

EINFÜHRUNG IN PSPICE

Schaltungssimulationen am PC

Eine Kurzeinführung von
Oscar Koller
im Rahmen
des Projektlabores
WS 2004/2005
TU-Berlin

1. EINFÜHRUNG.....	3
1.1. WAS IST PSPICE?.....	3
1.2. WAS STECKT DAHINTER? SPICE!	3
1.3. VORTEILE DER SCHALTUNGSSIMULATION	3
2. ARBEITEN MIT PSPICE STUDENT 9.1	4
2.1 CAPTURE	4
2.1.1. Neues Projekt anlegen.....	4
2.1.2. Arbeitsbereich.....	5
2.1.3. Erstellen des Simulationsprofils	6
2.1.4. Definition von Messpunkten (Marker).....	7
2.1.5. Fahrplan zum Umsetzen eines Schaltplans	8
2.2. PSPICE A/D	8
2.3. MODEL EDITOR – ODER WIE BESCHAFFE ICH MIR FEHLENDE BAUTEILE?	9
2.4. EINHEITEN UND GEBRÄUCHLICHE SHORTCUTS.....	10
3. BEISPIELE	11
3.1. DIODENGLEICHRICHTER (TIME DOMAIN(TRANSIENT) ZEITANALYSE)	11
3.2. TRANSISTOR AUSGANGSKENNLIENENFELD (ZWEIFACHER DC SWEEP)	12
3.3. BODEDIAGRAMM ANHAND DES VERSTÄRKERS AUS DEM 2. SEMESTER (AC SWEEP)	13
3.3.1. Bauteil-Modell-Beschaffung	13
3.3.2. Hinzufügen der Libraries (OLB).....	14
3.3.3. Aufbauen des Schaltplans	15
3.3.4. Erstellen des Simulationsprofils und einbinden der Libraries (LIB)	15
3.3.5. Setzen der Messpunkte	16
3.3.6. Simulationsergebnisse.....	17
4. TYPISCHE FEHLERQUELLEN	18
4.1. ANHAND EINES BEISPIELS.....	18
4.2. ALLGEMEINE TIPPS:	19
5. WEITERFÜHRENDE LITERATUR UND LINKS.....	20
5.1. BEZUGSQUELLEN DER STUDENTEN VERSION VON PSPICE	20
5.2. ALLGEMEINE LINKS.....	20
5.3. BAUTEIL MODELL BESCHAFFUNG	20
6. QUELLENANGABEN	20

1. Einführung

1.1. Was ist PSpice?

PSpice ist ein Schaltungssimulator von Cadence (früher Microsim, dann Orcad). Es bietet eine komplette Arbeitsumgebung mit graphischer Schaltplaneingabe (Capture oder Schematics), Signalquellen-Eingabe (Stimulus Editor), Bauteil-Editor (Model Editor) und schließlich der eigentlichen Schaltungssimulation PSpice AD. Je nach Software Paket stellt Cadence noch weitere Utilities wie ein Platinen Layouter und einen Optimizer zur Verfügung.

1.2. Was steckt dahinter? Spice!

Der Kern des Pakets, die PSpice A/D Schaltungssimulation, beruht, wie der Name bereits andeutet, auf dem Spice-Algorithmus (Simulation Program with Integrated Circuit Emphasis | Simulationsprogramm mit Schwerpunkt auf Integrierten Schaltungen). Spice kann zur Simulation analoger, digitaler und gemischt analog-digitaler Schaltungen verwendet werden. Es wurde in den siebziger Jahren an der Universität Berkeley in Californien entwickelt und ist heute zur lizenzenfreien Verwendung verfügbar. Auf dieser Basis entstanden kommerzielle Ableger wie PSpice und auch HSpice, die spezifische Erweiterungen und zusätzliche Module zur graphischen Schaltplaneingabe Ergebnisanzeige und Ablaufsteuerung enthalten. Während HSpice von Metasoft für den Entwurf integrierter Schaltungen mit mehreren Tausend Transistoren ausgelegt ist und in vielen IC-Design-Paketen als Simulator verwendet wird, ist PSpice ein im Vergleich dazu preisgünstiges und komfortabel zu bedienendes Programmsystem zum Entwurf kleiner und mittlerer Schaltungen auf PCs mit Windows als Betriebssystem.

1.3. Vorteile der Schaltungssimulation

Schaltungssimulationen sind aus der Industrie nicht mehr wegzudenken. Sie ermöglichen eine schnellere Schaltungsentwicklung, da Fehler frühzeitig erkannt und abgefangen werden können. Hieraus leitet sich ein enormer Kostenvorteil ab, der durch Einsparungen bei nicht notwendigem Labor Equipment und Bauteilen in der Forschung noch vergrößert wird. Spice stellt zudem einen langjährigen Industriestandard dar, was dazu führt, dass viele Bauteilbibliotheken direkt bei den Herstellern erhältlich sind und nicht mühsam selbständig erstellt werden müssen.

2. Arbeiten mit PSpice Student 9.1

PSpice ist als kostenlose, leicht eingeschränkte Test Version erhältlich. Im Vergleich zur etwa 10.000 Euro teuren Vollversion sind dabei Schaltpläne mit maximal 64 Knoten, 50 Bauteilen, davon 10 Transistoren und 3 Operationsverstärkern möglich.

Sämtliche meiner folgenden Ausführungen beziehen sich dabei auf die Version 9.1., die auch an den Rechnern der Labore installiert ist. Alle Schilderungen sind nur als grundlegender Einstieg gedacht – bei weiterem Interesse dienen unter 5. angeführte Links, Literaturangaben und die Programm Hilfe (F1).

Wie bereits angedeutet, gliedert sich PSpice in mehrere Teilprogramme. In diesem Referat als relevant erachtet werden:

- Capture (2.1.) zur Schaltplaneingabe
- PSpice AD (2.2.) zur Simulation des Schaltplanes (bzw. -kreises)
- Model Editor (2.3) zum Hinzufügen von Bauelementen, die nicht in der Bibliothek (Library) vorhanden sind

2.1 Capture

2.1.1. Neues Projekt anlegen

Capture dient der Schaltplaneingabe. Hier können alle Bauelemente ausgewählt, platziert, benannt und schließlich verbunden werden. Capture verwaltet die verschiedenen zugehörigen Dateien in so genannten Projekten. Im Menu unter File->New->Project kann ein neues Projekt angelegt werden. Wir verwenden hierbei ein „Analog or Mixed A/D“ Projekt und tragen einen beliebigen Namen ein.

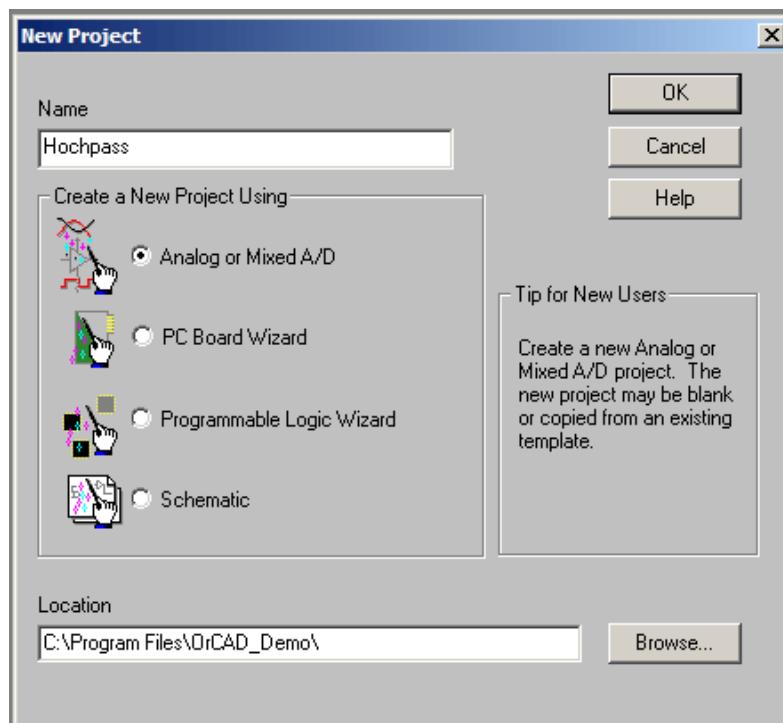


Abb. 01: New Project-Dialog

Im folgenden Fenster geben wir an, ein leeres Projekt zu erstellen:

