewünschte Signalfluss ergibt. Dies geschieht, indem man bei gedrückter linker Maustaste vom Ausgang (Eingang) des einen Blocks zum Eingang (Ausgang) des anderen Blocks zieht. Es ergibt sich in dem hier betrachteten Fall das untenstehende Bild des Modellfensters.



(Modellfenster mit eingefügten Blöcken)

### 3.2 Eintragen der Block-Parameter

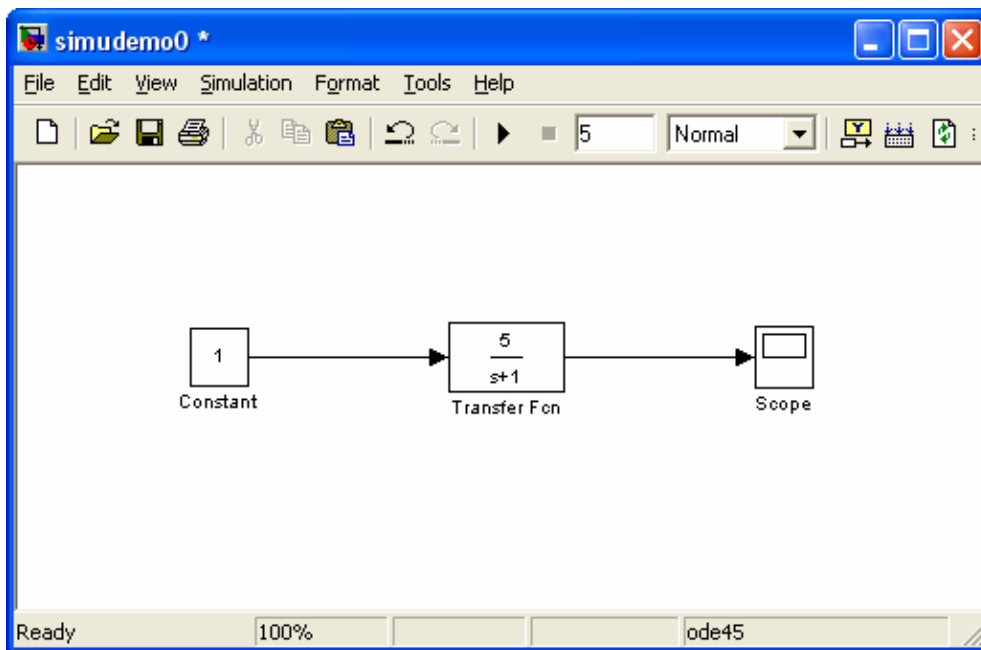
Die Konfigurierung der einzelnen Blöcke geschieht über entsprechende Einstellung ihrer Parameter. Dies wird in einem Fenster durchgeführt, das man entweder durch einen Doppelklick auf einen Block oder durch einen Klick mit der rechten Maustaste und nachfolgende Auswahl der Option „Blockname“ Parameters ... öffnen kann. Beispielsweise für den Block *Transfer Fcn* hat man so Einfluss auf drei Parameter:





Die Einträge können als Text editiert werden. Eine ausführliche Hilfe zur Bedeutung der einzelnen Blockparameter erhält man über den Button [Help](#).

Speziell bei diesem Block können unter Numerator und Denominator jeweils die Koeffizienten des Zähler- bzw. des Nennerpolynoms editiert werden. Würde man zum Beispiel den Numerator zu [3 4] und den Denominator zu [1 5 2] ändern, so ergäbe sich die neue Übertragungsfunktion  $\frac{3s + 4}{s^2 + 5s + 2}$ .

Der Einfachheit halber soll für die weitere Arbeit lediglich der Numerator zu [5] geändert werden. Die Anzeige des Blocks ändert sich im Modellfenster entsprechend:



### 3.3 Start und Dauer der Simulation


Die Simulation des erstellten Modells wird mit dem Button  gestartet. Die Simulationsdauer wird dabei durch den Zahlenwert (in Sekunden) bestimmt, der im Modellfenster oben, ungefähr in der Mitte der Werkzeugleiste, eingetragen ist. Soll die Simulation **endlos** laufen, so ist anstelle einer Zahl die Abkürzung *inf* (infinity) einzusetzen. Angehalten wird die Simulation dann über den Button .

