

# Eine kleine Einführung in SPICE

Thomas Falk, TU Dresden\*

26. September 2002

## Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>3</b>
<b>2. Was ist SPICE?</b>	<b>3</b>
<b>3. Netzliste</b>	<b>4</b>
<b>4. Modelle</b>	<b>7</b>
<b>5. Analysen</b>	<b>7</b>
<b>6. Ausgaben</b>	<b>8</b>
<b>7. Beispiel und Bedienung von SPICE</b>	<b>9</b>
<b>A. Behandlung von unabhängigen Quellen</b>	<b>13</b>
<b>B. Setzen von Anfangsbedingungen für die Transienten-Analyse</b>	<b>14</b>

---

\*falk@iee.et.tu-dresden.de

## Änderungen

- 22.10.2001 – erste Version
- 29.10.2001 – korrigierte Version (Rechtshreibung!)
- 11.12.2001 – MOSFET Parameter  $v_{t0}$  zu  $v_{to}$  geändert, Anhang zu speziellen Problemen und Fallstricken angefangen
- 15.1.2002 – kleinere Korrekturen, Beispiel erweitert
- 25.1.2002 – zur besseren Lesbarkeit im Acrobat-Reader Font gewechselt
- 11.2.2002 – Setzen von Anfangsbedingungen für die Transientenanalyse

Wenn Sie in diesem Text Fehler finden, Anregungen zur Erweiterung haben, oder Sie ihn einfach nur nützlich finden – oder auch nicht –, teilen Sie es mir bitte mit:

falk@iee.et.tu-dresden.de

## 1. Einleitung

SPICE ist der Urvater der heute verwendeten Programme zur Schaltungssimulation. Bereits in der ersten, 1972 veröffentlichten Version waren die drei Grundanalysen vorhanden und eine Syntax zur Schaltungsbeschreibung definiert, die sich bis heute in den meisten Simulatoren finden lassen.

Dieser Text soll eine kurze Einführung in die Syntax und die Grammatik der Netzlisten für SPICE geben. Das Verständnis dieser Netzlisten ist auch bei der Verwendung moderner Simulatoren mit graphischer Eingabe nützlich, da z.B. bei Modellen, Subcircuits und zur Verbindung von Eingabe und Simulator Netzlisten benutzt werden. Zusätzlich ist eine Netzliste eine kompakte Form der Darstellung einer Schaltung mit all ihren zur Simulation notwendigen Parametern.

## 2. Was ist SPICE?

SPICE ist ein Computerprogramm für die Analyse elektronischer Schaltungen. Ausgehend von einer Beschreibung der Schaltung durch den Benutzer kann SPICE den Arbeitspunkt (OP), das Frequenzverhalten (AC) und das zeitliche Verhalten (TRANS) bestimmen. Die dabei erhaltenen Daten können weiter verarbeitet werden um Kennwerte wie z.B. den Klirrfaktor oder Temperaturabhängigkeiten zu bestimmen.

Um mit SPICE eine Schaltung simulieren zu können, muß der Nutzer folgende Informationen zur Verfügung stellen:

**Schaltungsbeschreibung:** Was soll analysiert werden?

- Welche Elemente sind in der Schaltung vorhanden,
- welche Eigenschaften haben sie und
- wie sind sie verbunden

**Durchzuführende Analysen:** Welche Analysen sollen durchgeführt werden ? Z.B.

- Bestimmung des Arbeitspunktes,
- Analyse des Frequenzverhaltens oder
- des zeitlichen Verhaltens

**Auszugebende Ergebnisse:** Welche Ergebnisse sollen ausgegeben werden, z.B.

- alle Knotenspannungen,
- die Arbeitspunkte der Transistoren oder
- die Werte ihrer Kleinsignalparameter oder
- der Leistungsumsatz?

Diese Angaben werden SPICE mittels einer Datei mitgeteilt, dem so genannten *input deck*<sup>1</sup> oder *input file*.

Im folgenden soll die Form oder die Syntax und Grammatik der Eingabedatei erläutert werden. Dabei wird für Elemente in der Datei ein Proportionalfont benutzt.

