



Gunthard Kraus, DG 8 GB

Das interessante Programm

Heute: **Schaltungssimulation mit PSPICE**

1.

Worum geht es?

Wer sich mit Elektronik oder Schaltungsentwicklung beschäftigt, hat sicher schon etwas von PSPICE (= SPICE for the PC) gehört; viele kennen es und einige arbeiten täglich damit. Deshalb geht es hier um folgende Fragen:

- a. Was steckt hinter PSPICE?
- b. Was leistet es?
- c. Welcher finanzielle Aufwand ist dafür erforderlich und wie kommt ein Neuling oder Einsteiger möglichst schnell über die Anfangshürden?

2.

Was steckt dahinter?

PSPICE ist - schlicht und einfach formuliert - ein Programm zur Simulation des Verhaltens elektronischer Schaltungen oder Komponenten im Zeitbereich. Im Klartext bedeutet

dies, dass man eine Spannung mit frei wählbarer Kurvenform an den Eingang anlegen und nach dem zeitlichen Verlauf von Spannungen und Strömen an allen Stellen des Stromkreises fragen kann. Der Begriff „Stromkreis“ kann jedoch täglich weiter gefasst werden, weil laufend neue Modelle für alle möglichen Dinge, wie z. B. DC-Motoren oder irgendwelche Sensoren im Internet auftauchen. Entwickelt wurde der Grundstock dieser Software vor mehr als 30 Jahren von der Berkeley Universität in den USA, die auch heute noch über die Standards und Lizenzen wacht.

Das nötige Stichwort ist bereits gefallen: der Kern der Sache ist immer ein Modell des zu untersuchenden Bauteils. Es muss entweder im verwendeten Programm vorhanden sein oder vom Bauteilhersteller veröffentlicht werden. In einer solchen Datei stehen nach genau festgelegten Spielregeln (= PSPICE-Syntax) alle erforderlichen Informationen für die Simulation. Diese Modelldateien tragen bei einzelnen Komponenten die Endung „*.cir“, bei umfangreichen Modellsammlun-

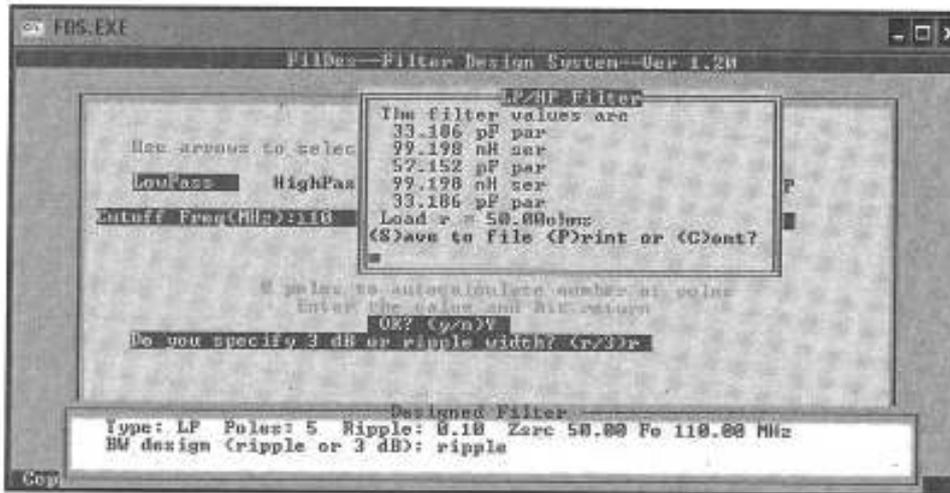


Bild 1:
Altbekannt,
aber immer
noch wir-
kungsvoll:
Bestimmung
von Filter-
Bauteilwerten
mit „fds“

gen (wie z.B. für alle NPN-Transistoren einer Firma) wird dagegen „*.lib“ verwendet.

Bei einer schnellen und fast idealen Schottky-Diode sieht das beispielsweise so aus:

```
.MODEL
.SCHOTTKY
D(IS=1.0E-10 EG=0.69 N=1.0 XTI=2.0
CJO=0.5PF)
```

Man ahnt schon (..bitte genau hinschauen: es geht immer mit einem Punkt als Einleitung los...):

„**model**“ liefert dem Programm grundsätzlich die Startinformation für eine neue Modellbeschreibung in einer Bauteil-Bibliothek (Library).

„**Schottky**“ ist die Bezeichnung dieses Modells.

Und mit „D(.....)“ folgt nun die echte physikalische Beschreibung für die Simulation, dabei steht hier der Buchstabe „D“ als Kennzeichnung einer Diode. In der Klammer selbst befinden sich diejenigen Details, die nur der Spezialist versteht: „IS“ zeichnet den Sättigungsstrom und „CJO=0.5 pF“ gibt die Sperrschichtkapazität wieder, was nicht schwer zu erraten ist. Beim Rest muss man allerdings schon „schlaue“ Bücher wälzen.

Leider reicht diese einfache Beschreibungsstruktur sehr schnell nicht mehr aus - man denke nur an einen Operationsverstärker, der viele Transistoren und / oder FETS und

```
*1Nxxx diode models and subcircuits
* rename of Thomatronik model D1N4007

.SUBCKT 1N4007 1 2
D1 1 2 D1N4007

.MODEL D1N4007 D IS = 1.09774E-008 N = 1.78309 RS = 0.0414388
+ EG = 1.11 XTI = 3
+ CJO = 2.8173E-011 VJ = 0.50772 M = 0.318974 FC = 0.5
+ TT = 9.85376E-006 BV = 1100 IBV = 0.1 AF = 1 KF = 0
ENDS
```

Tabelle 1:
Circuit-
Beschreibung
für eine Universal-
diode 1N4007



Dioden enthält. Hier taucht als neuer Sammelbegriff der „Subcircuit“ auf, der in der Modellbeschreibung leicht als „**SUBCKT**“ zu identifizieren ist und in dem selbst wieder viele Modelle eingesetzt werden.