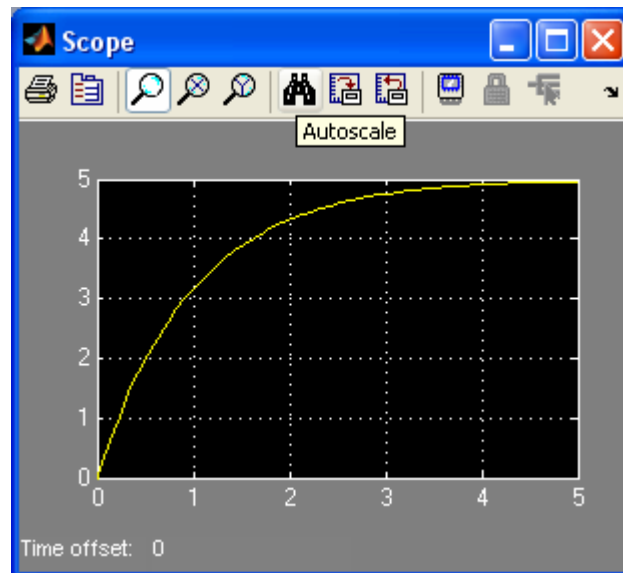
### 3.4 Ergebnisse der Simulation ansehen/auswerten.

Es gibt unter Simulink verschiedene Blöcke, die die Visualisierung von Simulationsergebnissen ermöglichen. Die am häufigsten verwendeten von diesen befinden sich in der Blockbibliothek im Ordner

Simulink → Sinks .

Neben den Blöcken zur Veranschaulichung der Messergebnisse (z.B. *Display* und *Scope*) findet man hier auch die Blöcke *To File* und *To Workspace*, mit denen die Signalverläufe entweder in einer Datei oder im Workspace von Matlab abgelegt werden können. Auf diese Weise ist nach der Simulation eine komfortable Weiterverarbeitung der Daten möglich.

Für das hier betrachtete [Beispiel](#) wird für die Veranschaulichung der Simulationsergebnisse der Scope-Block eingesetzt. Nach Start und Ausführung einer 5 Sekunden dauernden Simulation öffnet sich nach einem Doppelklick auf den Scope-Block das zugehörige Anzeigefenster. Nach Anklicken des Button  (Autoskalieren) zeigt diese Fenster den festgehaltenen Signalverlauf:



Zu sehen ist hier die Sprungantwort des PT1-Gliedes (Verstärkungsfaktor =5 und Zeitkonstante =1), das durch die Übertragungsfunktion  $\frac{5}{s+1}$  vorgegeben wurde.

Die Signalquelle *Constant* (=1) liefert zum Zeitpunkt  $t = 0$  einen Sprung, da Simulink für Zeiten kleiner Null alle Signalverläufe zu Null annimmt.

