

Ausgleichsvorgänge in elektro-mechanischen Systemen mit Maple analysieren

Grundwissen für Antriebstechnik und Mechatronik

2011 | 1. Auflage

Ergänzung Anhang B

B.4 Eingebettete Komponenten, DocumentTools und Startup-Code

Eingebettete Maple-Komponenten sind graphische Komponenten, wie Slider (Schieberegler), Buttons, Check-Boxen, Textfelder (Text Area), Plotfelder u. a. Sie bieten interaktiven Zugriff auf komfortable Maple-Funktionen für Ein- und Ausgaben und man kann sie über die Palette *Components* in eigene Dokumente einfügen. Die Beispiele in den Abschnitten B.4.3 und B.4.4 demonstrieren die Verwendung von Maple-Komponenten, von Befehlen des Pakets DocumentTools und die Nutzung von Startup-Code.

B.4.1 Das Paket DocumentTools

Das Paket ist eine Zusammenstellung von Befehlen für den Zugriff auf den Inhalt von Maple-Dokumenten, d. h. auf die Eigenschaften der eingebetteten Komponenten, auf mathematische Ausdrücke in Dokumenten bzw. Worksheets und auf die Eigenschaften von Maple-Dokumenten und Worksheets.

Do(ausdruck)

Do(name = ausdruck)

Dieser Befehl bewirkt die Auswertung von *ausdruck* mit Werten eingebetteter Komponenten bzw. die Auswertung und die Speicherung des Ergebnisses in einer Komponente oder einer Variablen. Bei Speicherung in einer Komponente ist das Ziel durch *%name* zu bezeichnen.

Das Standardattribut einer Komponente, das bei der Ausführung von **Do** abgefragt oder gesetzt wird, ist **value**. Soll ein anderes Attribut angesprochen werden, dann ist das durch die Angabe *%name(attr)* anzugeben.

Eine Typprüfung ist möglich durch die Notierung *%name::type* oder *%name(attr)::type*.

GetProperty(name, attr)

Abfrage des Attributs *attr* der eingebetteten Komponente *name*.

Beispiel: `GetProperty(Jahr, value)`

SetProperty(name, attr [, val])

Setzen des Attributs *attr* der eingebetteten Komponente *name* auf den Wert *val*. Fehlt die Angabe des Wertes *val*, dann wird das Attribut der angegebenen Komponente auf den Standardwert gesetzt.

Beispiel: `SetProperty(Slider1, value, 10)`

Weitere Befehle des Pakets sind **GetDocumentProperty** und **SetDocumentProperty** für das Abfragen bzw. Setzen der Attribute von Maple-Dokumenten (Autor, Titel usw.) und der Befehl **Retrieve** für das Lesen von Ausdrücken in Files.

B.4.2 Startup-Code

Dieser Code ist in der Startup-Region eines Maple-Dokuments gespeichert. Er wird immer dann ausgeführt, wenn das Dokument geöffnet wird und ist nur sichtbar, wenn der Startup-Code-Editor über den Menüpunkt *Edit* → *Startup Code* oder über das entsprechende Symbol in der Symbolleiste aufgerufen wird. Zum Editieren des Startup-Codes stellt der Editor kein Menü zur Verfügung. Es sind jedoch die Tastenfunktionen *Ctrl+C* für Kopieren, *Ctrl+V* für Einfügen und *Ctrl+X* für Ausschneiden anwendbar.

B.4.3 Beispiel Ortskurve

Als Beispiel dient die Darstellung der Ortskurve der Admittanz des im Bild B.1 dargestellten Schwingkreises.

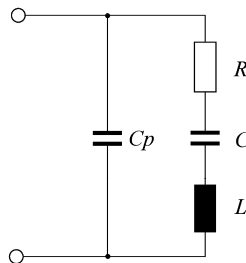


Bild B.1 Schwingkreis

Die Aufgabe wird mit drei Komponenten des Typs Schieberegler (*Slider*) zur Einstellung der Werte der Parameter R , L und C sowie einer Komponente *Plot* gelöst.