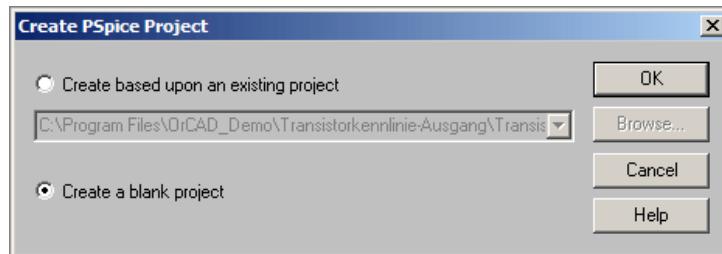


Abb. 02: Create Project-Dialog

2.1.2. Arbeitsbereich

Capture begrüßt uns nun mit dem in Abb.03 gezeigten Arbeitsbereich, der sich in zwei Elemente gliedert. Links ist der Projektmanager zu sehen, der alle am Projekt beteiligten Dateien hierarchisch darstellt und rechts davon befindet sich die eigentliche Arbeitsfläche, Schematic1 – Schaltkreis1. Nach dem Anklicken des Schematic-Fensters erscheinen auf der rechten Seite eine Reihe Symbole, die das einführen von Bauelementen erleichtern.



Place Part: Bauteile können ausgewählt und danach platziert werden.



Erdung (Ground: „0/SOURCE“) kann ausgewählt werden und platziert werden. Sie ist für alle analogen Schaltungen zwingend erforderlich, damit PSpice ein Bezugspotential hat.



Bauteile können mittels Klicks auf ihre Anschlüsse verdrahtet werden. Rote ausgefüllte Kreise symbolisieren eine erfolgreiche Verbindung. Achtung, Bauteile können auch sehr leicht aus versehen überbrückt werden!

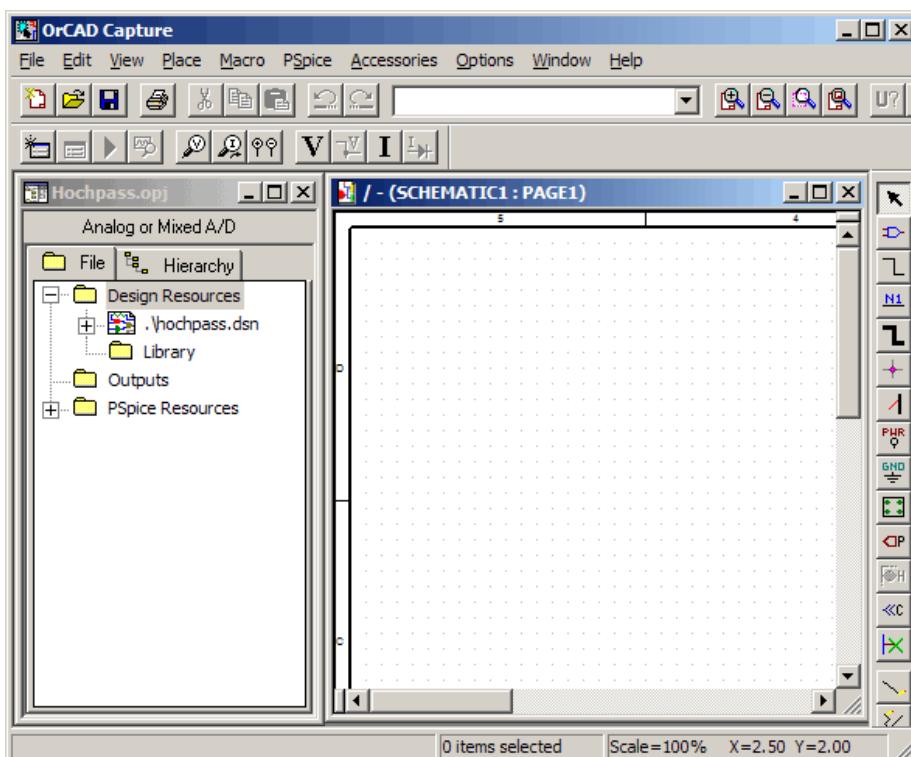


Abb. 03: Capture Arbeitsbereich

Über den „Place Part“ Button gelangen wir zur Auswahl der Bauelemente. Gegebenenfalls müssen hier über den „Add Library“ Button erst Bauteilbibliotheken hinzugefügt werden (*.OLB Dateien).

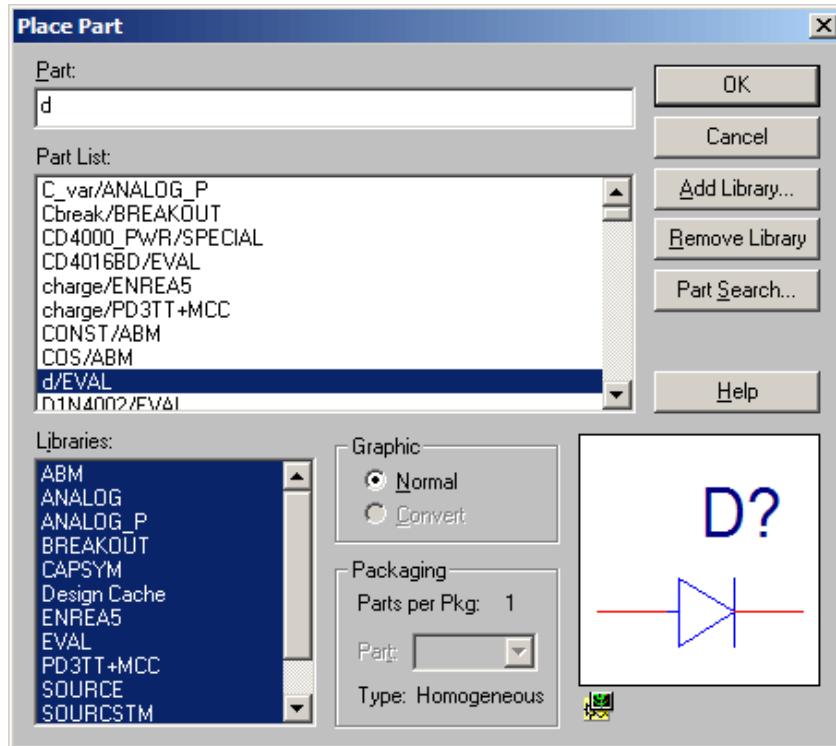


Abb. 04: Place Part-Dialog

Bauteile können nun durch einfaches Klicken platziert werden. Mittels „Escape“ (ESC) verlässt man den „place part“ Modus.

2.1.3. Erstellen des Simulationsprofils

Nachdem alle Bauteile platziert sind muss ein Simulationsprofil je nach Anforderungsbereich angelegt werden. Diese Profile sind vergleichbar mit dem Anschließen eines Oszilloskopes bzw. mit der Definierung davon, was eigentlich an der Schaltung gemessen werden soll. Ein Simulationsprofil erstellt man über Menu: PSpice->New Simulation Profile. Hier kann man für jedes Profil einen eigenen Namen angeben um es separat speichern und später wieder abrufen zu können.

PSpice unterscheidet vier Typen von Simulationen (zu sehen mittels des Pulldown Menus „Analysis type“).

1. Time Domain (Transient): Eine gewöhnliche Beobachtung der Spannungen über eine vorgegebene Zeitperiode, beispielsweise für Einschwingvorgänge.
2. DC Sweep (DC Durchlauf): Eine Gleichspannungsquelle kann als variabel deklariert werden und über einen vorzugebenden Spannungsbereich linear gesteuert werden.
3. AC Sweep (AC Durchlauf): Ein Wechselspannungsquelle kann als variabel deklariert werden und ihre Frequenz gesteuert werden.
4. Bias Point (Arbeitspunkt): Wird standardmäßig vor den anderen Analysen durchgeführt und berechnet die jeweiligen Spannungen im Arbeitspunkt.

Zusätzlich kann stets noch ein „parametric sweep“ (Parameter Durchlauf) durch das Anklicken des betreffenden Kästchens durchgeführt werden. Über entsprechende Deklaration kann so ein Widerstand kontinuierlich während des Simulationsprozesses verändert werden. In der Hilfe (F1) ist dies äußerst gut beschrieben.

