

Reihen- und Parallelschaltungen

Autor & Copyright: Dipl.-Ing. Harald Nahrstedt

Version: 2016 / 2019 / 2021 / 365

Erstellungsdatum: 14.01.2024

Überarbeitung:

Quelle: Vorlesungsscript

Beschreibung:

Ein Formular soll alle Formen von Widerstandsschaltungen in einem Gleichstromkreis berechnen können.

Anwendungs-Datei: 06-10-01_Schaltungen.xlsx

1 Reihenschaltung

Werden Widerstände (Verbraucher) in Reihe geschaltet (Bild 1),

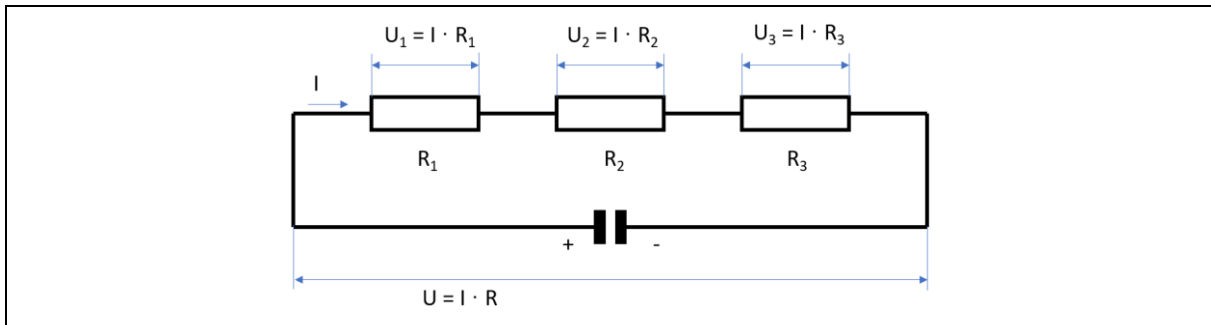


Bild 1. Reihenschaltung von Widerständen

so ist der durch die Widerstände fließende Strom konstant

$$I = I_1 = I_2 = I_3. \quad (1)$$

Die Anschlußspannung bestimmt sich aus der Summe der Teilspannungen

$$U = U_1 + U_2 + U_3. \quad (2)$$

Der Gesamtwiderstand bestimmt sich aus der Summe der Einzelwiderstände

$$R = R_1 + R_2 + R_3. \quad (3)$$

2 Parallelschaltung

Werden Widerstände (Verbraucher) parallel geschaltet (Bild 2),

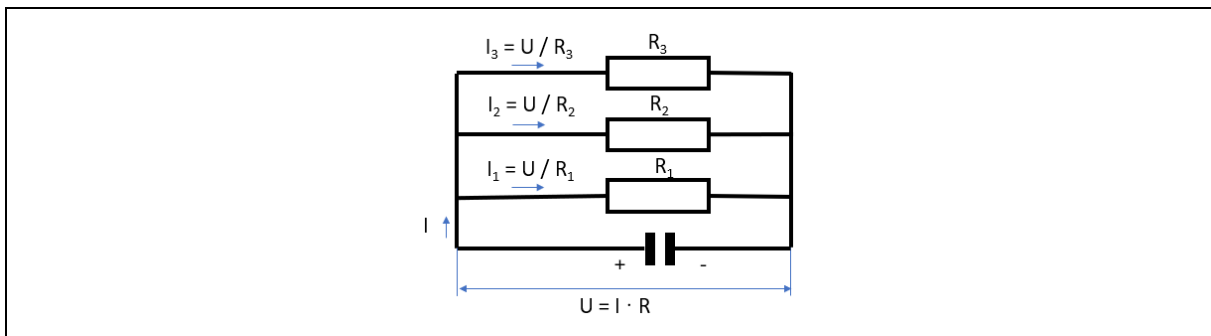


Bild 2. Parallelschaltung von Widerständen

So ist die Anschlußspannung konstant

$$U = \text{konstant}. \quad (4)$$

Der Gesamtstrom bestimmt sich aus der Summe der Teilströme

$$I = I_1 + I_2 + I_3. \quad (5)$$

Der Kehrwert des Gesamtwiderstands bestimmt sich aus der Summe der Kehrwerte der Einzelwiderstände