Die Eingabedatei für SPICE besteht aus einer Abfolge von Zeilen. Für ihre Gestaltung gibt es drei Grundregeln:

- Die erste Zeile enthält den Titel, z.B. den Namen der zu analysierenden Schaltung, z.B. Verstärkerschaltung  
Diese Zeile wird vom Programm benutzt, um zu erstellenden Datenfiles einen Titel zu geben, darüber hinaus wird die erste Zeile nicht ausgewertet.
- Die Eingabedatei endet mit der Anweisung `.end`. Alles danachkommende wird vom Simulator nicht ausgewertet. Für manche Simulatoren ist es notwendig, nach dem `.end` noch einen Zeilenvorschub anzufügen.
- Muß eine Zeile umgebrochen werden, so ist die folgende Zeile mit einem `+` zu beginnen, um anzuzeigen, daß diese Zeile eine Fortsetzung der vorigen ist. Hier ein Beispiel für eine Spannungsquelle mit stückweise linearem Spannungsverlauf:

```
v1 2 5 AC 1 DC 5 pw1(0 0 1us 5 2us 0 4us 5 10us 0 11us 5
+ 12us 0 13us 5 15us 0)
```

Die Reihenfolge in der Datei ist von den obigen zwei Punkten abgesehen ohne Bedeutung. Um eine gewisse Übersicht zu erhalten, bietet sich aber folgende Aufteilung an:

1. Netzliste, die Beschreibung der Topologie der Schaltung
2. Modelle, die Spezifikation der Eigenschaften der komplexeren Bauelemente
3. Analysen und ihre Ausgaben

Zusätzlich kann die Datei mit Kommentaren leserlich gemacht werden. Ein Kommentar ist dabei jede Zeile, die mit einem `*` beginnt, also:

```
* Dies ist ein Kommentar
```

Groß- und Kleinschreibung wird vom Simulator ignoriert. Im folgenden sollen die Elemente entsprechend der obigen Gliederung erläutert werden.

### 3. Netzliste

Jedes Element der Schaltung wird durch eine Element-Anweisung angegeben, die den Namen, die Knoten und seine elektrischen Parameter enthält. Der Name des Elements besteht dabei aus einem Buchstaben, der den Typ des Bauelements spezifiziert und einer beliebigen Zeichenfolge,

---

<sup>1</sup>Diesen Begriff findet man vor allem noch in älterer Literatur über SPICE und er geht wohl noch auf die Eingaben für Großrechner zurück.

Komponente	Angaben in Netzliste	
	Name	Knoten und Wert
Widerstand	Rxx	K+ K- wert
Kondensator	Cxx	K+ K- wert ic
Induktivität	Lxx	K+ K- wert ic
Diode	Dxx	K+ K- modell
Bipolarer Transistor	Qxx	Kc Kb Ke modell
MOSFET	Mxx	Kd Kg Ks Kb modell
unabh. Spannungsquelle	Vxx	K+ K- qual
unabh. Stromquelle	Ixx	K+ K- qual
Spannungsgesteuerte Stromquelle	Gxx	K+ K- C+ C- wert
Spannungsgesteuerte Spannungsquelle	Exx	K+ K- C+ C- wert
Stromgesteuerte Spannungsquelle	Hxx	K+ K- Cz wert
Stromgesteuerte Stromquelle	Fxx	K+ K- Cz wert

- K+, K- – positiver bzw. negativer Knoten eines Zweipols
- Kc, Kb Ke – Kollektor-, Basis- und Emitter-Knoten eines Bipolar-Transistors
- Kd Kg Ks Kb – Drain-, Gate-, Source- und Bulk-Knoten eines MOSFETs
- C+, C- – positiver und negativer Steuerknoten einer spannungsgesteuerten Quelle
- Cz – Name einer Quelle, durch die der steuernde Strom eines stromgesteuerten Elements fließt
- wert – numerischer Wert eines Elements, z.B. des Wertes eines Widerstands
- modell – Angabe eines Modells für ein Halbleiter-Bauelement
- qual – Angaben zum Verhalten einer Quelle, z.B. zu Kurvenform, Wert bei AC-Analyse und weiteren notwendigen Angaben
- ic – optionale Anfangsbedingung für Kondensatorspannungen und Spulenströme in der Form  $i_{c=3}$

*Tabelle 1: Schaltungselemente*

z.B. M1a für einen MOSFET mit den Namenskürzel 1a. Die Knoten werden durch ganze Zahlen<sup>2</sup> angegeben, die nicht aufeinanderfolgen müssen. Die Masse der Schaltung muß die Nummer 0 erhalten, ein Masseknoten muß in der Schaltung immer vorhanden sein. In der Tabelle 1 wird eine (unvollständige) Übersicht über die am häufigsten verwendeten Elemente gegeben.