**Beachte:** Das Anzeigefenster des Scopes kann, falls gewünscht, auch vor oder während der Simulation geöffnet werden.

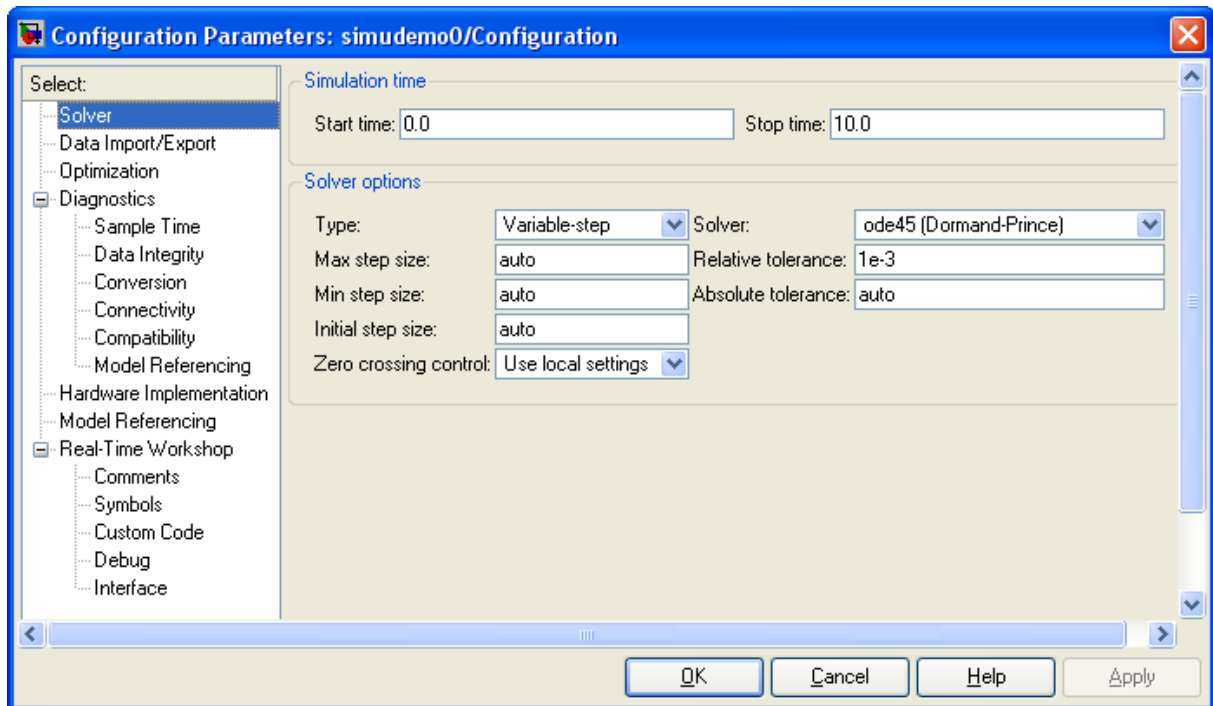
## 3.5 Grundlegende Einstellungen zur Simulation

### 3.5.1 Öffnen des Fensters für die Konfiguration

Art und Ablauf der Simulink-Simulation werden durch eine Vielzahl von Parametern und Voreinstellungen festgelegt. Diese können über den Menüpunkt **Simulation** → **Configuration Parameters ...** eingesehen und verändert werden. Es öffnet sich hierzu ein Fenster, an dessen linkem Rand sich eine Auswahlstruktur befindet, in der der gewünschte Oberpunkt angeklickt werden kann.

### 3.5.2 Der numerische Lösungsalgorithmus

Für die Simulation allgemein sehr wichtige Einstellungen sind unter dem Punkt *Solver* zu finden:



Auf der *Solver*-Seite können – neben Start- und Endzeit der Simulation – die Einstellungen für den numerischen Lösungsalgorithmus vorgenommen werden, mit dem Simulink das Differentialgleichungssystem löst, das durch das Strukturbild im Simulinkmodell repräsentiert wird. (Die Lösung der Dgl-Systeme erfolgt durch numerische Integration über die Simulationszeit.)

Standardmäßig ist in Simulink der ode-45-Algorithmus (Dormand-Prince) voreingestellt. Dieser Algorithmus hat sich für die meisten typischen Probleme als eine geeignete Wahl erwiesen und bildet einen guten Kompromiss zwischen Genauigkeit und benötigter Rechenzeit.

Der ode-45-Algorithmus ist schrittweitengesteuert (→ „*Variable-step*“). Das bedeutet, dass die Weite jedes einzelnen Zeitschrittes bei der numerischen Integration immer nur so klein gemacht wird, wie es für eine die Einhaltung der vorgegebenen Genauigkeit der Ergebnisse gerade notwendig ist. Auf diese Weise wird, insbesondere bei Modellen mit langsamer Dynamik, der Rechenaufwand für die Gesamtsimulation gering gehalten.

Weitere ausführliche Erklärungen zu den Einstellungen im Solver-Feld erhält man wieder über den *Help-Button*!