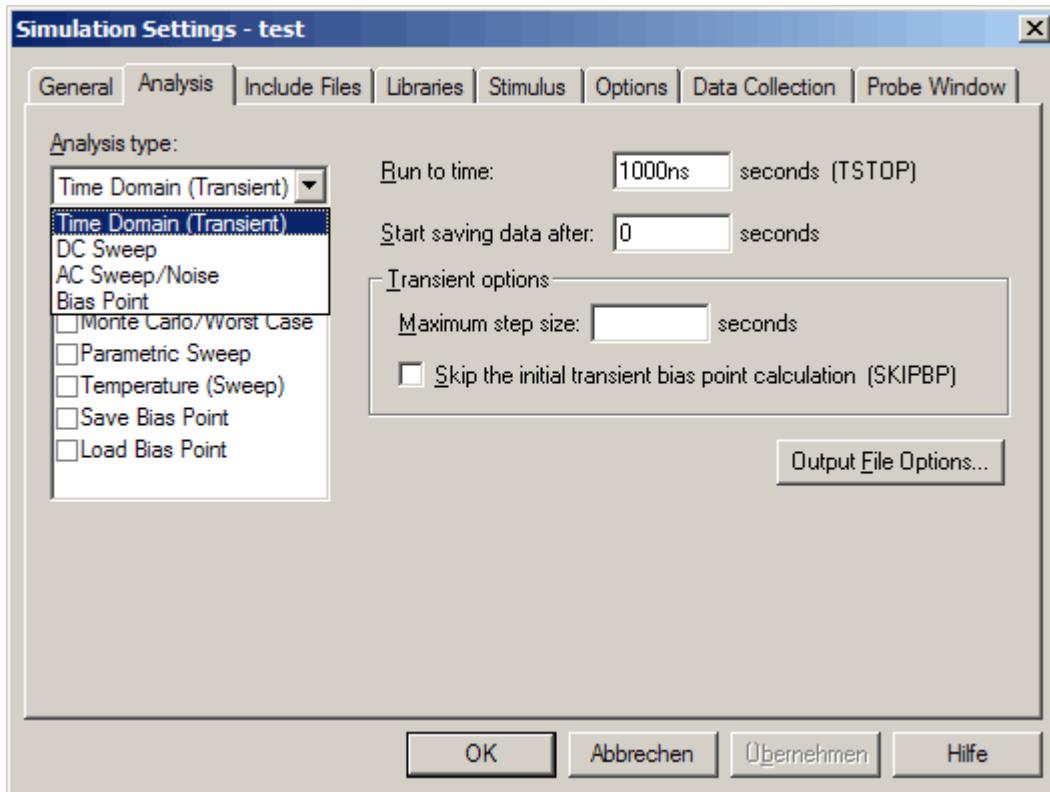


Abb. 05: Simulation Settings-Dialog

2.1.4. Definition von Messpunkten (Marker)

Nachdem ein Simulationsprofil angelegt wurde (sei es auch nur die standardmäßige Bias Point Simulation) können wir dazu übergehen Messpunkte zu definieren, an denen wir uns die Potentialunterschiede und Stromflüsse anzeigen lassen können.



Mit Hilfe dieser Knöpfe, oder über das Menu PSpice->Markers können Messpunkte in die Schaltung eingebracht werden. Stromabnehmer müssen dabei direkt an den Pin eines Bauteils angeschlossen werden.

Im Menu PSpice->Markers->Advanced findet man speziellere Messpunkte "DB Magnitude of Voltage/Current" und "Phase of Voltage/Current" die für Frequenz- und Phasengang-Simulationen bei Wechselspannungsanwendungen benötigt werden. Sie werden nur bei einem „AC Sweep“ verfügbar.

Nun ist es möglich die Simulation über das Menu: PSpice->Run zu starten.

2.1.5. Fahrplan zum Umsetzen eines Schaltplans

Zusammenfassend kann ein Schaltplan also folgendermaßen umgesetzt werden:

1. sicherstellen, dass benötigte Bauelemente in der Bibliothek vorhanden sind, sonst unter 2.3. nachlesen wie sich solche beschaffen lassen
2. alle Bauelemente und Spannungsquellen platzieren, Bauelemente verbinden, auf Schnittpunkte achten, diese müssen (um leitend zu sein) extra gesetzt werden
3. Masse platzieren, sie wird immer benötigt
4. Bauelemente benennen: Jedes Bauteil kann individuell benannt werden. Beim Setzen von Bauteilen werden diese durch ihre Abkürzung (R für Widerstand etc.) und durch eine laufende Nummer bezeichnet.
Diese Namen und weitere Optionen können durch Doppelklicken verändert werden.
Es dürfen jedoch niemals zwei Bauteile den gleichen Namen tragen.
5. Simulationsprofile festlegen
6. Messpunkte definieren
7. Simulation starten

2.2. PSpice A/D

Mit dem Ausführen des Run Befehls über Menu: PSpice->Run wird automatisch das nächste Teilprogramm des PSpice Bundles ausgeführt: PSpice A/D, das eigentliche Simulationsprogramm mit graphischer Ausgabe auf der Basis vom SPICE Algorithmus.

Zuerst prüft PSpice A/D dabei unseren mit Capture erstellten Schaltplan und die ihm zugrunde liegende „netlist“, also eine Art Source Datei, in die der Schaltplan umgewandelt wird und in der genau vermerkt ist, welche Bauelemente verwendet wurden und wie sie verknüpft sind. In der linken unteren Ecke des PSpice A/D Fensters sehen wir in einer kleinen Übersicht, ob diese Prüfung fehlerlos vonstatten gegangen ist. Ist dies nicht der Fall, so öffnet sich die Ausgabe Datei (Output File) und zeigt den Fehler an.

Dem rechten unteren Fenster sind Angaben zum Durchlauf der Simulation bzw. zu den Simulationsschritten zu entnehmen: beispielsweise während eines AC sweeps (AC Durchlauf) ist zu erkennen, wie die Schaltung für unterschiedliche Frequenzen simuliert wird. Das größte und über den anderen beiden liegende Fenster ist die eigentliche Ausgabe unserer Messung. Falls keine Messpunkte (Marker) definiert sind, sollte hier ein graues oder schwarzes Fenster ohne Graphen angezeigt werden – im Fehlerfall zeigt es unsere Ausgabe Datei. Falls alles nach Fahrplan (2.1.5) durchgeführt wurde, Messpunkte gesetzt wurden, können wir verschiedene Graphen ablesen, ähnlich dem Bildschirm eines High-End-Oszilloskops.

Wurden noch keine Marker (Messpunkte) definiert, so können auch über das Menu: Trace->Add Trace verschiedene Spannungs- oder Stromverläufe nachträglich zur Beobachtung hinzugefügt werden.