Die Werte können mit Zehnerpotenzen als z.B.  $4,3 \cdot 10^6$  oder mit den in Tabelle 2 aufgelisteten Größenbezeichnern angegeben werden.

Für Quellen gibt es eine Unzahl von unabhängigen und gesteuerten Typen, deren grundsätzliche Syntax mit in Tabelle 1 angegeben ist. Die Angaben qual spezifizieren weiter das Verhalten der Quellen. Die wichtigsten Varianten unabhängiger Quellen sind:

**Konstante Spannungsquelle:**

vxx K+ K- wert

**Sinusförmige Spannungsquelle:**

vxx K+ K- AC 1 sin(offset amplitude freq delay phi)

<sup>2</sup>In den meisten Simulatoren können statt Zahlen auch Namen wie z.B. in oder out verwendet werden, außer für die Masse, die heißt immer 0

Bez.	Name	Faktor	Bez.	Name	Faktor
T	tera	$10^{+12}$	m	milli	$10^{-3}$
G	giga	$10^{+9}$	u	micro	$10^{-6}$
Meg	mega	$10^{+6}$	n	nano	$10^{-9}$
k	kilo	$10^{+3}$	p	pico	$10^{-12}$
			f	femto	$10^{-15}$

Tabelle 2: Von SPICE erkannte Größenbezeichner

#### Pulsförmige Spannungsquelle:

```
vxx K+ K- AC 1 pulse(anfangswert highwert delay anstiegszeit
+ abfallzeit pulsebreite periode)
```

#### Stückweise lineare Spannungsquelle:

```
vxx K+ K- pwl (t0 v0 t1 v1 t2 v2 t3 v3 ...)
```

Die gleichen Versionen gelten für Stromquellen, wenn statt des v ein i verwendet wird. Bei jeder Quelle kann zusätzlich zu dem zeitlichen Verhalten ein DC- und ein AC-Wert angegeben werden, wie bei der sinusförmigen Quelle angegeben. Dieser Wert wird dann bei der DC- bzw. AC-Analyse eingesetzt. Soll eine AC-Analyse durchgeführt werden, *muß* bei mindestens einer Quelle der AC-Wert angegeben werden.

Weitere Hinweise zur Verwendung von unabhängigen Quellen sind im Anhang ab Seite [13](#) zu finden.

Sind in einer Schaltung gleiche Schaltungsstrukturen mehrfach vorhanden, so können sie zu einem Sub-Circuit zusammengefaßt werden und dann jeweils in die Schaltung eingefügt werden. Hier z.B. ein Darlington-Transistor:

```
** Definition
* Anschluesse:      c b e
.subckt darlington 1 2 3
q1 1 2 4 2n3904
q2 1 4 3 nmod1
.ends
```

```
** Verwendung:
x1 3 10 0 darlington
x2 5 11 0 darlington
```

Die im Sub-Circuit angegebenen Knoten gelten nur innerhalb der Teilschaltung, die in der .subckt-Zeile angegebenen Knoten sind die äußeren Anschlüsse. Der Knoten 0 spielt wieder eine Sonderrolle: wird er im Innern des Sub-Circuit verwendet, so ist dieser Knoten mit der äußeren Masse verbunden, als Verbindungsknoten kann 0 nicht verwendet werden.

## 4. Modelle

Halbleiterbauelemente werden in der Netzliste nur durch ihre Knoten und einen Modellbezeichner angegeben. Dieser Bezeichner wird in einer `.model`-Anweisung weiter spezifiziert:

```
.model modell typ (daten)
```

Dabei ist `modell` der in der Elementanweisung angegebene Name des Modells, `typ` ist der Typ des Elements, d.h. `npn` oder `pnp` für einen bipolaren Transistor und `nmos` oder `pmos` für einen MOSFET. Dioden haben keinen Typenbezeichner. Danach folgen in einem Klammerpaar die Daten des jeweiligen Modells. Ein Beispiel für einen bipolaren `npn`-Transistor mit dem Modellnamen `SC126` könnte sein:

```
.model SC126 npn (Is=1e-12, Bf=282)
```

und seine Verwendung in einer Netzliste ist dann wie folgt:

```
Q3 2 5 0 SC126
```