## 4 Erweiterung zu komplexeren Modellen

### 4.1 Ausbau des Beispiel-Modells zu einem Regelkreis

Der im letzten Abschnitt im Beispiel-Modell eingefügte Block *Transfer Fcn* entspricht einem linearen, zeitinvarianten (LTI-) System. Dieses kann als Modell einer physikalischen Strecke betrachtet werden. Eine typische Aufgabe der Regelungstechnik ist es, das Übertragungsverhalten einer solchen Strecke je nach Wunsch oder Vorgabe zu verändern bzw. zu verbessern. Wichtige Ziele sind hierbei oft eine schnelle Dynamik sowie stationäre Genauigkeit des geregelten Systems.

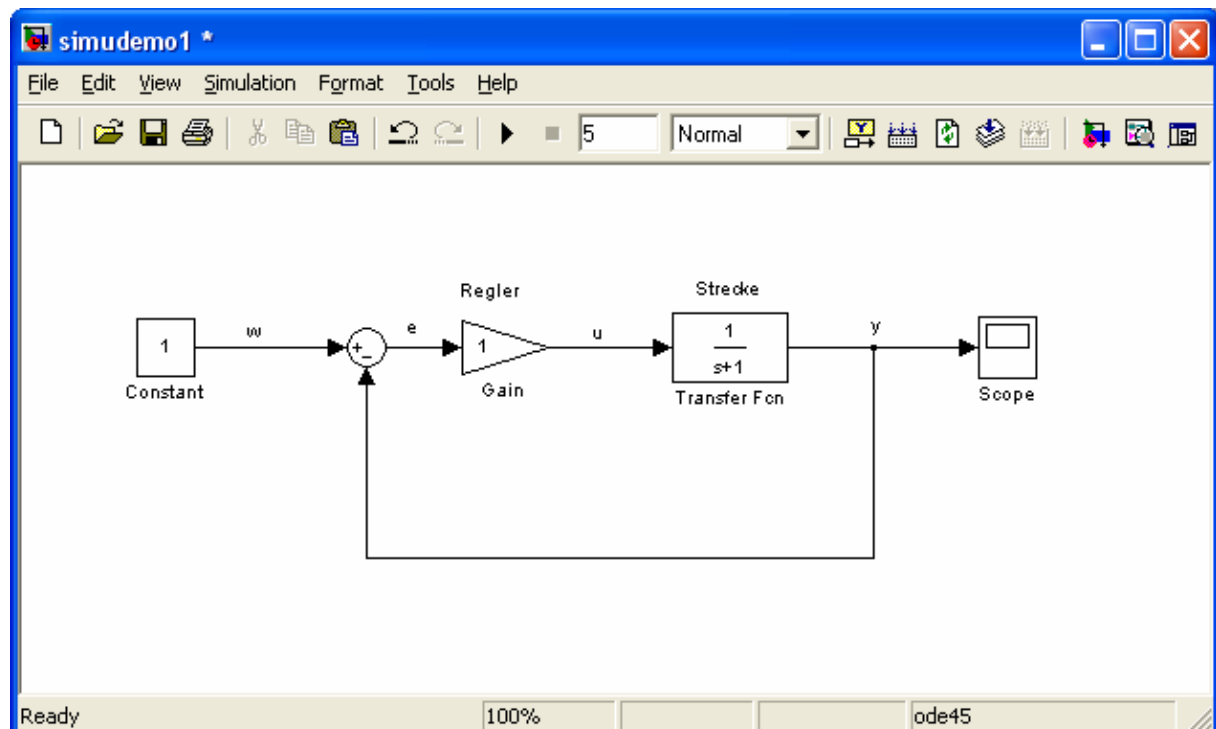
Der klassische Ansatz hierzu ist der Einbau einer Signalfückführung der Strecken-Ausgangsgröße, so dass sich ein geschlossener Regelkreis ergibt. Durch ein Regelglied, das vor der Strecke eingefügt wird, kann dann das Verhalten des Regelkreises gezielt beeinflusst werden. Im einfachsten Fall kann ein solches Regelglied ein Proportionalglied sein.