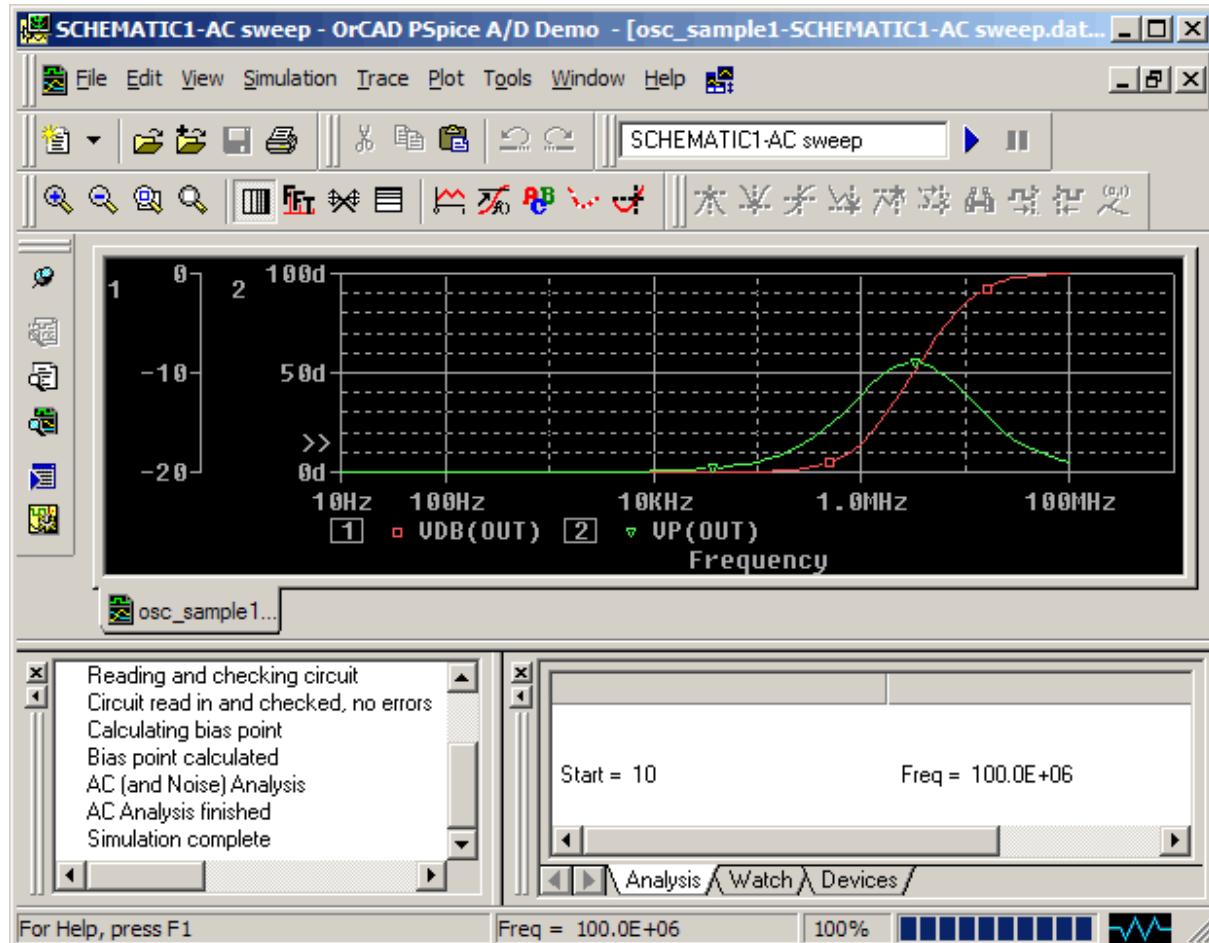


Abb. 06: PSpice AD Arbeitsbereich

2.3. Model Editor – oder wie beschaffe ich mir fehlende Bauteile?

Allen Bauelementen, die keine lineare Kennlinie haben, also zum Beispiel Transistoren oder Dioden, liegt in PSpice ein so genanntes Modell zu Grunde, welches die nötigen Informationen enthält, wie sich das Bauelement im Spannungsfall zu verhalten hat.

Möchten wir die Eigenschaften eines vorhandenen Elements ändern, da wir mit einem Typ arbeiten wollen, der nicht in unserer Datenbank vorhanden ist, über den wir aber notwendige Informationen wie Durchbruchspannung oder Ähnliches beispielsweise durch ein Datenblatt vorliegen haben, können wir den Model Editor benutzen um den Bauteiltyp anzupassen.

Hierzu markieren wir das Element (beispielsweise eine Diode) in Capture (siehe 2.1.) und rufen über das Menu: Edit->PSpice Model den Editor auf.

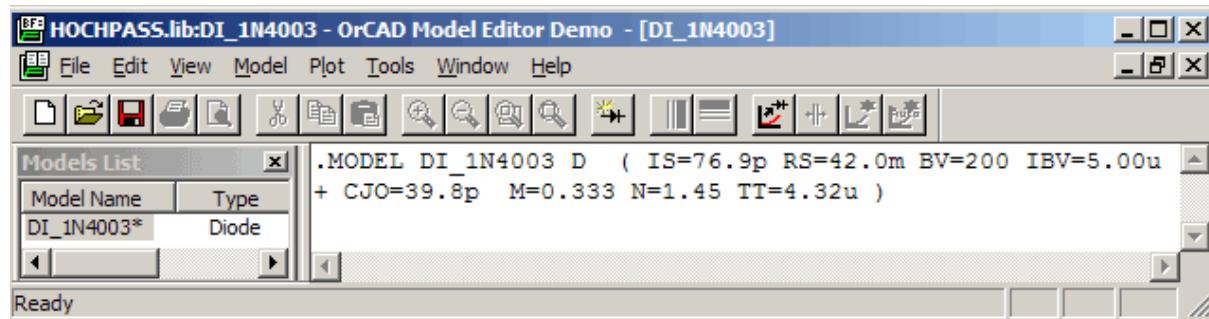


Abb. 07: Model Editor Arbeitsbereich

Im Model Editor lassen sich über das Menu->File->New auch ganz neue Typen erstellen. Die genauen Einstellungen sind äußerst gut in der Hilfe beschrieben und lassen sich über das Suchen ihrer Abkürzungen (zum Beispiel TT = Transit Time, wie im oberen Bild zu sehen) sehr leicht finden.

Ein durchaus einfacherer Weg, der in den meisten Fällen schneller zum Ziel führt, ist die Suche bei Google.de nach dem Bauteilnamen in Verbindung mit „model“ oder „pspice“. Heruntergeladen werden können dann meist *.LIB (Model Library) Dateien, die über den Model Editor geöffnet und über das Menu: File->Create Capture Parts in eine OLB-Datei (OrCAD Library) umgewandelt werden können. Diese sind dann wiederum über den „place Part“ Dialog (2.1.2.) über „Add Library“ in Capture zu benutzen. Siehe 3.3. als Beispiel hierfür.

2.4. Einheiten und gebräuchliche Shortcuts

PSpice unterscheidet nicht zwischen Groß- und Kleinschreibung. Demnach ergibt sich eine etwas ungewöhnliche Benutzung der Zehnerpotenzen:

meg = MEG = mega =	1E6
k = K = kilo =	1E3
m = M = milli =	1E-3
u = U = mikro =	1E-6
n = N = nano =	1E-9
p = P = piko =	1E-12

Alle Shortcuts zur schnelleren Programmverwendung können sehr schön aufgelistet in der Hilfe (nach „shortcuts“ suchen) gefunden werden. Eine kleine Auswahl möchte ich dennoch hier präsentieren:

R	-	rotiert das markierte Element gegen den Uhrzeigersinn
SHIFT+P	-	„place part“ Dialog wird geöffnet
SHIFT+W	-	„place wire“ (Bauteilverbindung) wird aktiviert
ALT+S, N	-	Erstellt neues Simulationsprofil
ALT+S, E	-	editiert aktives Simulationsprofil
ALT+S, R	-	Starten der Simulation

3. Beispiele

Die Umsetzung der Beispiele orientiert sich immer am Fahrplan, der unter 2.1.5. vorgestellt wurde. Das Starten von Capture, Anlegen eines neuen Projektes und Auswählen der Bauteile wird nicht erneut aufgegriffen, da es vergleichbar mit dem unter 2.1. präsentierten Verfahren ist.

Alle Beispiel Dateien stehen auf der Projektlaborseite (<http://projektlabor.ee.tu-berlin.de>) als gepackte Files zum Download zur Verfügung.

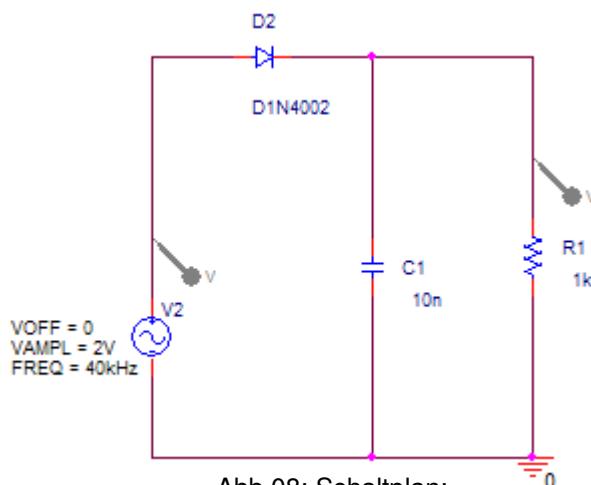
3.1. Diodengleichrichter (Time Domain(transient) | Zeitanalyse)

Wir bauen die Schaltung gemäß folgendem Schaltplan auf:

Die Diode D2 ist in den Standardbibliotheken von PSpice enthalten.