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}. \quad (6)$$

3 Kombinationen

In der Praxis sind die Schaltungen in einem Gleichstromkreis komplizierter und bestehen aus einer Kombination von Reihen- und Parallelschaltungen. Zur Berechnung des Gesamtwiderstands wird schrittweise vorgegangen. Zuerst wird der Widerstand der parallel geschalteten Widerstände berechnet

$$R_E = \frac{1}{\sum_{i=1}^n \frac{1}{R_i}}. \quad (7)$$

Anschließend wird der Gesamtwiderstand des gesamten Stromkreises durch Addition der Reihenwiderstände und der als Ersatzwiderstände ausgedrückten Parallelwiderstände berechnet

$$R_G = \sum_{j=1}^m R_j. \quad (7)$$

4 Schaltungsschema

Excel-Arbeitsblätter besitzen eine Tabellenstruktur. Es stellt sich die Frage, wie Schaltungen mit dieser Struktur dargestellt werden können. Die Antwort ist relativ einfach, wenn man von einer horizontalen (Reihen) und vertikalen (Parallel) Einteilung ausgeht. Legen wir die Schaltung auf ein Tabellenschema (Bild 3),

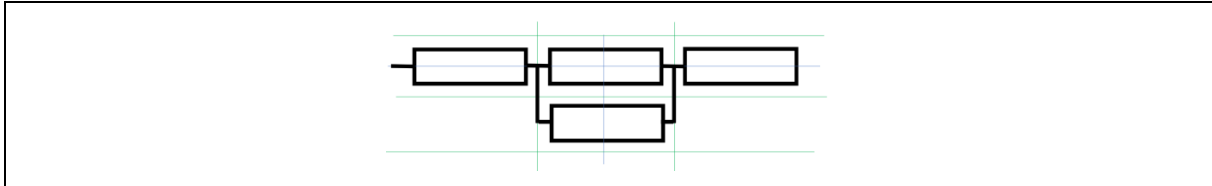


Bild 3. Schaltung auf einer Tabellenstruktur

dann können wir auch die Verbindungslinien entfernen und erhalten so die Einteilung auf einem Arbeitsblatt (Bild 4).

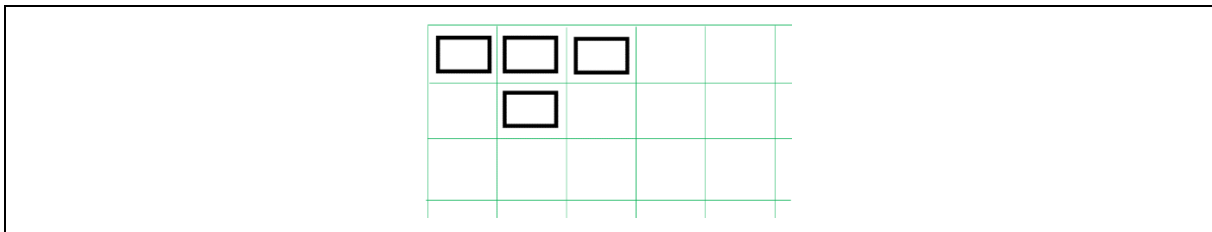


Bild 4. Belegung auf dem Arbeitsblatt

Statt der grafischen Elemente werden die Widerstandswerte eingetragen. Der Berechnungsalgorithmus muss also lediglich die belegten Zellen als Schaltelemente interpretieren. So lassen sich auch komplizierte Schaltungen darstellen (Bild 5).

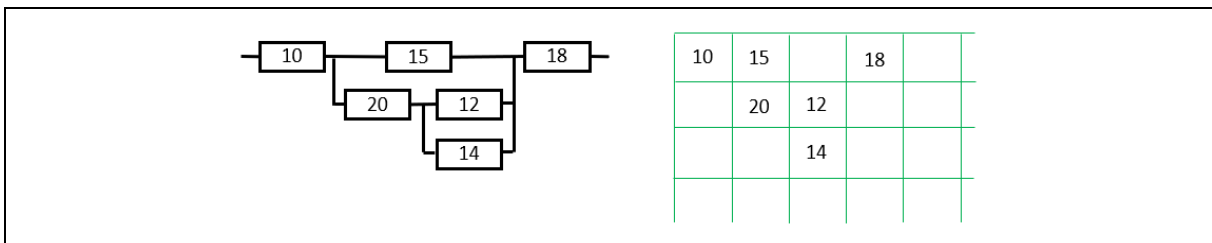


Bild 5. Beispiel einer kombinierten Schaltung

5 Algorithmus

Code-Liste 1. Berechnungsalgorithmus im Pseudocode

```

Suche die letzte belegte Zeile
Solange es zusammenhängende Werte gibt
    Bestimme den Ersatzwiderstand zusammenhängender Werte als Reihenschaltung.
    Trage den Ersatzwiderstand in die Zelle des ersten Wertes ein.
    Lösche alle anderen zusammenhängenden Werte.

Zu jedem einzelnen Wert in der Zeile
    Bestimme den Ersatzwiderstand zusammen mit dem Wert in der vorherigen Zeile
    als Parallelschaltung.
    Trage den Ersatzwiderstand in die Zelle der vorherigen Zeile ein.
    Lösche den Wert in der Zeile.
    
```

Betrachten wir diesen Algorithmus am Beispiel (Bild 5) im Arbeitsblatt.