Dort findet man auch die Anschlussbein-Nummern dieses Bauteils aufgeführt, die man immer als Information für das Erstellen des Schaltplans benötigt.

Ganz wichtig ist, dass die oft sehr umfangreiche Subcircuit-Beschreibung immer mit „**ENDS**“ korrekt abgeschlossen werden muss.

Für die wohlbekanntere Universal-Netzdiode 1N4007 sieht das folgendermaßen aus, wie in **Tabelle 1** dargestellt (das Sternchen * leitet einen Kommentar ein, der für die Simulation ignoriert wird!).

Bitte genau hinsehen:

- a. Die genaue Attributbeschreibung wird mit „MODEL“ eingeleitet und trägt den Namen „D1N4007“.
- b. Diese Beschreibung „D1N4007“ wird von einem Diodenmodell D1 verwendet, das die beiden Anschlussbeine „1“ und „2“ besitzt und durch die Zeile „D1 1 2 D1N4007“ erzeugt wird.
- c. Und erst mit „SUBCKT 1N4007 1 2“ bekommt man ein komplettes Bauteil zur Verwendung im Simulationsprogramm (wobei es PSPICE nun völlig egal ist, wie dieser Subcircuit selbst aussieht - er wird einfach als Bauklötzchen betrachtet!). Die letzte Zeile „ENDS“ stellt den erfolgreichen Abschluss des Subcircuits dar.

Nachdem nun die Bauteile zur Verfügung stehen, geht es an die Erstellung der Schaltung. Das war früher ein beträchtlicher Aufwand, da man dazu einen Texteditor brauchte und Zeile für Zeile angeben musste, zwischen welchen Knoten (= Verbindungsstellen von zwei oder mehreren Bauteilen) ein bestimmtes Teil angeordnet ist und welche Modellbeschreibung dafür gilt. Diese Auflistung trägt den Namen „**Netzliste**“ (= netlist) und auch heute noch benützt jede PSPI-

CE-Version diese Netzliste als Vorgabe für die Simulation. Nur kann sich heutzutage kein Programmierer mehr trauen, seinen Kunden die Eintipperei zuzumuten. Jetzt arbeitet man ausschließlich mit komfortablen „Schaltungseditoren“ und erstellt damit seine Schaltung auf dem Bildschirm. Daraus erzeugt letztlich die Software die erforderliche Netzliste!

Aber was geschieht, wenn alles fertig gezeichnet ist und die Simulation gestartet wird? Da gilt:

Zunächst führt das Programm eine „Gleichstrom-Analyse“ durch und reagiert prompt, wenn es bei irgendeinem Knoten in der Schaltung keine Gleichstrom-Verbindung nach Masse findet. Der Benutzer wird sofort durch eine Bildschirmmeldung dazu aufgefordert, diesen groben Fehler zu beseitigen. In der Praxis sieht das z.B. so aus, dass man bei einer Reihenschaltung von zwei Kondensatoren einen Widerstand von 10 bis 100 MOhm vom Verbindungspunkt nach Masse einfügen muss. Genauso werden fehlende Verbindungen zwischen zwei Bauteilen angemahnt, wenn die Beinchen nur lose nebeneinander liegen. Auch Induktivitäten sind recht tückisch, da nach der Theorie in einer idealen Induktivität der Strom beim Anlegen einer Gleichspannung irgendwann unendlich groß werden kann. Also setzt man nie eine Spule oder einen Transformator ein, ohne gleich einen (wenn auch noch so winzigen) Serienwiderstand vorzusehen.

Ist diese Hürde genommen und das Programm mit der erstellten Schaltung zufrieden, beginnt die Simulation. Der hierbei für die Simulation angewandte Trick ist so einfach, dass man fast selbst darauf kommen könnte. Man geht nur einen so kleinen Zeitschritt weiter, dass man alle entstehenden Änderungen als **LINEAR** betrachten kann und deshalb mit recht einfachen mathematischen Formeln auskommt. Ist für diesen Zeitausschnitt alles berechnet, geht man wieder einen kleinen Zeitschritt weiter, rechnet erneut die linearen Änderungen aus und

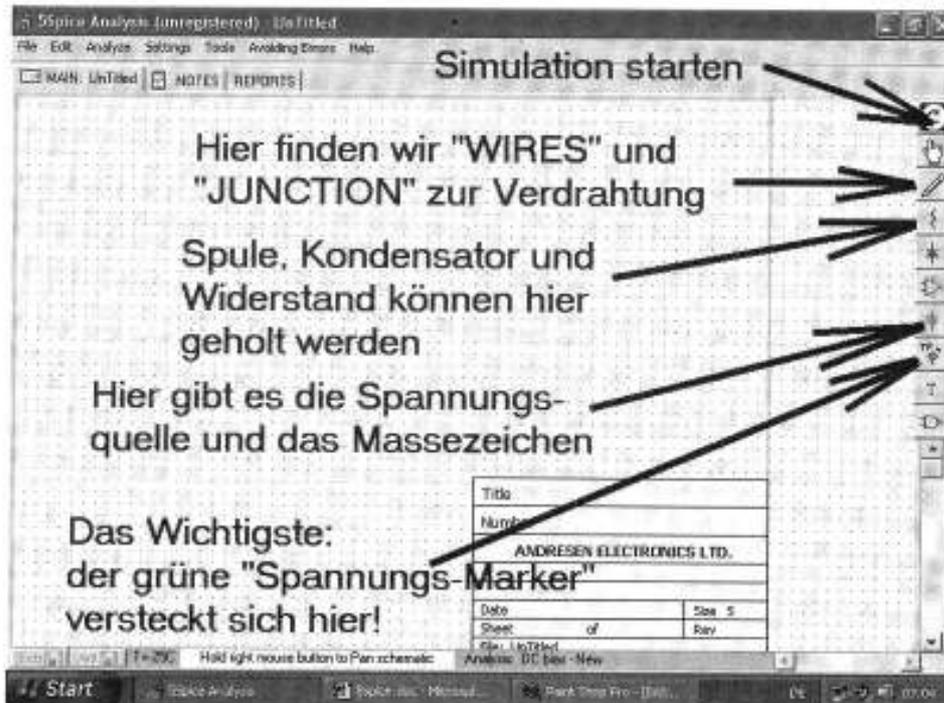


Bild 2:
Leicht zu bedienen:
Das Bauteilmenü von
"5Spice"

setzt das Ganze nach diesem Prinzip bis zum Ende der vorgesehenen Simulationszeit fort.