Bei der Spannungsquelle handelt es sich um das Modell VSIN, welches durch ein Doppelklicken auf die Attribute eine Offsetverschiebung von 0V, eine Amplitude von 2V und eine Frequenz von 40kHz zugewiesen bekommen hat.

Die Werte des Kondensators und des Widerstandes müssen gleichermaßen angepasst werden. Wichtig ist, die Masse nicht zu vergessen.



Anschließend erstellen wir ein neues Simulationsprofil (2.1.3.). Wir verwenden hierbei die Zeit Analyse, also den voreingestellten Punkt „Time Domain (transient)“. Dies ist notwendig, da wir die Amplitudenverläufe über einige Perioden beobachten möchten.

Anzupassen ist demnach in den Simulations Settings nur noch die Durchlaufzeit (Run to time). Wir setzen den Wert auf 100us. Gleichzeitig setzen wir den Wert der „maximum Step size“ auf 100ns um eine saubere Sinuskurve zu erhalten. Wir schließen den Dialog.

Nun platzieren wir zwei Messpunkte (2.1.4) nach obiger Skizze und starten die Simulation.

Im sich öffnenden PSpice AD Fenster können wir folgende Spannungsverläufe erkennen:

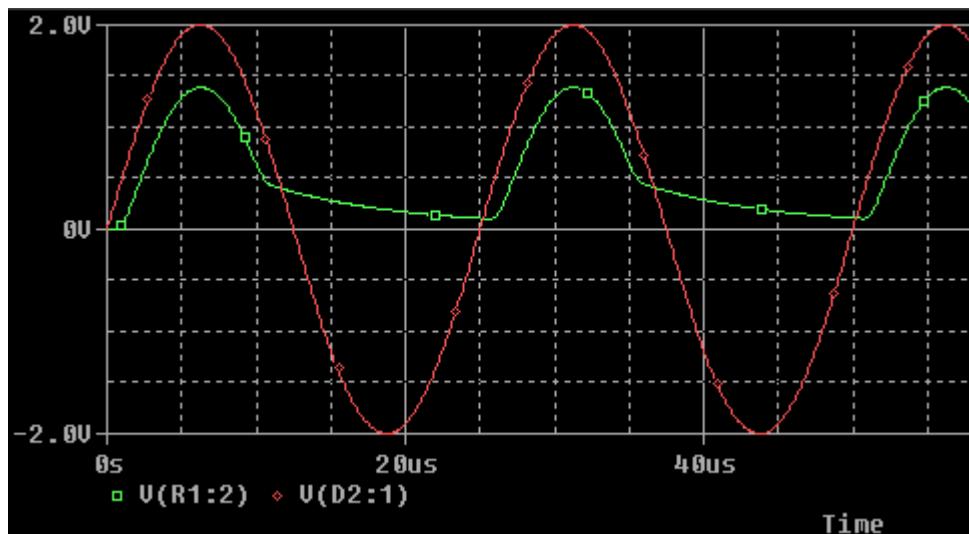


Abb. 09: Spannungsverläufe Gleichrichter

3.2. Transistor Ausgangskennlinienfeld (zweifacher DC sweep)

Wir bauen die Schaltung gemäß folgendem Schaltplan auf:

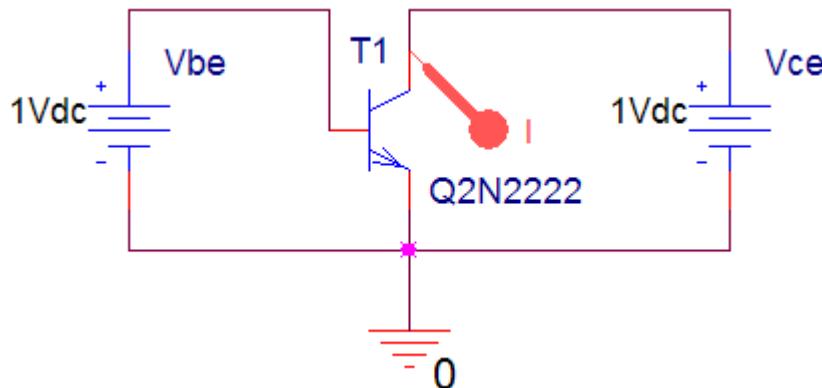


Abb. 10: Schaltplan Transistorkennlinien

Der Transistor T1 (Q2N2222) ist in den Standardbibliotheken von PSpice enthalten.

Bei den Spannungsquellen handelt es sich stets um das Modell VDC, welche durch ein Doppelklicken auf die Attribute eine Gleichspannung von 1V und jeweils die Bezeichnungen Vbe und Vce zugewiesen bekommen haben.

Wichtig ist, die Masse nicht zu vergessen.

Anschließend erstellen wir ein neues Simulationsprofil (2.1.3.). Wir aktivieren hierbei den Analyse Typen DC Sweep aus dem Pulldown Menu unter der Option Primary Sweep (erster Durchgang) und tragen die Spannungsquelle (Voltage source) als Vbe ein, sowie folgende Durchlaufoptionen:

Start value: 0.5V End value: 12V Increment: 0.2V

Bei der Option Secondary Sweep machen wir einen Haken und spezifizieren unsere Spannungsquelle als Vce mit gleichen Werten wie unter Vbe, bis auf:

End value: 2V

Wir schließen den Dialog.

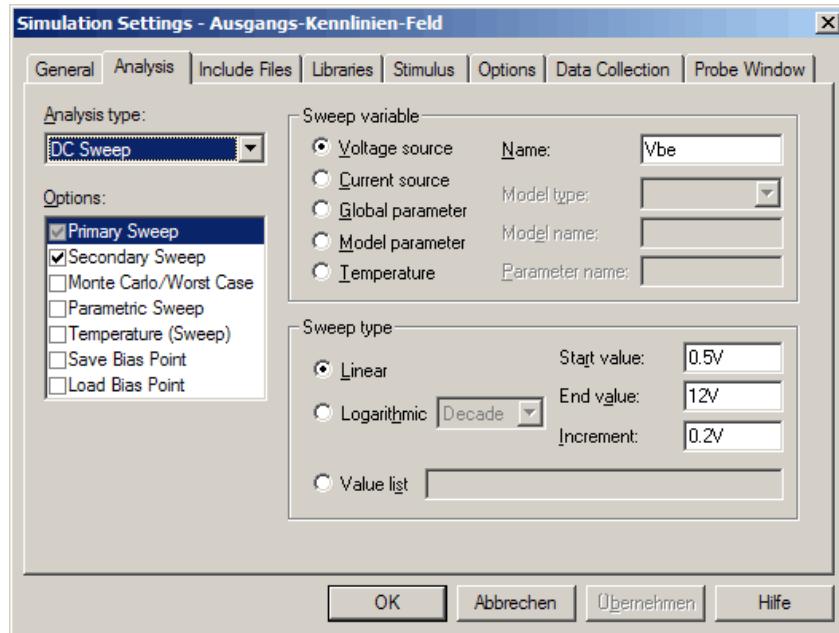


Abb. 11: Simulation Settings-Dialog

Nun platzieren wir einen Strommesspunkt (2.1.4) nach obiger Skizze auf den Kollektorpin des Transistors und starten die Simulation.