Um einen derartigen Regelkreis im hier betrachteten [Beispiel-Modell](#) einzurichten, müssen die zusätzlichen Blöcke

Simulink → Math Operations → Sum

Simulink → Math Operations → Gain

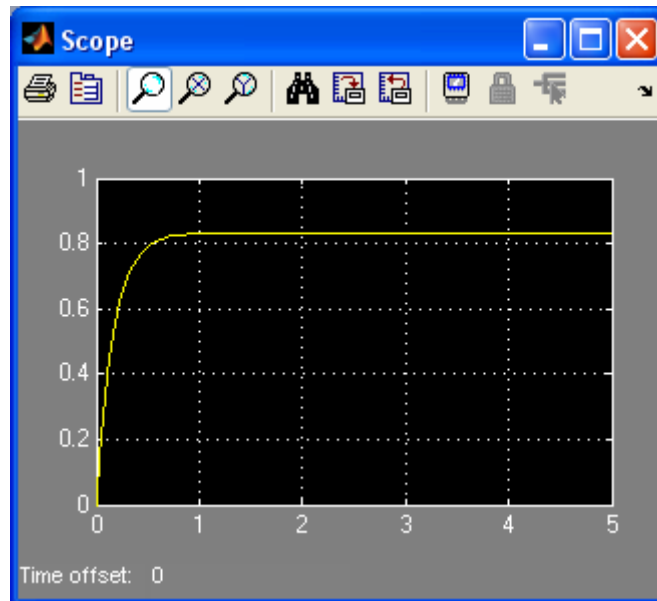
eingefügt werden. Soweit notwendig können die Signalfusslinien zwischen den bereits bestehenden Blöcken entfernt werden, indem sie zuerst durch links-Mausklick ausgewählt und anschließend mit `delete` gelöscht werden. Nach entsprechender Beschriftung der Blöcke und der Signallinien entsteht so das folgende Modell:



**Tipp:** Zusätzlicher Text kann im Modellfenster an beliebiger Stelle durch einen Maus-Doppelklick und anschließende Eingabe der gewünschten Zeichenkette eingefügt werden. Bestehender Text kann immer durch Anklicken ausgewählt und bearbeitet werden.

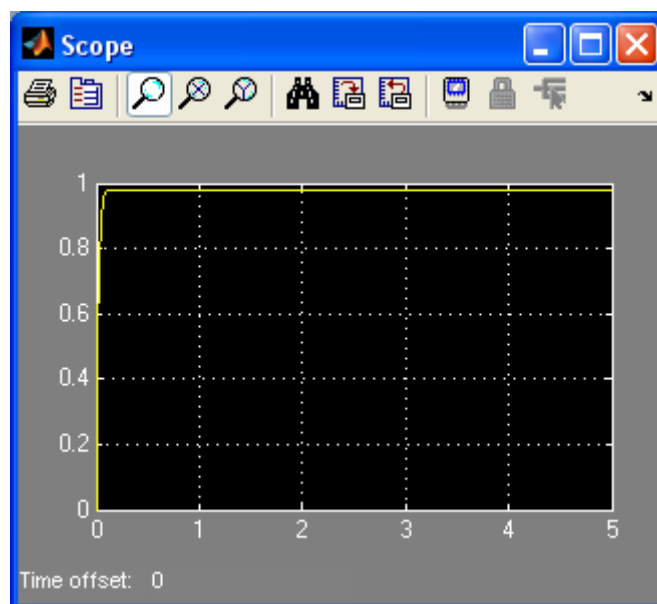


Für die gewählte Reglerverstärkung Eins (im *Gain*-Block) sieht der simulierte Verlauf von  $y$  wie folgt aus:



Das Systemverhalten ist – verglichen mit dem Verhalten der nicht rückgekoppelten Strecke (s. Abschnitt 3.4) – etwas schneller geworden. Der stationäre Wert des Ausgangssignals stellt sich aber noch recht weit unterhalb des durch die Führungsgröße vorgegebenen Wertes (Eins) ein. Es liegt schlechte stationäre Genauigkeit vor.