Für jeden in der Schaltung verwendeten Bauelementtyp muß eine Modell-Anweisung vorhanden sein, aber mehrere Bauelemente des gleichen Types (z.B. MOSFET's) gleichen Daten benötigen nur eine Modellanweisung. Hier eine Übersicht zu den üblichen Bauelementen:

Bauelementtyp	Modellangabe
---------------	--------------

Diode	<code>.model name d(is=... n=... usw.)</code>
-------	---

Bipolarer Trans.	<code>.model name npn (oder pnp) (is=... bf=... usw.)</code>
------------------	--

MOSFET	<code>.model name nmos (oder pmos) (vto=... kp= ... usw.)</code>
--------	--

Da Modell-Spezifikationen häufig länger als eine Zeile sind, sei hier nochmals an das + erinnert, das am Anfang einer Zeile stehen muß, wenn diese eine Fortsetzung der vorherigen ist (siehe S. 4).

Zusätzlich zur direkten Angabe der Modelle in der Netzliste können Modelle in sogenannten Libraries gesammelt werden, um als gesonderte Datei abgelegt und in verschiedenen Schaltungen verwendet zu werden. Dabei werden die verschiedenen `.model`-Anweisungen einfach hintereinander geschrieben. In dieser Form können auch bei den meisten Halbleiterherstellern Modelle ihrer Produkte über ihre Web-Seite erhalten werden.

## 5. Analysen

Nachdem die Schaltung beschrieben wurde, müssen die gewünschten Analysen angegeben werden. Es gibt drei Analysen:

**DC-Analyse:** Diese ermittelt statische Lösungen der Schaltung in Abhängigkeit einer bzw. zweier Strom oder Spannungsquellen:

```
.dc x xstart xend xstep y ystart yend ystep
```

wobei `x` und `y` die Namen unabhängiger Quellen sind, deren Werte verändert werden sollen.

**AC-Analyse:** Die AC-Analyse ermittelt das Kleinsignal-Frequenzverhalten einer Schaltung, wobei als Eingänge die Quellen mit einem AC-Wert angenommen werden:

```
.ac dec n fstart fstop
```

Die Angabe `dec` gibt an, daß die Frequenzvariation dekadisch durchgeführt werden soll, andere Möglichkeiten sind: `oct` für Veränderungen in Oktaven und `lin` für eine lineare

Variation.  $n$  ist dabei jeweils die Anzahl der Punkte pro Dekade bzw. Oktave oder insgesamt.

**Transienten-Analyse:** Die Transienten-Analyse ermittelt das Verhalten des Netzwerkes in Abhängigkeit von der Zeit. Der Befehl lautet:

```
.tran tstep tstop tstart tmax uic
```

`tstep` ist dabei eine Empfehlung für die Dauer eines Zeitschrittes, `tstop` ist die Endzeit der Simulation. `tstart` und `tmax` sind optional und gibt das Intervall an, innerhalb dessen eine Ausgabe erzeugt wird. Werden sie nicht angegeben, so gilt `tstart=0`, `tmax=tstop`.

Das Schlüsselwort `uic` steuert die Ermittlung der Anfangswerte des Netzwerkes. Näheres dazu ist im Anhang ab Seite 14 zu finden.

Zusätzlich gibt es in manchen Simulatoren noch die Möglichkeit, mit dem Befehl `.op` den Arbeitspunkt der Schaltung ausgeben zu lassen, andere Simulatoren geben diesen aber immer mit aus.

Weitere Analysen, die aber nicht in allen Simulatoren implementiert sind, sind

`.disto:` Verzerrungsanalyse

`.noise:` Rauschanalyse

`.pz:` Ermittlung von Polen und Nullstellen von Übertragungsfunktionen

`.sens:` Empfindlichkeitsanalyse

`.tf:` Übertragungsfunktion (ähnlich AC-Analyse)

## 6. Ausgaben

Die Ausgabe von Simulationsergebnissen kann mit den Anweisungen `.print` für eine numerische oder `.plot` für eine graphische Darstellung angewiesen werden. Dabei muß jeweils angegeben werden, von welcher Analyse die Ergebnisse ausgegeben werden sollen und natürlich, welche Ergebnisse, also z.B.

```
.plot dc v(2) v(5) v(3)-v(4)
```

Dabei können auch Ströme in der Form `i(zweipol)`, z.B. `i(r1)` ausgegeben werden<sup>3</sup>.