Eines muss man natürlich immer im Auge behalten:

Wählt man die Zeitschritte sehr klein, sehen die Spannungs- oder Stromverläufe perfekt aus; aber das kostet Rechenzeit und Speicherplatz für die Ablage aller Zwischen- und Endergebnisse. Umgekehrt erkennt man zu große Zeitschritte sofort daran, dass z.B. sinusförmige Signale plötzlich aus lauter kurzen, geraden Stücken zusammengebastelt sind und entsprechend unattraktiv aussehen. Hier muss der Anwender selbst entscheiden, was ihm wichtiger ist.

3. Was leistet PSPICE?

Das ist leicht zu beantworten:
Die PSPICE-Simulation wird so genau, wie

der Anwender alle beteiligten Effekte in Form von Bauteilen in die Schaltung einbringt und wie genau jedes Modell das tatsächliche Verhalten jedes Bauteils beschreibt!

Einer der größten Vorzüge ist aber, dass PSPICE sich immer mit den zeitlichen Verläufen der Signale beschäftigt und deshalb den Anwender über jede Nichtlinearität oder Verzerrung innerhalb einer Schaltung automatisch informiert. Die Bestimmung der enthaltenen Oberwellen oder Spektralanteile erfolgt dabei durch den Wechsel in die „Frequency Domain“ über das mathematische Verfahren der „Fourier-Transformation“. Auch die Ermittlung von Frequenzgängen (...sie trägt hier den Fachnamen „AC-Sweep“) ist kein Problem und wird selbst von der einfachsten Programmversion korrekt erledigt.

So stellt man sich als Anwender die Frage, wozu man überhaupt noch zu linearen S-Parameter-CAD-Programmen wie „PUFF“ greifen soll, wenn PSPICE ein solcher Alleskön-

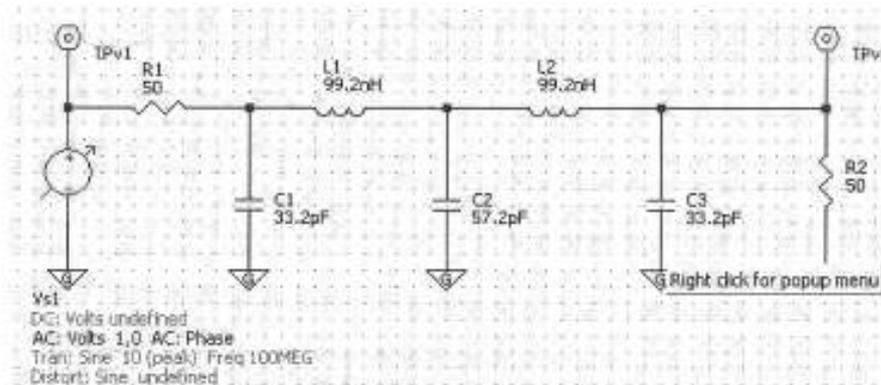


Bild 3:
Ein alter Bekannter:
110 MHz-Tschebyschev-Tiefpass.
Hier mit zwei
zusätzlichen
Spannungs-
markern

ner ist. Die Lösung ist im Prinzip recht einfach und zeigt, dass in der Praxis beide Methoden oder sogar eine Kombination aus beiden oder erst etwas ganz Neues wie „Harmonic Balance“ zum gewünschten Ziel führen. Das Ziel ist nämlich die möglichst exakte Übereinstimmung von Simulation und aufgebauter Schaltung bei unterschiedlichen Aussteuerungen - und bei PSPICE bilden die Modelle und ihre damit erzielbaren Genauigkeiten immer noch den Stolperstein, an dem pausenlos herumgeschliffen wird. Wenn man z. B. an Mikrostreifenleitungen und ihr Drumherum denkt: das sind alles Dinge, die viel leichter in der "Frequency Domain" behandelt oder einbezogen werden und mit einem Modellvorrat für PSPICE sieht es da (noch) nicht so gut aus.

Mit etwas Nachdenken wird dann auch die Stärke der Entwürfe mittels S-Parametern sofort klar: diese basieren meist auf gemessenen und in Tabellenform gefassten Informationen (Siehe die von jedem Hersteller publizierten S-Parameter-Touchstone-Files) und liegen deshalb mit ihren Voraussagen näher an der Praxis. Die Simulationsergebnisse werden sofort an der fertig aufgebauten Schaltung mit Network-Analysern kontrolliert und moderne Microwave-CAD-Programme besitzen heute sogar eine Schnittstelle, über die alle Messergebnisse wieder in die Simulation eingespielt werden können. In einer nächsten Runde kann dann verbessert und

so recht flott das angestrebte Entwicklungsziel erreicht werden. Allerdings geht es wiederum nicht ohne PSPICE, wenn z. B. das Übersteuerungsverhalten einer Sender-Endstufe bei 2,3 GHz ermittelt oder ein Schaltenteil entworfen werden soll....

4. Wie schaffe ich leicht und günstig den PSPICE-Einstieg?

Was heute auf dem Markt an Simulations-Software angeboten wird, verdient ausnahmslos das Prädikat „ausgezeichnet“, denn nach 30 Jahren Entwicklungszeit wurde sowohl bei der Simulationsgenauigkeit wie auch beim Eingabe- oder Ausgabekomfort ein sehr hoher Standard erreicht. Da gibt es einfach keinen Schund mehr!

Die Kehrseite dieses hohen Levels ist einerseits der hohe Kaufpreis, der mühelos das Niveau eines neuen Oberklassewagens erreicht. Andererseits sind die modernen Programme mit derart vielen Optionen und Funktionen ausgestattet, dass entweder der Bildschirm sehr schnell überladen wirkt oder man in unzähligen Menüs und Untermenüs wählen muss. All dies ermutigt einen Einsteiger nicht gerade sehr....