Wie aus den Grundlagenvorlesungen bekannt, kann hier Abhilfe geschaffen werden durch Erhöhung des Regler-Verstärkungsfaktors. Für den Wert 50 ergibt sich beispielsweise folgende Sprungantwort:



Das Gesamtsystem ist jetzt noch deutlich schneller geworden und der Führungswert Eins wird nahezu erreicht.

## 4.2 Verwendung von Untersystemen

In der Praxis sollen oft sehr große und komplizierte Systemmodelle unter Simulink implementiert werden. Um die so entstehenden komplexen Strukturen übersichtlich und handhabbar zu halten, ist eine Untergliederung des Gesamtsystems in Untersysteme („Subsystems“) notwendig. Den hierzu geeigneten Block findet man in der Simulink-Blockbibliothek unter

Simulink → Ports & Subsystems → Subsystem

Diesen Block kann man wie gewohnt in das Modellfenster ziehen. Dort kann es mit Doppelklick geöffnet und mit einer Blockstruktur gefüllt werden.

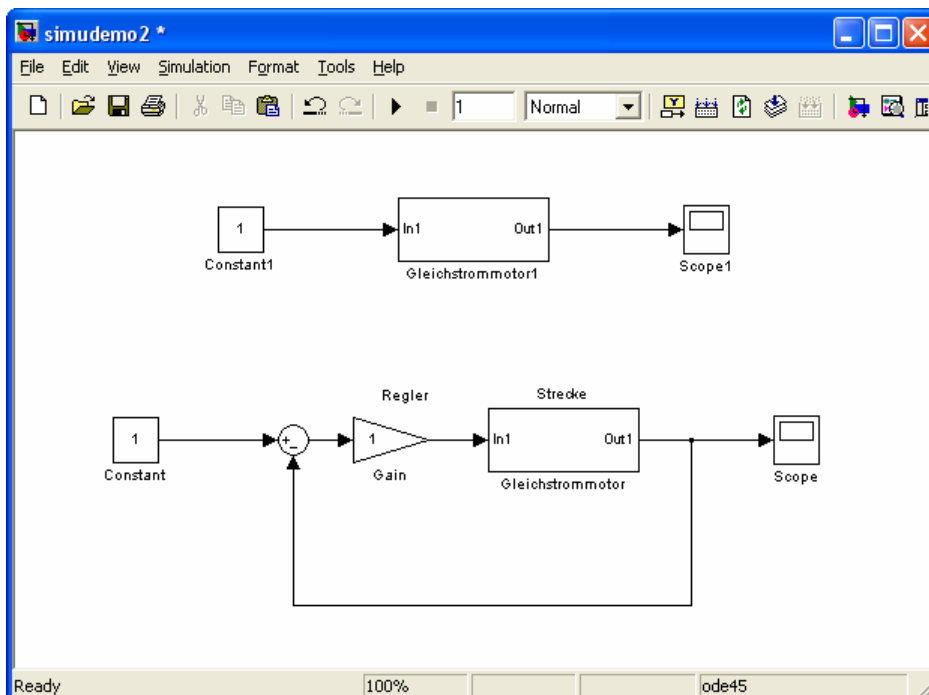
Für die Kommunikation des Untersystems mit dem übergeordneten Gesamtmodell sind die *in-* und *out-*Blöcke zu verwenden. Sie sind in der Blockbibliothek im selben Ordner zu finden, wie das *Subsystem* selbst

Für das hier betrachtete regelungstechnische [Beispiel](#) wird der Einsatz eines Untermodells z.B. sinnvoll, wenn als Regelstrecke an Stelle des einfachen PT1-Gliedes das etwas kompliziertere Modell eines Gleichstrommotors eingesetzt wird.