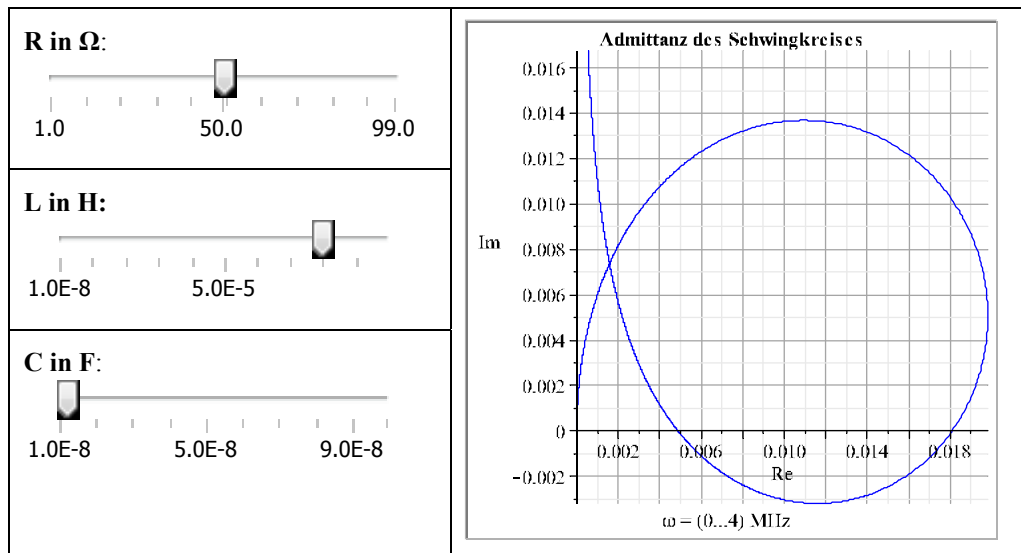


Bild B.2 Eingebettete Komponenten des Beispiels „Ortskurve“

Die graphischen Komponenten werden aus der Maple-Palette *Components* entnommen und im Worksheet platziert. Anschließend konfiguriert man sie, indem man mit der rechten Maustaste ihr Kontextmenü sichtbar macht und über *Component Properties* das Eigenschaftsfeld öffnet. Das Feld *Slider Properties* des Schiebreglers für den Widerstand R zeigt Bild B.3. Bei jeder Bewegung eines Schiebereglers soll die Prozedur *Ortskurve* aufgerufen und die graphische Darstellung im Plot-Fenster (Plot1) aktualisiert werden. Das wird durch den unter *Action When Value Changes* eingetragenen Befehl

```
Do(%Plot1 = Ortskurve(%R,%L,%C));
```

bewirkt. Das Zeichen % vor einem Namen weist auf eine Maple-Komponente hin.