Im sich öffnenden PSpice AD Fenster können wir folgendes Ausgangskennlinienfeld erkennen:

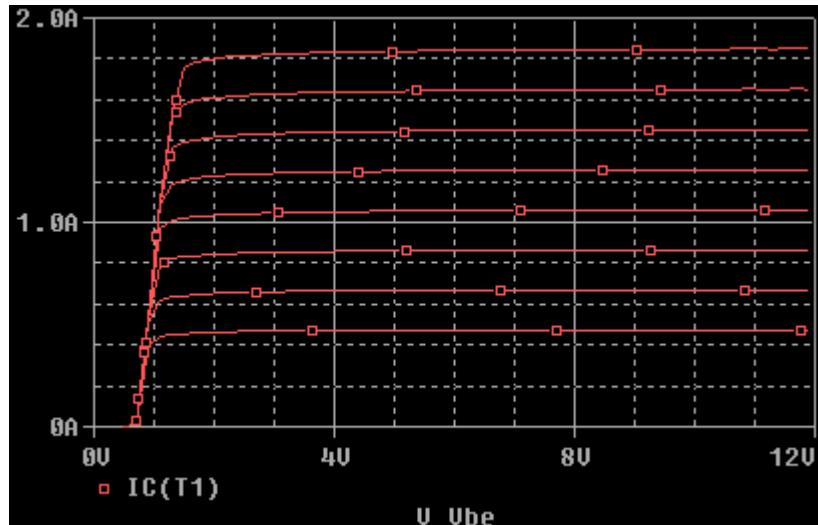


Abb. 12: Ausgangskennlinienfeld

3.3. Bodediagramm anhand des Verstärkers aus dem 2. Semester (AC Sweep)

3.3.1. Bauteil-Modell-Beschaffung

Wir benötigen folgende Bauteile:

- Transistor BC549C
- Transistor BC141 / Transistor BC161
- Diode 1N4003

- a) Suche bei Google nach: .model BC549C
 Erster Treffer: <http://orion.ipt.pt/~jorge/Electronica1/PSPICE/libs/ebipolar.lib>
 Datei ebipolar.lib im PSpice Verzeichnis unter /Capture/Library/PSpice speichern und im Model Editor öffnen. Dort im Menu unter File->Create Capture Parts wieder die Datei ebipolar.lib auswählen und die OLB-Datei generieren lassen.
- b) Die Transistoren T3, T4 sind auch nach langwierigem Suchen nicht zu finden. Wir benutzen Standardtypen.
- c) Suche bei Google nach: .model 1N4003
 Erster Treffer: <http://homepages.which.net/~paul.hills/Circuits/Spice/ModelIndex.html>
 Von dort aus werden wir weiter verwiesen auf die Seite:
<http://www.diodes.com/products/spicemodels/index.php> wo wir nach 1N4003 suchen und folgendes geliefert bekommen
- ```
*SRC=1N4003;DI_1N4003;Diodes;Si; 200V 1.00A 3.00us Diodes, Inc. Diode
.MODEL DI_1N4003 D (IS=76.9p RS=42.0m BV=200 IBV=5.00u
+ CJO=39.8p M=0.333 N=1.45 TT=4.32u)
```
- Wir legen eine neue Textdatei an mit dem Namen DI\_1N4003.lib, kopieren dort die gefundenen Diodendaten hinein und öffnen die Datei mit dem PSpice Model Editor. Im Menu: File->Create Capture Parts wählen wir unsere erstellte DI\_1N4003.lib an und bestätigen den Dialog. Die Schaltplandatei DI\_1N4003.OLB wird erzeugt.

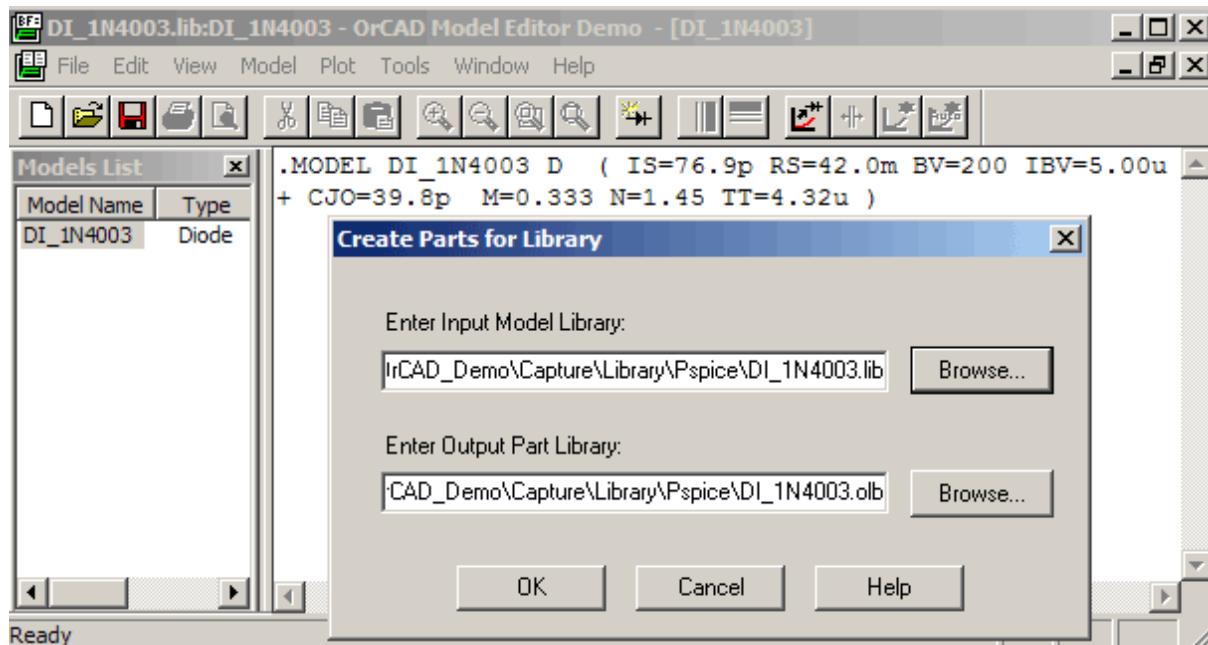


Abb. 13: Create Parts Dialog im Model Editor

Falls alles problemlos verläuft erhalten wir nun die Nachricht, dass „0 Error messages“ produziert wurden.

### 3.3.2. Hinzufügen der Libraries (OLB)

Um die Bauteile auf dem Schaltplan in Capture platzieren zu können, müssen wir die Schaltzeichen erst hinzufügen. Dies tun wir über den Place Part-Dialog (2.1.2.). Wir wählen „Add Library“ und suchen unsere frisch erzeugten OLB-Dateien.

### 3.3.3. Aufbauen des Schaltplans

Wir bauen den Schaltplan folgendermaßen auf:

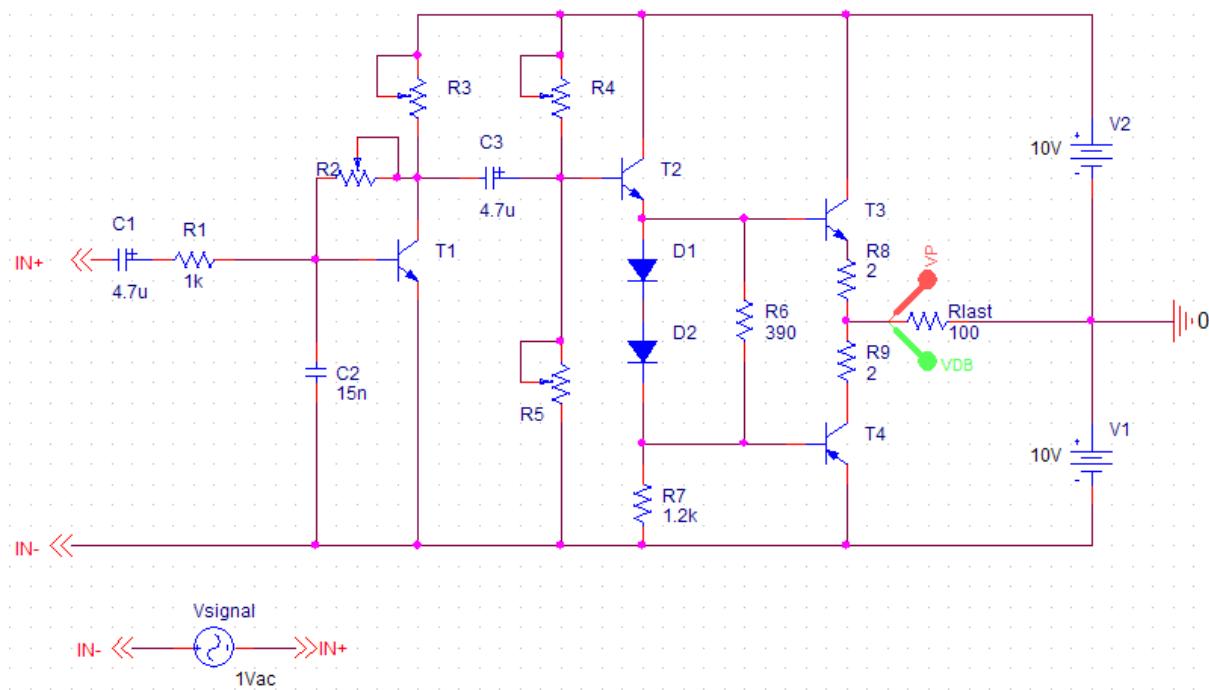


Abb. 14: Schaltplan des NF-Verstärkers

Wir benutzen dabei die Bauteile, wie im Skript Get1a beschrieben:

|          |            |                                                                                            |
|----------|------------|--------------------------------------------------------------------------------------------|
| R1=1k    | R6=390     | C2=15nF                                                                                    |
| R2=480k  | R7=1.2k    | T1,2=BC549C                                                                                |
| R3=1k    | R8,9=2     | T3,4 sind ersatzweise Standard Transistoren (einmal NPN und PNP z.B. Q2N2222 und Q2N2907A) |
| R4=48.9k | Rlast=100  |                                                                                            |
| R5=71.1k | C1,3=4.7uF | D1,2=1N4003                                                                                |