Die Software-Firmen geben sich bei der Preisgestaltung durchaus Mühe und bieten entweder kostenlose 30-Tage-Testversionen oder eine im Leistungsumfang reduzierte „Studentenversion“ an. Auf der anderen Seite wird die Verwendung von Raubkopien oft sehr energisch verfolgt (bei den Preisen kein Wunder!).

Die Studentenversionen haben sicher ihre Daseinsberechtigung, aber bei ernsthafter Arbeit klemmt es recht schnell: meist wird die Anzahl der maximal verwendbaren Bauteile begrenzt, oder der belegbare Arbeitsspeicher eingeschränkt oder es sind wichtige Funktionen bzw. Bauteile nicht freigegeben.

Beim Autor waren es eine ganze Reihe verschiedener Auslöser für die Suche nach einer alternativen Simulationssoftware:

a. Die Lehrerkollegen wünschten sich ein Programm für den Unterricht, das billig oder gar kostenlos ist, trotzdem genau simuliert, nach kurzer Einarbeitung leicht bedient und ohne Gewissensbisse kopiert und an die Schüler oder Studenten weitergegeben werden kann.

b. Wie erwähnt, werden die Bauteilmodelle immer besser, folglich aber auch immer aufwendiger. Und so passierte es, dass schon bei der Simulation einer simplen Testschaltung (schneller OP als nichtinvertierender Verstärker) eine Studentenversion den Dienst mit der Bemerkung „memory restricted in this version“ komplett verweigerte. Also kam von meiner Seite noch der Wunsch dazu, dass die Alternative jederzeit auch komplizierteste Modelle ohne Protest verarbeitet und die Aufnahme weiterer Modelle in die Bauteilbibliothek nicht in einen abendfüllenden Prozess ausartet.

c. Zudem bat mich ein Funkfreund daraus wieder einen Artikel zu erarbeiten.

Also wurde eine Internet-Suche gestartet - aber nicht nur bei den in Frage kommenden Software-Firmen, sondern auch bei denjenigen Leuten, die man vom Hobby oder von Problemdiskussionen her per Email irgendwo auf dem Globus kennt. Das Ergebnis war

recht überraschend, da die angesprochenen Fragen offensichtlich auch Andere bewegen.

Auf Empfehlung hin stieß ich auf:

4.1. Die Software „5Spice“

Es kristallisierte sich heraus, dass dieses mehrfach empfohlene Programm offensichtlich von jemand geschrieben wurde, der sehr viel mit Schaltungssimulation zu tun hat und sich über die Kompliziertheit, sowohl bei der Einarbeitung, als auch bei der Bedienung der großen Programmpakete ärgerte. Deshalb nahm er eine original „Berkeley-SPICE-Machine“ und packte sie in eine sehr leicht bedienbare Windows-Oberfläche. Lieferbar sind zwei Versionen als Download aus dem Internet (<http://www.5spice.com>):

a. Das kostenlose „Grundmodell“ kann nach der Installation sofort eingesetzt werden. Sehr originell ist hier die Begrenzung ausgeführt, denn es wurde die maximale Zeichenfläche für den Eingabeschaltplan auf eine „halbe DIN-A4-Seite“ beschränkt.

b. Nach einer Lizenzierung für etwa 150 US-\$ stehen nicht nur eine maximale Zeichenfläche von DIN-A2, sondern auch die Rauschsimulation, die Verzerrungsanalyse, die Monte Carlo-Bauteilstreuungs-Analyse, mehr Digitalbausteine und andere Dinge zusätzlich zur Verfügung. Außerdem gibt es jetzt Support per Email.

Am besten lernt man bekanntlich Neues kennen, wenn man damit selbst umgeht. Deshalb wird die Leistungsfähigkeit und Bedienerfreundlichkeit nachfolgend an einigen ausgewählten Beispielen getestet und demonstriert.

Wer sich intensiver in das Programm einarbeiten will, der lade sich aus der Homepage des Autors (siehe letztes Kapitel) ein über 50 Seiten starkes Tutorial mit dem Titel „Freude an PSPICE-Schaltungssimulation“ herunter, das für interessierte Kollegen, Schüler, Studenten und Funkfreude geschrieben wurde. Darin reicht die Spanne der simulierten Schaltungsbeispiele vom RC-Tiefpass bis zu

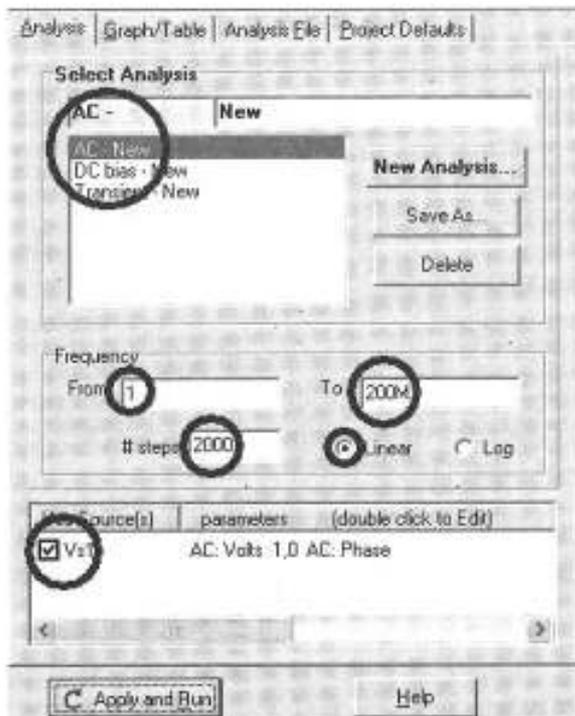


Bild 4: AC-Sweep-Programmierung: Bitte jede Einstellung sorgfältig kontrollieren

verschiedensten Gleichrichterschaltungen sowie vom zweistufigen Transistorverstärker bis zu den üblichen DC/ DC-Konvertern.