Zeile 3 ist die letzte Zeile mit einem Eintrag (Bild 6). Da es in Zeile 3 keine zusammenhängenden Werte stehen und damit auch keine Reihenschaltung, wird für die einzelnen Einträge eine Parallelschaltung berechnet.

	A	B	C	D
1	10	15		18
2		20	12	
3			14	
4				

Bild 6. Ausgangssituation für den Algorithmus

Der Algorithmus berechnet nun den Ersatzwiderstand

$$R_E = \frac{1}{\frac{1}{14} + \frac{1}{12}} = 6,46.$$

Das Ergebnis wird in der vorherigen Zeile eingetragen und der Wert in der Zeile gelöscht (Bild 7).

	A	B	C	D
1	10	15		18
2		20	6,46	
3				

Bild 7. Ergebnis der Parallelschaltung

Nun wird die Zeile 2 zuerst wieder auf Reihenschaltung untersucht. Da die Zellen B2 und C2 Werte enthalten, bilden sie eine Reihenschaltung. Der Algorithmus berechnet den Ersatzwiderstand

$$R_E = 20 + 6,46 = 26,46.$$

Der Wert wird in die erste Zeile eingetragen und alle nachfolgenden Werte gelöscht (Bild 8).

	A	B	C	D
1	10	15		18
2		26,46		
3				

Bild 8. Ergebnis der Reihenschaltung

Da es keine weiteren Reihenschaltungen gibt, wird zum Einzelwert wieder der Wert in der vorherigen Zeile zu einer Parallelschaltung genutzt. Der Algorithmus berechnet den Ersatzwiderstand

$$R_E = \frac{1}{\frac{1}{15} + \frac{1}{26,46}} = 9,57.$$

Das Ergebnis wird in der vorherigen Zeile eingetragen und der Wert in der Zeile gelöscht (Bild 9).

	A	B	C	D
1	10	9,57		18
2				

Bild 9. Ergebnis der Parallelschaltung

In Zeile 1 werden alle Werte zu einer Reihenschaltung zusammengefasst, auch wenn sie nicht zusammenstehen. Der Algorithmus berechnet den Endwiderstand

$$R_E = 10 + 9,57 + 18 = 37,57 \Omega.$$

Damit die Anfangssituation bestehen bleibt, verwenden wir für die Berechnung eine Kopie des Arbeitsblatts.

6 Berechnungsprozeduren

Zuerst werden die Prozeduren zur Berechnung von Parallel- und Reihenschaltung erstellt. Zum Aufruf der Prozedur *Parallel* wird vorausgesetzt, dass es einen einzelnen Wert (ohne Nachbarwerte) gibt, dessen Zeile und Spalte als Parameter an die Prozedur übergeben werden. Die Prozedur liefert dann den Ersatzwiderstand und löscht die oberste Zelle,

Codeliste 2. Die Prozedur berechnet den Ersatzwiderstand einer Parallelschaltung

```
Sub Parallel(iZl As Integer, iSp As Integer)
    Dim dW1 As Double 'obere Zelle
    Dim dW2 As Double 'untere Zelle
    Dim dW As Double
```

```

dW1 = CDb1(Cells(iZl, iSp))
dW2 = CDb1(Cells(iZl - 1, iSp))
dW = 1 / dW1 + 1 / dW2
dW = 1 / dW
Cells(iZl - 1, iSp) = dW
Cells(iZl, iSp) = ""
End Sub

```

Alternativ werden bei einer Reihenschaltung alle nebeneinanderstehenden Werte direkt verwendet und der Ersatzwiderstand in der ersten Zelle eingetragen. Alle anderen Zellen werden gelöscht.