Für die veränderbaren Widerstände werden uns in PSpice die Bauteile POT angeboten, deren Wert wir durch Doppelklicken und daraufhin entsprechendes Suchen nach „Value“ ändern können. Die Signalquelle ist vom Typ VSIN und die Gleichspannungsquellen VAC. Die „Offpage Connectors“ IN+ und IN- sind über folgendes Symbol zu erreichen: , wichtig ist bei diesen „Verbindungsstellen“, dass zu verbindende Stränge gleiche Namen erhalten!

### 3.3.4. Erstellen des Simulationsprofils und einbinden der Libraries (LIB)

Anschließend erstellen wir ein neues Simulationsprofil (2.1.3.). Wir aktivieren hierbei den Analyse Typen AC Sweep aus dem Pulldown Menu und tragen folgende Durchlaufoptionen ein:

Start Frequency: 1  
 End Frequency: 50Meg  
 Points/Decade: 11

Außerdem ist es wichtig, die LIB-Dateien (ebipolar.lib und DI\_1N4003.lib) im Simulationsablauf einzubinden, so dass PSpice die Modelle zu den verwendeten Schaltplansymbolen findet.

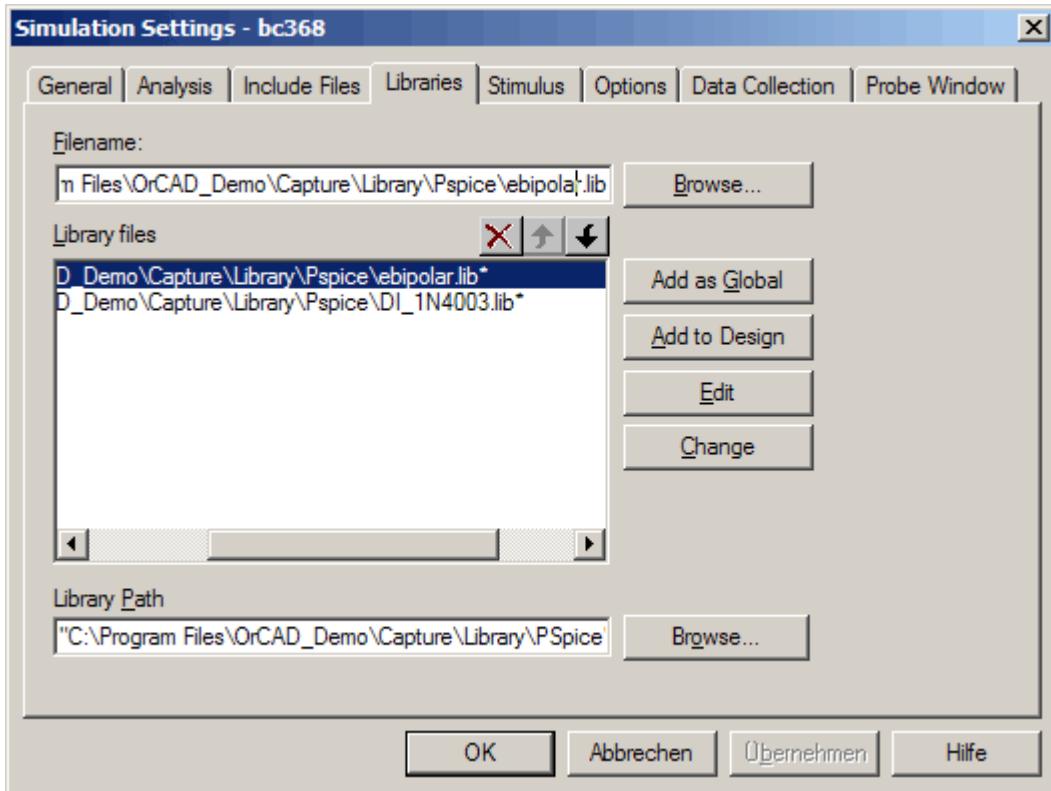


Abb. 15: Hinzufügen der Lib-Dateien im Simulation Settings Dialog

Dazu wählen wir den Libraries Tab-Reiter, suchen unsere Lib-Dateien und klicken auf „Add as Global.“

Wir verlassen den Dialog mittels OK.

### 3.3.5. Setzen der Messpunkte

Da wir die AC Sweep Analyse ausgewählt haben, ermöglicht uns PSpice so genannte „advanced markers“ zu verwenden. Sie sind über das Menu: PSpice->Markers->Advanced zu erreichen und messen, ohne weitere Einstellungen, den Phasen- (Phase of Voltage) und Frequenzgang (DB Magnitude of Voltage) unserer Schaltung. Wir platzieren die beiden vor unserem Widerstand Rlast.

### 3.3.6. Simulationsergebnisse

Es ergeben sich folgende Kurvenverläufe:

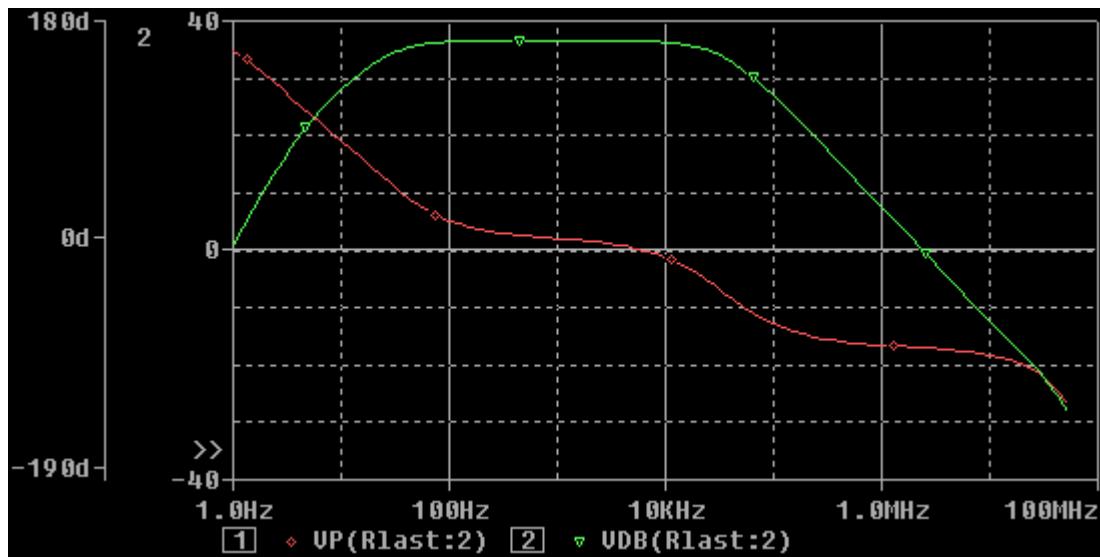


Abb. 16: Simulationsergebnisse: Frequenz- und Phasengang

Ein Vergleich mit der damaligen Musterlösung belegt uns auch gleich die Richtigkeit unserer Simulation. Eventuelle Abweichungen resultieren aus den unterschiedlichen Transistortypen T3 und T4.