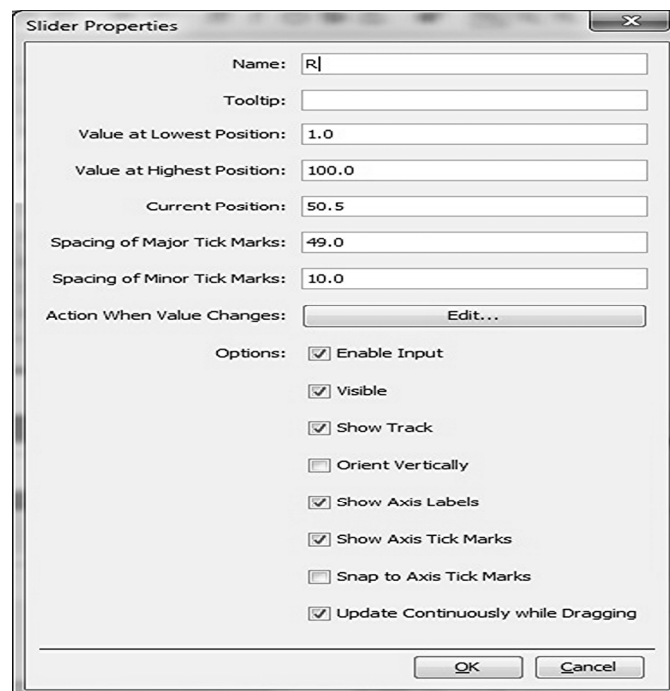


Bild B.3 Eigenschaftsfeld des Schiebreglers R

Die für das Zeichnen der Ortskurve verwendete Prozedur *Ortskurve* ist als Startup-Code definiert.

```
interface(imaginaryunit=j):
Ortskurve := proc(R, L, C) local Y, Cp;
    Cp:= 5*10^(-9);
    Y:= j*omega*Cp+1/(R+j*(omega*L-1/(omega*C)));
    plot([Re(Y), Im(Y), omega=0..4000000], gridlines, color=blue,
        title="Admittanz des Schwingkreises", titlefont=[TIMES,12,BOLD],
        caption=typeset(omega, " = (0...4) MHz, labels=["Re","Im"]);
end proc;
```

Das Eigenschaftsfeld einer Komponente *Plot* erreicht man über *Component* in deren Kontextmenü. Im vorliegenden Fall wird darin nur der Name durch *Plot1* ersetzt.

B.4.4 Beispiel Wochentagsberechnung

Das Programm berechnet für ein beliebiges Datum den entsprechenden Wochentag. Die Eingabe des Datums (Jahr, Monat, Tag) erfolgt über drei Maple-Komponenten des Typs *Text Area*. Die Wochentagsberechnung wird über den Button *Berechnung* veranlasst. Dessen Betätigung bewirkt den Aufruf der Prozedur *WoTag*, die als Startup-Code abgelegt ist. Für die Anzeige des Wochentags wird ebenfalls eine Komponente *Text Area* genutzt.

Jahr:	<input type="text" value="2011"/>	Wochentag: <input type="text" value="Freitag"/> <input type="button" value="Berechnung"/>
Monat:	<input type="text" value="7"/>	
Tag:	<input type="text" value="29"/>	

Bild B.4 Beispiel Wochentagsberechnung

Die Komponenten "Text Area"

Dialogfelder für *Text Area Properties* zeigen die Bilder B.5 und B.6. Unter *Action When Contents Change* sind in beiden Fällen keine Eintragungen erforderlich.

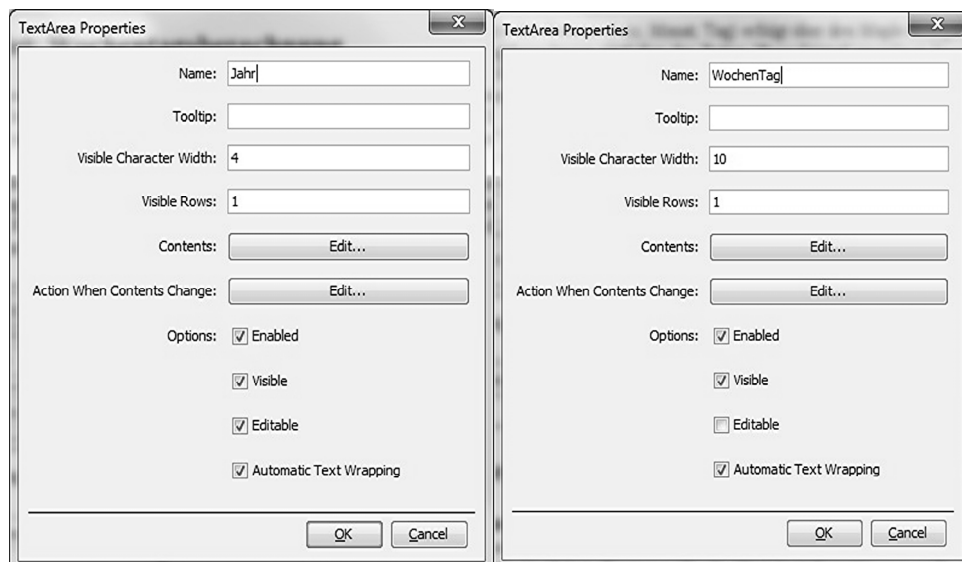


Bild B.5 Eigenschaften des Textfeldes Jahr.