4.2. Einstieg über eine wohlbekannte Sache: 110 MHz-Tiefpass

4.2.1. Erstellen des Schaltplans

Nichts geht über praktisches Üben, deshalb gilt nun: „Bitte schon mal den PC einschalten“.

Den Anblick von Bild 1 sollten UKW-Berichte-Leser eigentlich kennen, denn es ist der bekannte Tschebyschef-Tiefpass mit den Daten:

Grenzfrequenz = 110 MHz

Ripple = 0,1 dB

Filtergrad $n = 5$

$Z = 50 \text{ Ohm}$

und mit den Bauteilwerten:

$C1 = C3 = 33,2 \text{ pF}$

$C2 = 57,2 \text{ pF}$

$L1 = L2 = 99,2 \text{ nH}$

Dieser wurde schon einige Male eingesetzt und in dieser Form liefert das kostenlose Filterprogramm „fds.zip“ alle Informationen, die man nachfolgend braucht.

Nach erfolgreicher Installation und Start von „5Spice“ wird man mit einem Bildschirm nach **Bild 2** empfangen. Wichtig ist nun die Menüleiste am rechten Bildrand. Im Bild ist schon eingetragen, wie man an alle Bauteile samt Massezeichen, Spannungsquellen und Mess-Marker herankommt, deshalb soll es gleich ans Zeichnen und Verdrahten gehen (**Bild 3**).

Mit dem Schaltungseditor geht das wirklich sehr gut und man muss lediglich wissen, dass die zum sauberen Zeichnen notwendigen Optionen (Drehen, Spiegeln, Bezeichnung verschieben..) nach einem rechten Mausklick auf ein Bauteil in Form eines Auswahlimenüs zur Verfügung stehen. Den rechten Mausklick braucht man auch dann, wenn man ein Bauteil erfolgreich geholt und abgesetzt hat und nun das Teil am Cursor wieder los werden will.

Und noch ein Tipp:

Nur mit „WIRES“ kann man elektrisch leitende Verbindungen im Schaltplan erzeugen! Es gibt nämlich zum „normalen“ Zeichnen auch „Lines“, aber die dienen nur zur Dekoration. Kreuzen sich zwei Leitungen und man möchte sie verbinden, braucht man die Funktion: „JUNCTION“.

Hat man auch noch alle Attributfelder der Spannungsquelle aus dem Weg geschoben und jedes Bauteil, nach einem Doppelklick auf sein Schaltzeichen durchnummeriert, sollte der Bildschirm wie Bild 3 aussehen. Bitte alles nochmals genau mit dem eigenen Entwurf vergleichen, bevor es weitergeht!

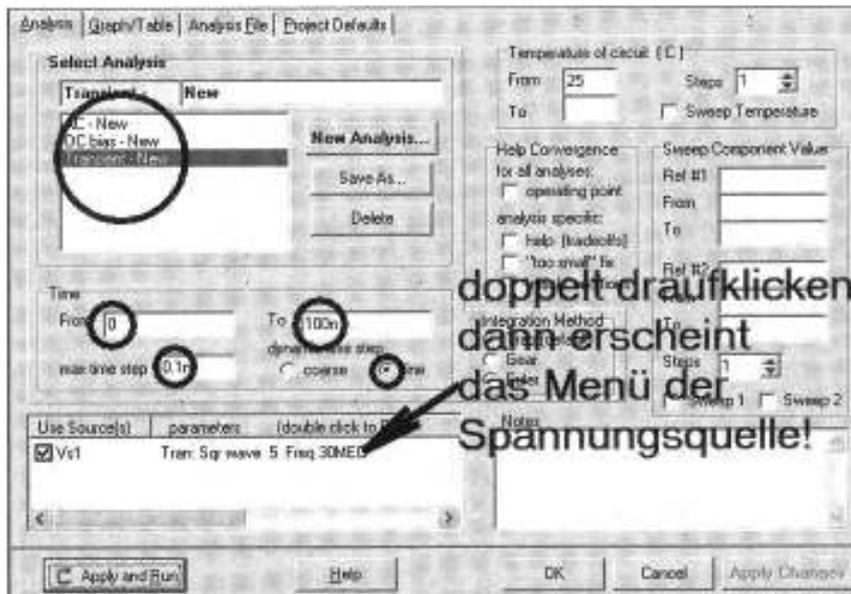


Bild 7:
Programmierung
einer Transienten-
analyse für die Zeit
von 0 bis 100 ns

d. Die linke vertikale Achse soll den Bereich von -6 bis -6,5 dB zeigen. Dieser Bereich soll als Raster mit 5 Kästchen dargestellt werden, jedes Kästchen ist wieder fünffach unterteilt.

e. Es soll die „Magnitude in dB“ angezeigt werden.

Ist alles korrekt eingestellt, drückt man auf „Apply and Run“. Beim anschließend auftauchenden Ergebnis kann man einen Cursor auf die Tschby-schef-Wellen des Filters legen und ihn mit der Maus hin- und herziehen. Stets wird der gerade aktive Kurvenpunkt mit dem dazugehörigen Frequenz- und Amplitudenwert eingeblendet (Bild 6).

4.2.3. Transienten-Analyse mit verschiedenen Eingangssignalen

Ein Ergebnis nach Bild 6 hätte man sicher auch mit PUFF so oder sogar besser hinkommen. Interessanter wird es, wenn man Vorgänge im Zeitbe-

reich simuliert, denn dafür ist PSPICE schließlich gemacht.