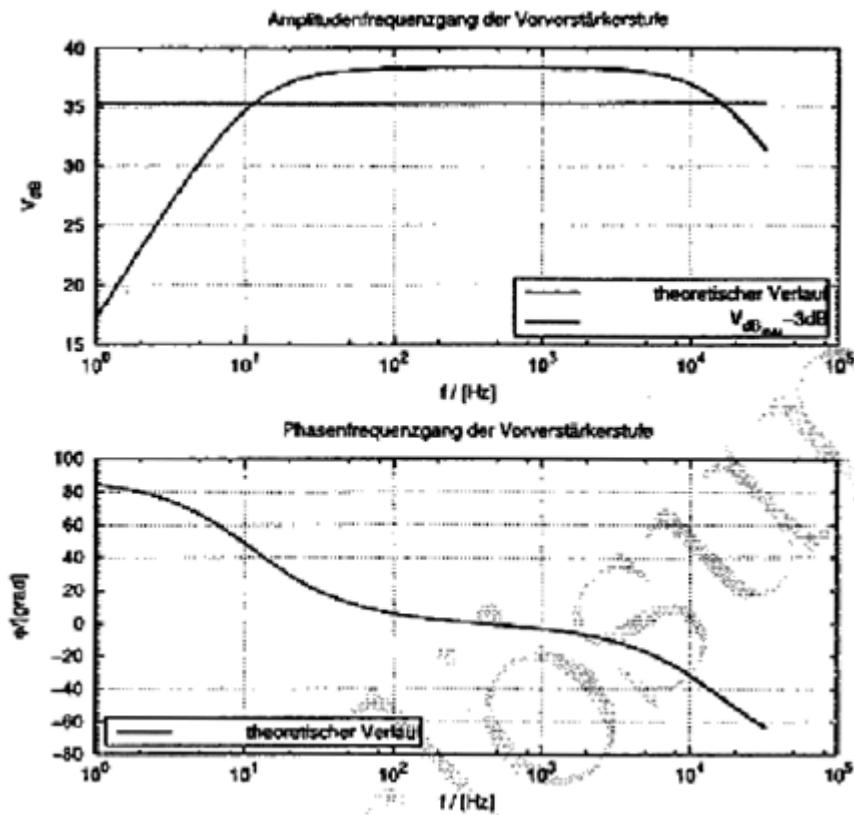


Abb. 16: Frequenz- und Phasengang aus der Musterlösung des GDE1b Skriptes

## 4. Typische Fehlerquellen

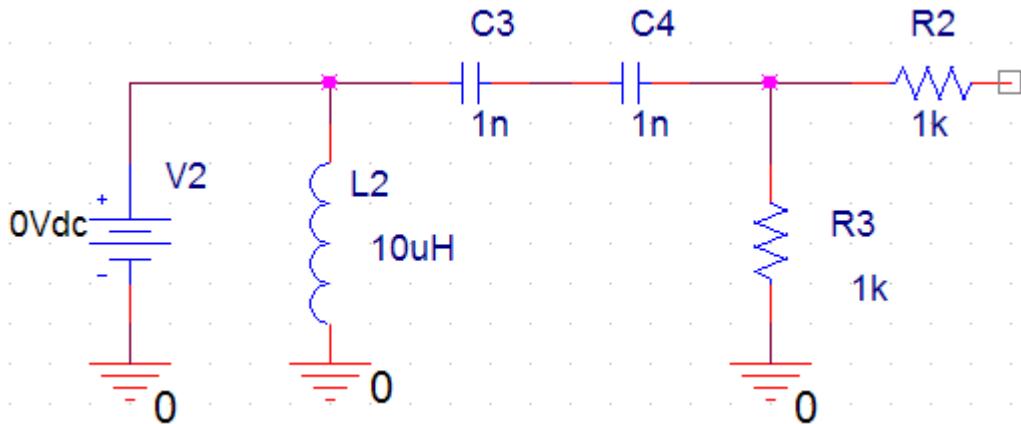


Abb. 15: Schaltplan mit Fehlern

### 4.1. Anhand eines Beispiels

Die typischen Fehler werden anhand des Schaltplans in Abb. 15 erläutert, der mehrere Fehler enthält. Wenn ein Fehler auftritt, erscheinen vor oder nach der Simulation die Fehlermeldungen.

- *Floating Pin:*  
Ein Anschluss eines Bauteils ist nicht angeschlossen, z.B. bei  $R_2$  in Abb. 15. Dieser Fehler tritt bereits bei der Erzeugung der Netzliste auf; es wird ein Dialog mit dem Hinweis *ERC: Netlist/ERC errors - netlist not created*, dann *Floating pin: R2 pin 2* angezeigt. Im Allgemeinen muss jeder Anschluss beschaltet sein. Eine Ausnahme sind speziell konfigurierte Bauteile oder Makromodelle, die an einem oder mehreren Anschlüssen bereits eine *interne* Beschaltung aufweisen, so dass keine *externe* Beschaltung erforderlich ist.
- *Node <Knotenname> is floating:*  
Die Spannung eines Knotens kann nicht ermittelt werden, weil sie unbestimmt ist; das ist in Abb. 15 beim Knoten  $K_2$  der Fall. Diese Fehlermeldung tritt immer dann auf, wenn an einem Knoten nur Kapazitäten und/oder Stromquellen angeschlossen sind; durch letzteres ist die Kirchhoff'sche Knotenregel nicht erfüllt. Jeder Knoten muss über einen Gleichstrompfad nach Masse verfügen, damit die Knotenspannung eindeutig ist. Im Fall des Knotens  $K_2$  in Abb. 15 kann man zum Beispiel einen hochohmigen Widerstand von  $K_2$  nach Masse ergänzen, um den Fehler zu beheben.
- *Voltage and/or inductor loop involving <Bauteil>:*  
Es existiert eine Masche aus Spannungsquellen und/oder Induktivitäten, die gegen die Kirchhoff'sche Maschenregel verstossen, z.B. wird in Abb. 15 die Spannungsquelle  $U_1$  durch die Induktivität  $L_1$  gleichspannungsmäßig kurzgeschlossen.

#### 4.2. Allgemeine Tipps:

- F1 drücken und in der Hilfe nach dem angegebenen Fehler suchen
- Leiterkreuzungen überprüfen, sollen die zwei Leiter eine Verbindung haben, so muss vielleicht ein Knotenpunkt gesetzt werden.
- Ist an allen dafür vorgesehenen Punkten die Masse angeschlossen?
- Sind die Sweeps (Durchläufe) in den Simulations-Settings sinnvoll/richtig eingestellt?

## 5. Weiterführende Literatur und Links

### 5.1. Bezugsquellen der Studenten Version von PSpice

<http://www.orcad.com/downloads/>

offizielle Download Seite, wohl aber nicht in der alten Version 9.1 erhältlich

<http://www.electronics-lab.com/downloads/schematic/013/>

Neben der Student-Version noch Tutorials und anderes

### 5.2. allgemeine Links

[http://projektlabor.ee.tu-berlin.de/projekte/roboter/downloads/referate/PSpice/PSpice-Ref\\_schriftl.pdf](http://projektlabor.ee.tu-berlin.de/projekte/roboter/downloads/referate/PSpice/PSpice-Ref_schriftl.pdf)

schriftliche Referatsausarbeitung des PSpice-Referenten im SS2004

<http://www.orcadpcb.com/pspice/movies.asp?bc=F>

Lernvideos, Tutorials, etc.

<http://stuweb.ee.mtu.edu/~syeoh/PSpice%20User%20Guide.pdf>

<http://www.ee.ucla.edu/~jjudy/classes/ee100/pspice/manuals/PSpice%20User%20Guide.pdf>

OrCAD Pspice User's Guide

[http://www-ibt.etecl.uni-karlsruhe.de/linette/pspice/pspice\\_index.html](http://www-ibt.etecl.uni-karlsruhe.de/linette/pspice/pspice_index.html)

grundlegendes Tutorial

### 5.3. Bauteil Modell Beschaffung

<http://www.spicelab.de/links.htm>

<http://www.orcadpcb.com/pspice/models.asp?bc=F>

<http://homepages.which.net/~paul.hills/Circuits/Spice/ModellIndex.html>

<http://www.5spice.com/links.htm>

<http://www.spectrum-soft.com/links.shtml>

## 6. Quellenangaben

o.V.: OrCAD – PSpice A/D User's Guide

[http://www.electronics-lab.com/downloads/schematic/013/psp\\_pdf.zip](http://www.electronics-lab.com/downloads/schematic/013/psp_pdf.zip)

(Abruf: 7. Nov. 2004)

o.V.: PSpice Online Hilfe (im Programm integriert)

Paul W. Tuinenga, Spice A Guide to Circuit Simulation and Analysis Using PSpice

Prentice Hall (1993) ISBN: 0-13-834607-0