Zusätzlich ist die Angabe von algebraischen Ausdrücken möglich, um eine rechnerische Verknüpfung bzw. Auswertung der Werte vornehmen zu können, z.B. die Leistung über  $(v(4)-v(3))/i(r3)$ , wenn `r3` zwischen den Knoten 3 und 4 hängt. Weitere Nützliche Funktionen sind z.B. die Ausgabe einer Spannung nach Betrag und Phase, oder Ableitungen durch Differenzenbildung. Hinsichtlich dieser Funktionen unterscheiden sich die einzelnen Simulatoren sehr stark, so daß hier auf die jeweilige Dokumentation verwiesen werden soll.

---

<sup>3</sup>Im originalen SPICE können nur Ströme durch Spannungsquellen so ausgegeben werden. Für Ströme in anderen Zweigen muß jeweils eine Spannungsquelle mit  $U = 0$  eingefügt werden und deren Strom gemessen werden.



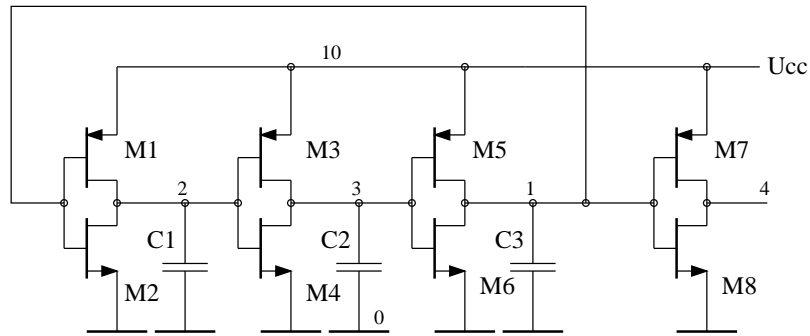


Abbildung 1: CMOS-Ring-Oszillator

## 7. Beispiel und Bedienung von SPICE

Auf Seite 11 ist die Netzliste zur Transienten-Simulation des einfachen CMOS-Ring-Oszillators in Abb. 1 gezeigt. Im folgenden soll an diesem Beispiel die Bedienung von SPICE (unter Linux) kurz erläutert werden. Die Netzliste befindet sich in der Datei `osc.cir`.

Es gibt zwei Möglichkeiten der Benutzung von SPICE:

**interaktiv:** Das Programm wartet auf Eingaben, die Ergebnisse können graphisch ausgegeben werden.

**nicht interaktiv:** SPICE arbeitet die Netzliste mit den Simulationsanweisungen ab und schreibt die Ergebnisse in Dateien.

### Interaktiver Betrieb

Hierbei wird mit SPICE im Dialog gearbeitet, d.h. man gibt SPICE eine Anweisung und SPICE gibt eine Antwort zurück. Im folgenden wird eine kurze Sitzung gezeigt, wobei die oben gezeigte Netzliste in der Datei `osc.cir` steht. Die Eingaben sind jeweils die Texte hinter dem Spice-Prompt `->`:

```
tom@marvin:~/et/spice/osc > spice
Program: Spice, version: 3f4
Date built: Fri Dec 11 21:03:04 /etc/localtime 1998
Spice 1 -> source osc.cir

Circuit: Ring-Oszillator

Spice 2 -> run
Warning: Source ip has no DC value, transient time 0 value used
Spice 3 -> plot v(1)
Spice 4 -> quit
Warning: the following plot hasn't been saved:
tranl Ring-Oszillator, Transient Analysis
```

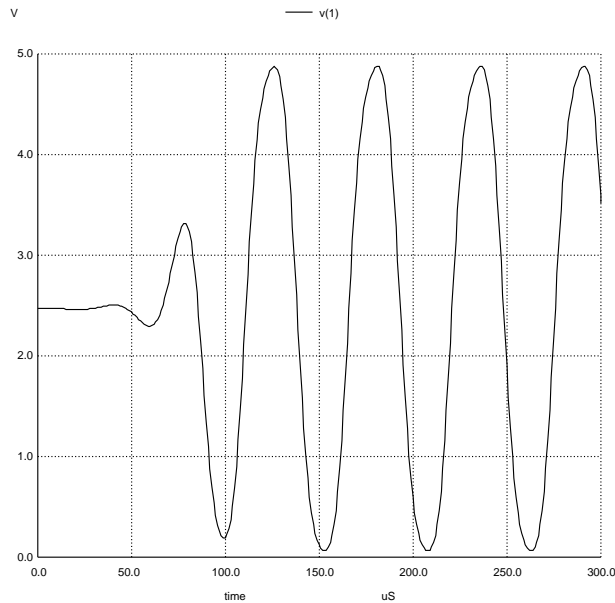


Abbildung 2: Verlauf der Spannung am Knoten 1

```
Are you sure you want to quit (yes)?
Spice-3f4 done
tom@marvin:~/et/spice/osc >
```