Bild B.6 Eigenschaften des Textfeldes WochenTag

Die Prozedur WoTag (Wochentagsberechnung)

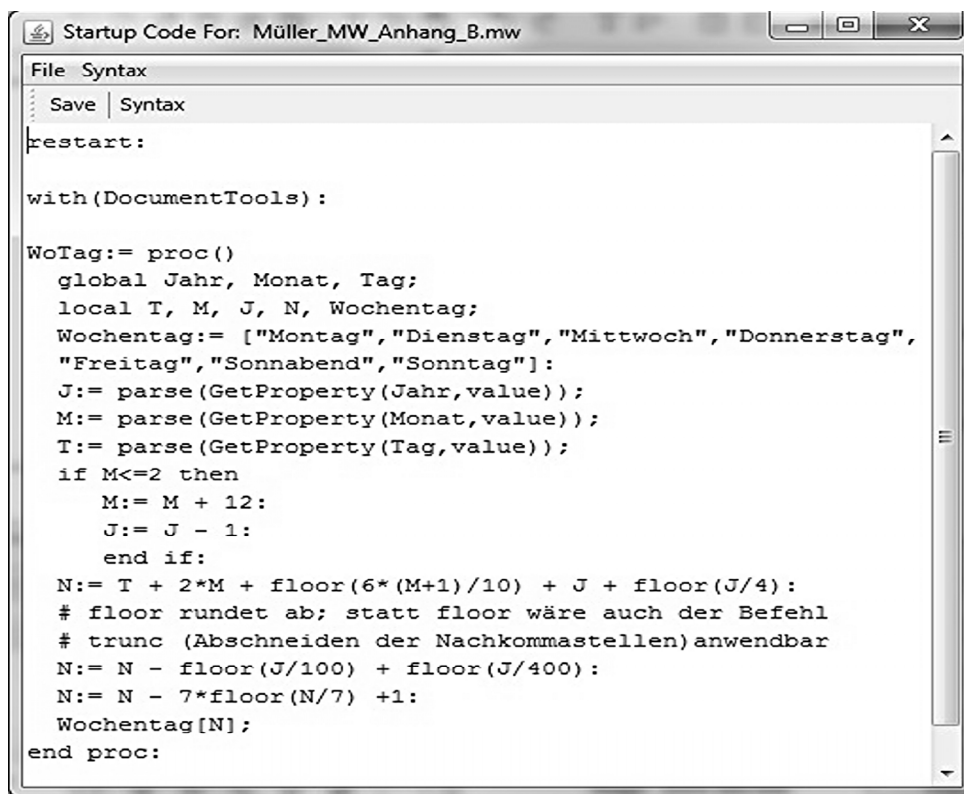
Die Prozedur ist als Startup-Code gespeichert und kann über das Menü *Edit* → *Startup Code* angezeigt bzw. geändert werden (Bild B.7). Der heutige (Gregorianische) Kalender wurde am 15. Oktober 1582 durch Papst Gregor eingeführt. Jedes vierte Jahr ist ein Schaltjahr (366 Tage), ausgenommen die nicht ganzzahlig durch 400 teilbaren Jahre.

Die Inhalte der Felder für Jahr, Monat und Tag werden mit dem Befehl **GetProperty** des Pakets **DocumentTools** ausgelesen. Weil die Informationen in diesen Feldern als String abgelegt sind, ist außerdem das Konvertieren in das Zahlenformat Integer erforderlich. Das geschieht mit dem Befehl **parse**, z. B.