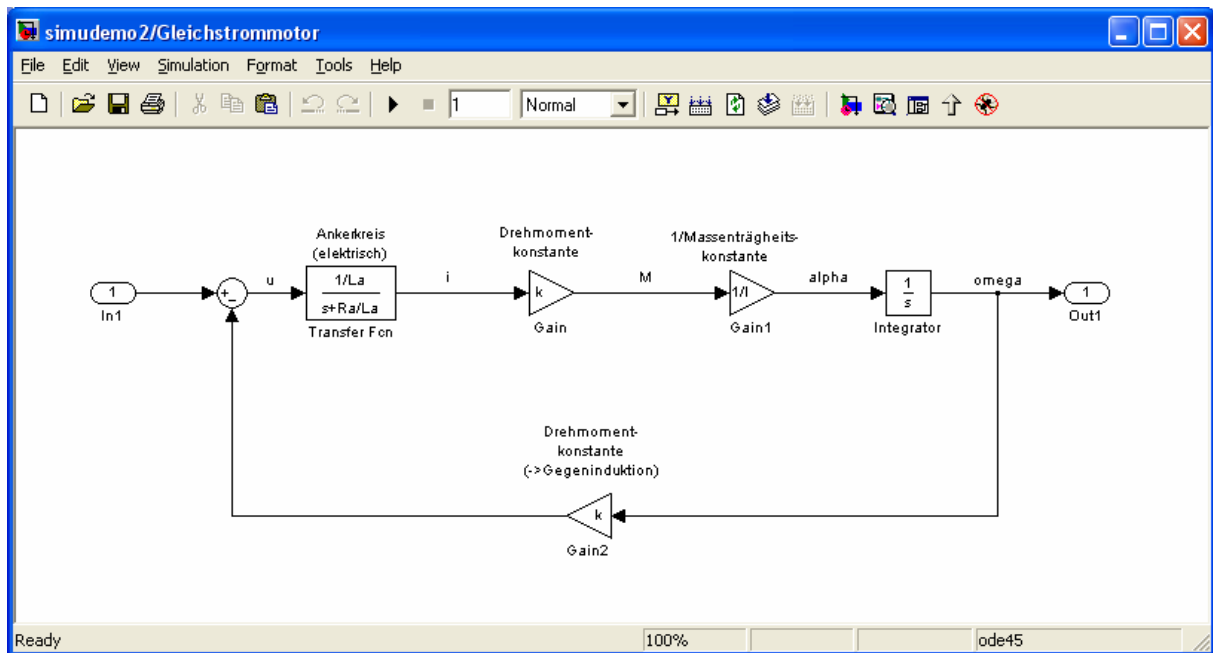
Für eine Simulation soll die zum Gleichstrommotor gehörige Simulink-Struktur nun als *Subsystem* in das oben erstellte Regelkreismodell eingesetzt werden. Außerdem wird – im selben Simulink-Modell – Das Untersystem Gleichstrommaschine noch einmal allein stehend, ohne Regelkreis, zwischen *Constant*-Block und *Scope*-Block geschaltet, so dass gleichzeitig eine Analyse des Streckenverhaltens mit und ohne Regelung möglich ist.

Es ergeben sich folgende Bilder für Gesamtmodell und Untermodell:

### Gesamtmodell:



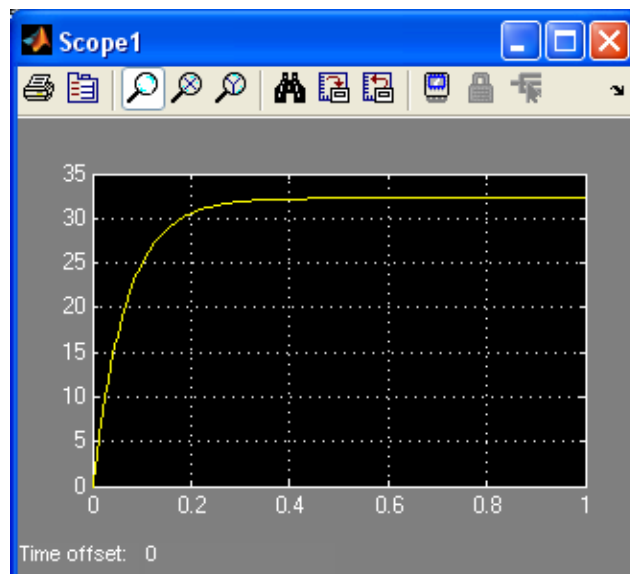
## Untermmodell (Gleichstrommotor):