Man legt beispielsweise ein TTL-Signal ($U_{min} = 0V / U_{max} = +5V$) mit der Frequenz $f = 30\text{ MHz}$ an den Eingang des zu untersuchenden Filters und stellt die Simulation um:

1. Schritt
 Mit Taste F8 holt man die Karteikarte „Analy-

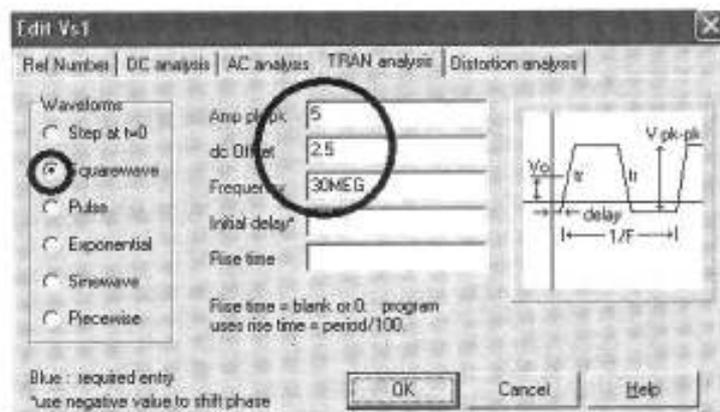


Bild 8: So einfach lässt sich für die Eingangsspannung eine andere Kurvenform wählen!

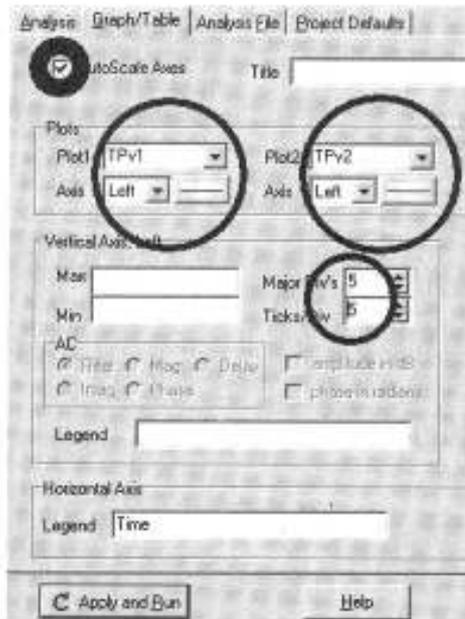


Bild 9: Auch die Ergebnisausgabe muss etwas umgestellt werden

sis" (Bild 7). Darauf aktiviert man „Transient - New“, wählt einen Simulationsbereich von 0 bis 100 ns und „Dynamic Time Steps = fine“ mit einem „max time step = 0,1ns“.

2. Schritt

Im unteren linken Eck klickt man auf den Eintrag der Spannungsquelle und erhält dadurch das Menü nach Bild 8. Da sieht man eine ganze Reihe von vorprogrammierten Kurvenformen und bei der Wahl „Squarewave“ gibt man einfach die Attribute „Amp pk-pk = 5V / dc Offset = 2,5 V / Frequency = 30 MEG“ ein.

3. Schritt

Nach dem Bestätigen mit OK braucht man erneut die Karteikarte „Graph/Tables“ (...notfalls wird vorher nochmals F8 gedrückt...). Ganz wichtig ist nun die Aktivierung von „AutoScale Axes“, die Einstellungen für die Plots TPV1 und TPV2 sowie für die Rasterteilung können vom vorigen Beispiel her übernommen werden (Bild 9).

4. Schritt

Jetzt ein Klick auf „Apply and Run“ und dann sollte Bild 10 auf den Schirm flattern. Da kann PUFF natürlich nicht mehr mithalten!

Wer jetzt wissen will, wie das bei einer Eingangsfrequenz von 50 MHz aussieht, der wechselt zum Schaltbild und klickt einfach doppelt auf das Symbol der Spannungsquelle. Auch damit bekommt man das Span-

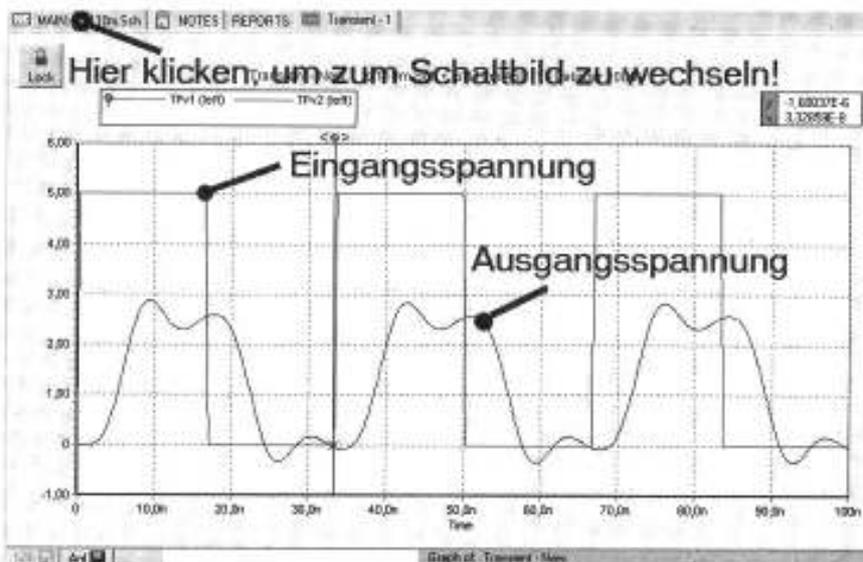


Bild 10: Ein Scope-Bildschirm könnte nicht schöner sein....