Nach der Eingabe des `plot`-Befehls wird ein Fenster mit dem in Abb. 2 dargestellten Verlauf geöffnet. Nach der mit `run` gestarteten Simulation können anstatt der Anzeige als Plot auch andere Auswertungen vorgenommen werden.

Eine Hilfe zur Bedienung, zu den Schaltungselementen und den Analysen kann mit dem Befehl `help` aufgerufen werden.

### Nicht interaktiver Betrieb

Spice wird mit

```
> spice -b -r osc.dat osc.cir
```

aufgerufen und erzeugt die Ergebnisdatei `osc.dat` mit folgendem Inhalt:

```
Title: Ring-Oszillator
Date: Fri Jan 29 09:34:07 1999
Plotname: Transient Analysis
Flags: real
No. Variables: 5
No. Points: 339
Command: version 3f4
```

```

Variables:
    0      time      time
    1      V(1)     voltage
    2      V(2)     voltage
    3      V(3)     voltage
    4      @m1[gm]  voltage

Values:
0          0.000000000000000e+00
  2.469203429541267e+00
  2.469203429541227e+00
  2.469203429541182e+00
  9.575896532196574e-05
1          1.000000000000000e-11
  2.469203429541266e+00
  2.469203429541226e+00
  2.469203429541182e+00
  9.575896532196578e-05
2          2.000000000000000e-11
  2.469203428042092e+00
...

```

Insgesamt folgen noch 336 solcher Wertblöcke, jedes für einen Punkt der berechneten Trajektorie. Der Aufbau des ersten Teils sollte klar sein, die Werteblocke haben folgende Bedeutung:

lfd. Nummer	Zeit
	1. Zustandsvariable
	2. Zustandsvariable
	3. Zustandsvariable

Diese Daten können für weitere Analysen in Mathematik-Programmen z.B. MatLab, eingelesen und ausgewertet werden

### Netzliste für die Transienten-Simulation des Oszillators in Abb. 1

Ring-Oszillator

```

*** Anfang der Netzwerkbeschreibung
* Betriebsspannung
vdd 10 0 5V
* 3 Inverterstufen mit Ausgangskapazitaet
x1 2 1 10 inv
c1 2 0 1n
x2 3 2 10 inv
c2 3 0 1n
x3 1 3 10 inv

```

```

c3 1 0 1n

* Ausgangsstufe
m7 1 4 10 10 pmos3 l=2u w=12.8u
m8 1 4 0 0 nmos3 l=2u w=6.4u

* Impuls zum Starten des Oszillators
ip 0 1 pulse(0 100u 100n 10n 10n 10n)

*** CMOS-Inverter in out vdd
.subckt inv      1 2 3
m1 2 1 3 3 pmos3 l=2u w=6.4u
m2 2 1 0 0 nmos3 l=2u w=3.2u
.ends

*** Ende der Netzwerkbeschreibung
* Transistormodelle
.model NMOS3 NMOS Level=3
+ Tox=40.29n Kp=51.71u Vto=0.86 Nsub=1.39E15
+ Gamma=0.26 Phi=0.62 Theta=0.05 U0=611.37 Delta=0.85
+ Cj=69.01u Mj=0.5 Pb=0.65 Cjsw=343.25p Mjsw=0.27
+ Rsh=33.42 Nfs=1.35E11 Eta=0.07 Kappa=1.4 Vmax=158k
+ Ld=0.22u Xj=0.3u
.model PMOS3 PMOS Level=3
+ Tox=42.46n Kp=19.14u Vto=-0.85 Nsub=9.11E15
+ Gamma=0.69 Phi=0.67 Theta=0.12 U0=233.84 Delta=0.96
+ Cj=310.5u Mj=0.5 Pb=0.76 Cjsw=367.38p Mjsw=0.38
+ Rsh=35.01 Nfs=3.92E11 Eta=0.06 Kappa=9.23 Vmax=225.67k
+ Ld=0.35u Xj=0.5u

* verwendete Integrationsmethode
.options method=gear
* abzuspeichernde Variablen, @m1[gm] ist die Steilheit von m1
.save tran v(1) v(2) v(3) @m1[gm]
* Simulationsbefehl
.tran 1n 300u 0 1u
.end

```

## Anhang

Im Anhang sollen spezielle Probleme erläutert werden, die oft nicht beachtet werden und zu erheblichen Fehlern führen können.