Codeliste 3. Die Prozedur berechnet den Ersatzwiderstand einer Reihenschaltung

```

Sub Reihe(iZl As Integer, iSpi As Integer, iSp As Integer)
    Dim dWert As Double
    Dim iSpn As Integer
    Dim iSpx As Integer

    dWert = CDb1(Cells(iZl, iSpi))
    iSpn = iSpi + 1
    If iZl > 1 Then
        Do While Not Cells(iZl, iSpn) = 0
            dWert = dWert + CDb1(Cells(iZl, iSpn))
            Cells(iZl, iSpn) = ""
            iSpn = iSpn + 1
        Loop
    Else
        For iSpx = iSpn To iSp
            If Not Cells(iZl, iSpx) = "" Then
                dWert = dWert + CDb1(Cells(iZl, iSpx))
                Cells(iZl, iSpx) = ""
            End If
        Next iSpx
    End If
    Cells(iZl, iSpi) = dWert
End Sub

```

Darin werden alle Werte in Zeile 1, auch die nicht zusammenhängenden, verwendet.

Während im Arbeitsblatt *Schaltung* die Ausgangswerte eingetragen werden, erzeugt die folgende Prozedur eine Kopie mit dem Namen *Berechnung*. Dabei ist zu beachten, dass ein vorhandenes Arbeitsblatt *Berechnung* zuvor gelöscht werden muss.

Codeliste 4. Die Prozedur erzeugt eine Kopie des Arbeitsblatts *Schaltung*

```

Private Sub Kopiere()
    Dim wshCalc As Worksheet
    'vorh. Berechnung löschen
    On Error Resume Next
    Worksheets("Berechnung").Delete
    On Error GoTo 0
    'neue Berechnung anlegen
    Worksheets("Schaltung").Copy After:=Worksheets("Schaltung")
    Set wshCalc = ActiveSheet
    wshCalc.Name = "Berechnung"
End Sub

```

Die Funktion *FindeSpalte* sucht in der aktuellen Zeile nach einer Zelle mit Wert und liefert deren Spaltenindex.

Codeliste 5. Die Funktion findet eine Zelle mit einem Wert in der aktuellen Spalte

```

Function FindeSpalte(iZl As Integer, _
    iSp As Integer, _
    iSpi As Integer) As Integer
    Dim iSi As Integer
    iSi = iSpi
    Do While iSi < iSp And Cells(iZl, iSi) = ""
        iSi = iSi + 1
    Loop
    If Cells(iZl, iSi) = 0 Then
        FindeSpalte = 0
    Else
        FindeSpalte = iSi
    End If
End Function

```

Bleibt abschließend noch die eigentliche Berechnungsprozedur.

Codeliste 6. Die Prozedur führt die Berechnung aus

```
Sub Berechnung()  
    Dim wshCalc As Worksheet  
    Dim iZl As Integer  
    Dim iSp As Integer  
    Dim iSpi As Integer  
  
    'Kopie mit unterdrückter Fehlermeldung  
    Application.DisplayAlerts = False  
    Call Kopiere  
    Application.DisplayAlerts = False  
  
    Set wshCalc = Worksheets("Berechnung")  
    wshCalc.Activate  
    iZl = wshCalc.UsedRange.Rows.Count  
    iSp = wshCalc.UsedRange.Columns.Count  
  
    Do While iZl > 0  
        iSpi = 1  
        'Reihen  
        Do While iSpi > 0  
            iSpi = FindeSpalte(iZl, iSp, iSpi)  
            If iSpi > 0 And iSpi < iSp And  
                Not Cells(iZl, iSpi + 1) = "" Then  
                Call Reihe(iZl, iSpi, iSp)  
            'Stop  
            Else  
                If iSpi > 0 And iSpi < iSp - 1 Then  
                    iSpi = iSpi + 1  
                Else  
                    iSpi = 0  
                End If  
            End If  
        Loop  
        'Parallel  
        iSpi = 1  
        Do While iSpi > 0  
            iSpi = FindeSpalte(iZl, iSp, iSpi)  
            If iSpi > 0 And iZl > 1 Then  
                If Not Cells(iZl, iSpi) = "" Then  
                    Call Parallel(iZl, iSpi)  
            'Stop  
            End If  
            If iSpi < iSp Then  
                iSpi = iSpi + 1  
            Else  
                iSpi = 0  
            End If  
        Else  
            iSpi = 0  
        End If  
    Loop  
    'Zeilenwechsel  
    iZl = iZl - 1  
    Loop  
    Set wshCalc = Nothing  
End Sub
```

Wer die einzelnen Schritte beobachten will, muss die Stop-Anweisungen in der Prozedur aktivieren.

7 Aufgaben

1. Erstellen Sie eine ähnliche Anwendung für Kondensatoren.
2. Erweitern Sie die Konstruktion um Spannungen und Ströme.
3. Erstellen Sie mithilfe von Normalformen aus den Schaltungsdaten eine Grafik, die den Schaltplan zeigt.
Hilfe finden Sie in meinem Buch Excel + VBA für Controller. Dort wird aus Fertigungsdaten ein Netzplan erstellt.