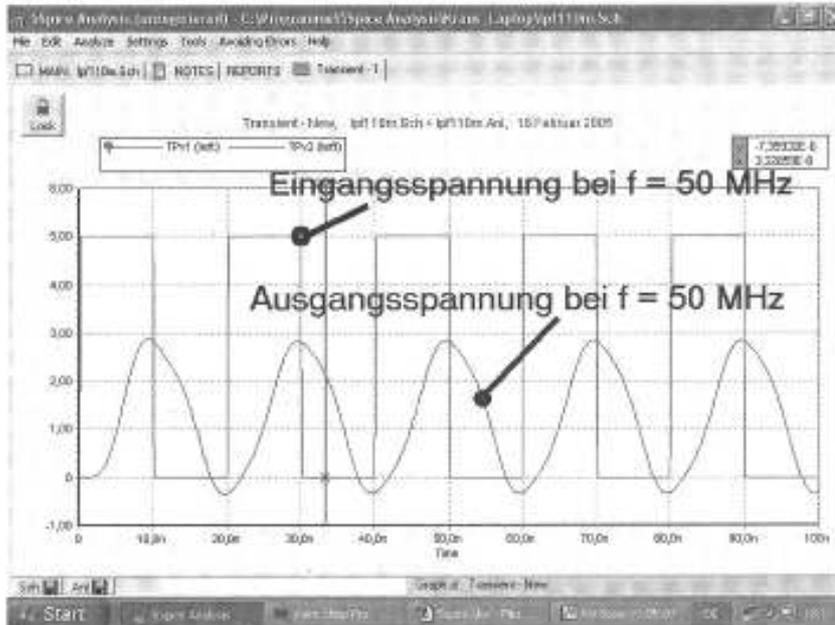


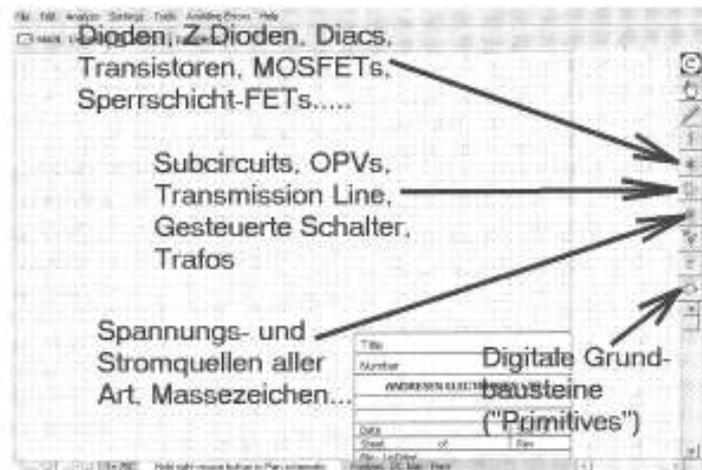
Bild 11:
Bei 50 MHz fehlen nun natürlich am Ausgang langsam die Oberwellen des Rechtecks

nungsquellen-Attributmenü und nach einer Änderung der Frequenz auf 50 MHz hat man bei einer erneuten Simulation **Bild 11** vor sich; dauert nur Sekunden!

Will man nur das grundsätzliche Verhalten einer ausgetüftelten OP-Schaltung kennenlernen, dann reicht zunächst eine Simulation mit dem im Menü bereitstehenden „Linearen OP-Modell“. Da braucht man sich nicht einmal um eine Versorgungsspannung zu kümmern und kann sich voll auf die Reaktionen der Schaltung konzentrieren. Bitte mal selber ausprobieren!

4.3. Beispiel mit Einbindung eines neuen Modells

Wer sich **Bild 12** ansieht, findet sich bei der Suche nach den Bauteilen für seine Schaltung sicher gut zurecht. Sehr viele Modelle werden mitgeliefert, aber beim raschen Fortschreiten der technischen Entwicklung ist man doch immer häufiger auf die Modellsuche im Internet angewiesen. Deshalb soll an einem weiteren Beispiel das Vorgehen in einem solchen Fall gezeigt werden.



4.3.1. Untersuchung eines Gainblocks mit dem Operationsverstärker „OPA355“

Bild 12: Das steckt hinter den übrigen Buttons der Bauteil-Menüleiste!

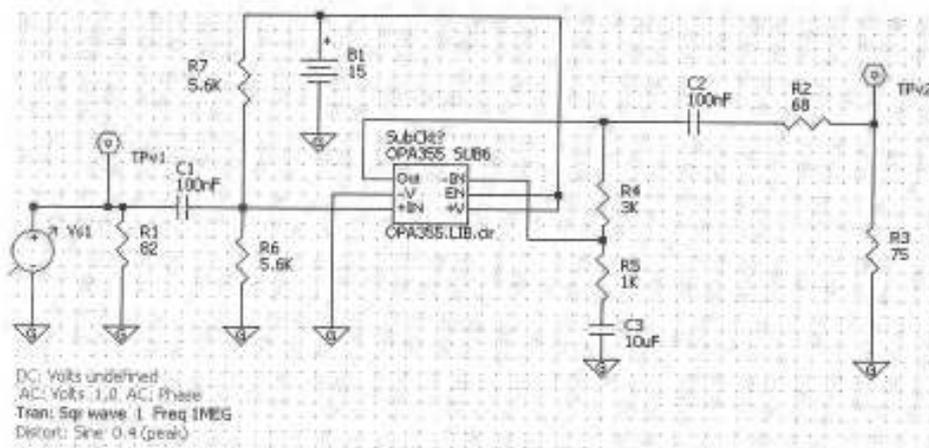


Bild 13:
Eine praktische
HF-Schaltung
wird simuliert:
Gainblock für
 $Z = 75 \text{ Ohm}$

Sobald es dagegen um gezielte Entwicklungen für einen konkreten Anwendungsfall geht, sieht die Sache ganz anders aus. Hier interessiert plötzlich das Verhalten bei Eingangssignalen mit verschiedenen Kurvenformen oder Frequenzen, der Frequenzgang, die Schwingneigung, die Einflüsse der Versorgungsspannung, das Einschwingen oder Überspringen sowie der Klirrfaktor am Ausgang, das Rauschen..... und das alles bei unterschiedlichen OP-Typen. Deshalb sollte die Wiederholung der Simulation mit einem anderen OP eine Kleinigkeit sein.

Das Programm sieht hierzu die Verwendung von „Subcircuits“ vor, die das eingesetzte Bauteil exakt beschreiben und selbst wieder (man denke an einen OP mit seinen vielen integrierten Transistoren, Dioden und FETS!) eine sehr aufwendige Schaltung mit vielen Einzelmodellen darstellen. Für die meisten modernen Bauteile werden diese Subcircuit-Beschreibungen in Form der SPICE-Modelle von ihren Produzenten bereitgestellt.