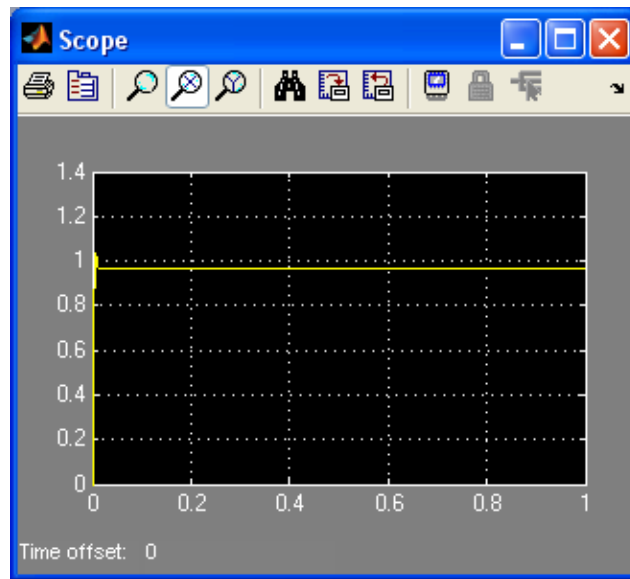
**Beachte:** Bei dem dargestellten Untermmodell sind die Parameter der einzelnen Blöcke nicht als Zahlenwerte, sondern als Variablen eingetragen. Diese Variablen werden auf die im nächsten Abschnitt beschriebene Weise mit Zahlenwerten belegt.

Vergleicht man nun wieder die Sprungantworten der Systeme ohne und mit Regelkreis (s.u.), so sieht man, dass das System mit Signalrückführung wiederum deutlich schneller ist und auch eine vergleichsweise gute stationäre Genauigkeit aufweist. Neu ist allerdings ein leichtes Überschwingen des geregelten Systems direkt nach dem Aufschalten des Sprungs.

## Sprungantwort der Gleichstrommaschine *ohne* Regelkreis:



## Sprungantwort der Gleichstrommaschine *mit* Regelkreis:



### 4.3 Parameter setzen unter Matlab

Bei größeren Simulink-Modellen ist es oft sinnvoll, bestimmte Blockparameter als Variablen im Workspace von Matlab vorzugeben. Dies ist möglich, indem der Variablenname als Parameter bei den entsprechenden Blöcke eingetragen wird und die Zahlenwertzuweisung der Variablen über Matlab vorgenommen wird.

Bei einer größeren Anzahl von Parametern sollte die Wertzuweisung mithilfe eines Matlab-Skriptes geschehen. Für die Parameter des Gleichstrommotors wurde zum Beispiel das folgende Parameterskript verwendet:

```
Editor - E:\MUENCHEN1\Lehre_M1\Studenten_Labor\MatlabSimuli...
File Edit Text Cell Tools Debug Desktop Window Help
Find text
1 % GM_Parameter.m
2 % Parametersatz einer Gleichstrommaschine
3 |
4 - Ra = 4; %Ankerwiderstand [Ohm]
5 - La = 5e-3; %Ankerinduktivität [H]
6 - k = 3.1e-2; %Drehmomentkonstante [Nm/A]
7 - I = 165e-7; %Massenträgheitsmoment [Kg m^2]
8
9
script Ln 3 Col 1 OVR
```

**Achtung:** Bevor die Simulink-Simulation des Regelkreises mit Gleichstrommaschine ausgeführt werden kann, muss dieses Skript einmal durch Eingabe seines Dateinamens (GM\_Parameter) in Matlab ausgeführt werden!

## 5 Weitere Informationsquellen

Ähnlich wie zu Matlab findet man auch weiterführende Simulink-Infos

- über den Menüpunkt **Help** → **Using Simulink** im Simulink-Modellfenster
- auf der Hersteller-Homepage *www.mathworks.de*
- auf Seiten im Internet, die man z.B mit der Suchaufforderung „Simulink“ und „Einführung“ oder „Tutorial“ finden kann.

**Beispiele** zur Anwendung von Simulink erhält man über den Menüpunkt **Help** → **Demos** im Simulink-Modellfenster.