`parse(GetProperty(Jahr, value)).`

GetProperty hat zwei Argumente. Das erste Argument (Jahr) bezeichnet die eingebettete Komponente, die zweite den aus dieser Komponente zu übernehmenden Inhalt.

Das Programm ist bewusst sehr einfach gehalten, verfügt also beispielsweise über keine Plausibilitätskontrollen für die Eingabewerte.



```
restart:

with(DocumentTools):

WoTag:= proc()
    global Jahr, Monat, Tag;
    local T, M, J, N, Wochentag;
    Wochentag:= ["Montag", "Dienstag", "Mittwoch", "Donnerstag",
        "Freitag", "Sonntag", "Sonntag"];
    J:= parse(GetProperty(Jahr, value));
    M:= parse(GetProperty(Monat, value));
    T:= parse(GetProperty(Tag, value));
    if M<=2 then
        M:= M + 12;
        J:= J - 1;
    end if;
    N:= T + 2*M + floor(6*(M+1)/10) + J + floor(J/4);
    # floor rundet ab; statt floor wäre auch der Befehl
    # trunc (Abschneiden der Nachkommastellen) anwendbar
    N:= N - floor(J/100) + floor(J/400);
    N:= N - 7*floor(N/7) + 1;
    Wochentag[N];
end proc;
```

Bild B.7 Startup-Programm für die Wochentagsberechnung

Der Button "Berechnung"

Die Betätigung dieses Buttons muss die Berechnung des Wochentags veranlassen. Sein Eigenschaftsfeld zeigt das Bild B.8. Die durchzuführende Aktion wird über *Action When Clicked* eingetragen. Der in diesem editierbaren Feld vorgegebene Text, der zumeist aus Kommentar besteht, wird ergänzt durch die Zeile

```
Do (%WochenTag) = WoTag();
```

Die hinzugefügte Zeile bewirkt die auszuführende Aktion: Der Komponente *WochenTag* wird der Wert zugewiesen, der beim Aufruf der Prozedur *WoTag* zurückgegeben wird. Das Zeichen % vor dem Namen weist auf eine Maple-Komponente hin.

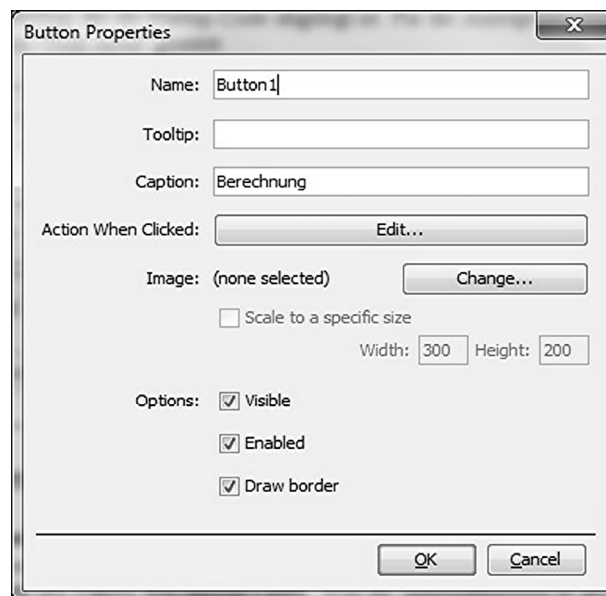


Bild B.8 Eigenschaftsfeld des Buttons „Berechnung“

Ausgleichsvorgänge in elektro-mechanischen Systemen
mit Maple analysieren

Grundwissen für Antriebstechnik und Mechatronik

Müller, R.

2011, XII, 284 S. 69 Abb. Mit zahlreichen Beispielen und
Maple-Plots., Softcover

ISBN: 978-3-8348-1217-9