### A. Behandlung von unabhängigen Quellen

Bei den unabhängigen Quellen  $v$  und  $i$  können Werte getrennt für die AC-, DC- und Transienten-Analyse angegeben werden. Dadurch können unterschiedliche Analysen ohne Änderung der Schaltung bzw. ihrer Parameter durchgeführt werden.

So verhält sich eine Spannungsquelle mit der Anweisung

```
v1 1 0 ac 1 dc -2 pulse(1 3 1m 10n 10n 1m 5m)
```

je nach Simulation unterschiedlich (Bild 3):

**AC-Analyse:** In der AC-Analyse entspricht dieser Quelle eine sinusförmige Spannungsquelle mit 1V Amplitude und durch die AC-Analyse vorgegebener Frequenz.

**DC-Analyse:** In der DC-Analyse erzeugt diese Quelle eine negative Spannung von 2V, wenn ihr Wert nicht in der DC-Anweisung als variable Größe angegeben wurde.

**Transienten-Analyse:** Während der Analyse des Zeitverhaltens erzeugt die Quelle eine Impulsfolge mit 1 bzw. 3V-Pegel und einer Frequenz von 200Hz.

Wurden nicht alle Werte angegeben, so werden dafür durch den Simulator Default-Werte eingesetzt, das ist z.B. 0 für DC- und AC-Werte.

Die Werte können in manchen Fällen nicht unabhängig voneinander gewählt werden. Besonders kritisch ist dies bei der AC-Analyse, da vor dieser durch den Simulator der Arbeitspunkt ermittelt wird, der vom DC-Wert abhängt.

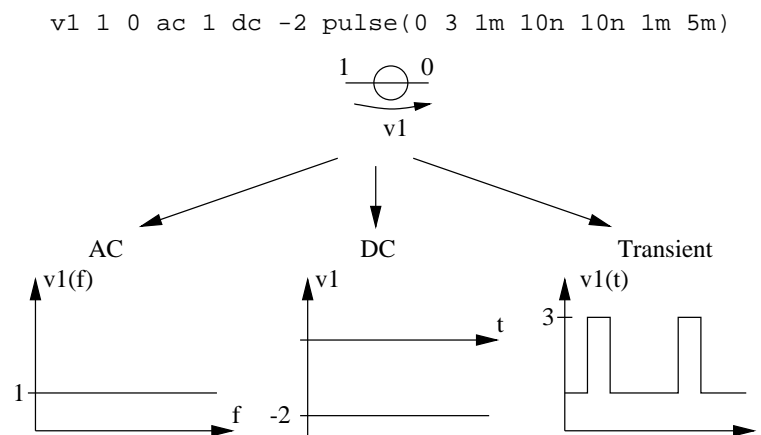


Abbildung 3: Verhalten einer Quelle in unterschiedlichen Simulationen

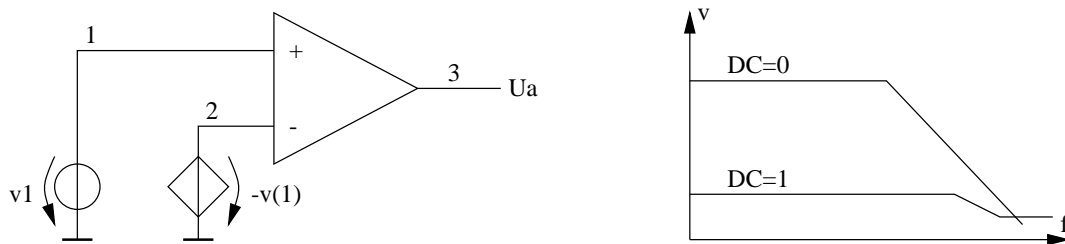


Abbildung 4: Schaltung zur simulativen Ermittlung der Differenzverstärkung eines Operationsverstärkers

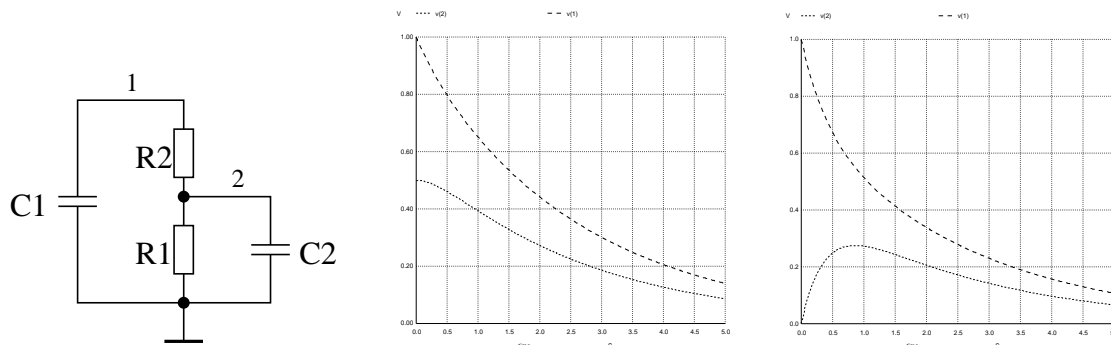


Abbildung 5: Transienten-Simulation einer einfachen RC-Schaltung mit  $U_{C1}(0) = 1V$  ohne (mitte) und mit UIC (rechts)

So soll z.B. die Schaltung in Bild 4 zur Ermittlung der Differenzverstärkung eines Operationsverstärkers dienen. Dazu soll eine AC- und eine DC-Analyse durchgeführt und deren Ergebnisse verglichen werden. Der DC-Wert der Eingangsspannung  $v1$  ist während der DC-Analyse unkritisch, da der Wert von  $v1$  in der DC-Anweisung als Veränderliche angegeben wird. Ist der DC-Wert aber ungleich Null, wird in der AC-Analyse die Verstärkung fehlerhaft ermittelt, da der OPV durch den DC-Anteil u.U. bereits weit ausgereizt wird und damit sein Arbeitspunkt völlig falsch liegt.

## B. Setzen von Anfangsbedingungen für die Transienten-Analyse

Die Option `uic` der `.trans`-Anweisung beeinflusst das Setzen der Anfangsbedingungen für eine Transienten-Simulation. Das Setzen selbst erfolgt dabei mit der `.ic`-Anweisung.

Die Netzliste für die Simulation der in Abb. 5 gezeigten einfachen RC-Schaltung lautet:

Test zur Wirkung von UIC

```
c1 1 0 1
r1 1 2 1
r2 2 0 1
c2 2 0 1
```

```
.ic v(1)=1
.tran 0.1 5 0 0.01
.end
```

Der Kondensator  $C1$  soll dabei zum Zeitpunkt  $t = 0$  auf  $U_{C1}(0) = 1V$  aufgeladen sein. Mit der angegebenen Simulationsanweisung

```
.tran 0.1 5 0 0.01
```

erhält man den in Abb. 5 in der Mitte gezeigten Verlauf. Die Spannung über dem Kondensator  $C2$  beträgt dabei entsprechend dem Spannungsteiler  $R1, R2$   $0.5V$ . Diese Spannung wurde den Simulator in einer anfänglichen DC-Analyse ermittelt.

Fügt man die `uic`-Option hinzu:

```
.tran 0.1 5 0 0.01 uic
```

so verlaufen die Lösungen entsprechend Abb. 5 links. Als Startwert für  $U_{C2}$  wurde hier durch den Simulator  $0V$  angenommen. Dieser Anfangswert setzt aber einen Strom durch  $R1$  voraus, so daß sich die Schaltung nicht im (DC-)Gleichgewicht befindet.

Zusammenfassend gilt, daß bei der Verwendung der `uic`-Option alle Spannungen über Kondensatoren und alle Ströme durch Induktivitäten vorgegeben werden müssen, wenn sich die Schaltung in einem eingeschwungenen Zustand befinden soll. Werden sie nicht angegeben, so werden sie zu Null gesetzt. Die `uic`-Option ist damit vor allem zur Untersuchung von Einschaltvorgängen und zur Fortsetzung einer Simulation ausgehend von einem vorher gespeicherten Zustand notwendig, im allgemeinen sollte sie aber nicht verwendet werden.

Abschließend eine Frage zur eigenen Untersuchung: Wie sehen die Lösungen aus, wenn man parallel zu  $C1$  eine Induktivität (z.B.  $1H$ ) schaltet?