Betrachtet man die Schaltung nach **Bild 13** genauer. Man hat es hier mit einem OP in nicht-invertierender Schaltung zu tun, die folgende Eigenheiten und Daten aufweist:

a. Es wird nur mit einer Betriebsspannung (+15 V) gearbeitet.

b. Deshalb liegt am nichtinvertierenden Eingang („+IN“) die von einem Spannungsteiler erzeugte halbe Betriebsspannung. So erhält man eine Ausgangsgleichspannung mit derselben Amplitude.

c. Mit den beiden Gegenkopplungswiderständen $R4 = 3 \text{ k}\Omega$ und $R5 = 1 \text{ k}\Omega$ erreicht man vierfache Verstärkung

d. Der Eingangswiderstand der Gesamtschaltung wird durch $R1$ auf ca. $75 \text{ }\Omega$ gebracht.

e. Durch den Widerstand $R2 = 68 \text{ }\Omega$ erreicht man einen Innenwiderstand von ca. $75 \text{ }\Omega$.

f. Das Ganze stellt folglich einen „Gainblock“ mit $R_i = R_a = 75 \text{ }\Omega$ dar, der ein S_{21} von 6 dB besitzt.

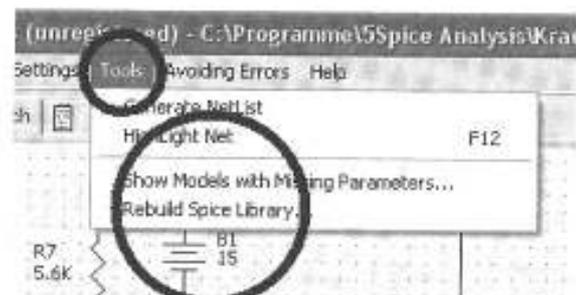


Bild 14: So wird nach dem Hinzufügen neuer Modelle der Bibliotheksindex neu aufgebaut

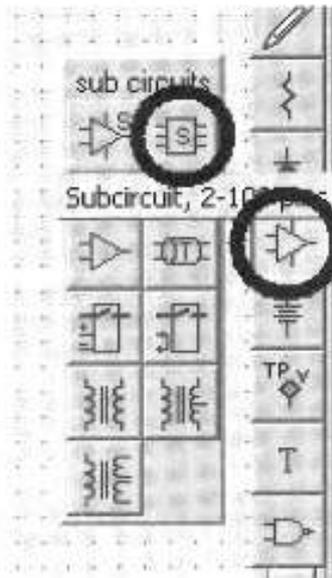


Bild 15: Hinter dem „OP-Button“ sind die Subcircuits versteckt

hinein muss die neue OPA355-Datei. Anschließend wird das Programm gestartet und im Menü „Tools“ auf „Rebuild Spice Library“ geklickt (**Bild 14**). Das ist schon alles, denn der Rest folgt nach dem Platzieren des Bauteils und läuft so ab:

1. Schritt

Ein Klick auf den „OPV-Button“ am rechten Bildrand gibt den Weg zum „2 bis 100 poligen Subcircuit“ frei (**Bild 15**). Der wird angeklickt, das Bauteil auf dem Schirm abgesetzt und mit einem rechten Mausklick der Platzierung abgeschlossen.

2. Schritt

Ein Doppelklick auf das Schaltzeichen öffnet das Attribut-Menü des Subcircuits (**Bild 16**). Zuerst bekommt darin dieses Bauteil den Namen „SubCkt1“ und anschließend sucht man in der Vorratsliste den hineinkopierten OPA355. Ein Klick auf seinen Namen in der Liste lädt die zugehörige SPICE-Modelldatei und man sieht darin gleich etwas sehr Wichtiges: die Zuordnung der Anschlusspins!

g. Die Schaltung sollte mindestens bis 30 MHz ohne Verstärkungsverlust einsetzbar sein.

h. Der verwendete OPA355 besitzt einen „ENABLE“-Anschluss. Wird dieser mit der Betriebsspannung verbunden, ist der Verstärker freigegeben. Legt man „EN“ dagegen an Masse, wird die Schaltung gesperrt. Damit lässt sich sehr einfach ein Anlogschalter oder sogar ein Eintakt-Mischer realisieren!

4.3.2. Ein neues Bauteil für die Bibliothek

Nach einem kurzen Internet-Besuch mit einer Suchmaschine ist man im Besitz des SPICE-Modells für den OPA355 und kopiert diese Datei in die „Library“ von „5Spice“. Dort befinden sich zwei Ordner: einer ist für „Dioden / Transistoren / FETS“ (= Grundmodelle) vorgesehen, der andere enthält nur „Subcircuits“. Dort

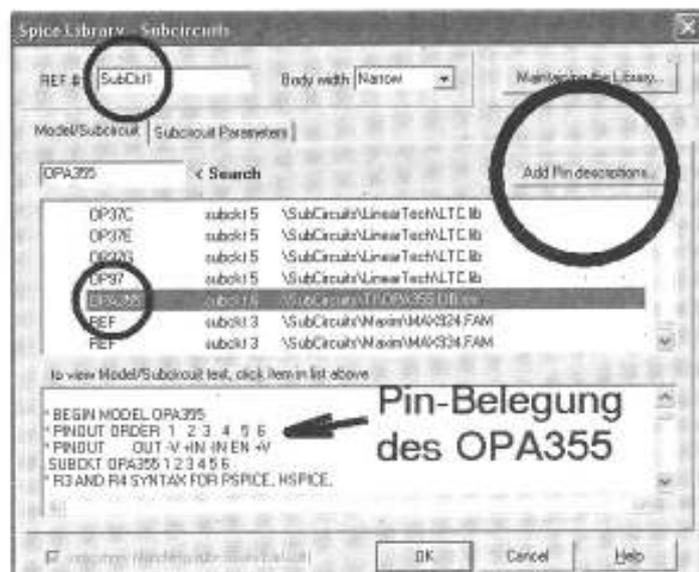


Bild 16: Erst kommt die Nummerierung, dann die Einbindung des Modells. Anschließend werden die Pins zugeordnet

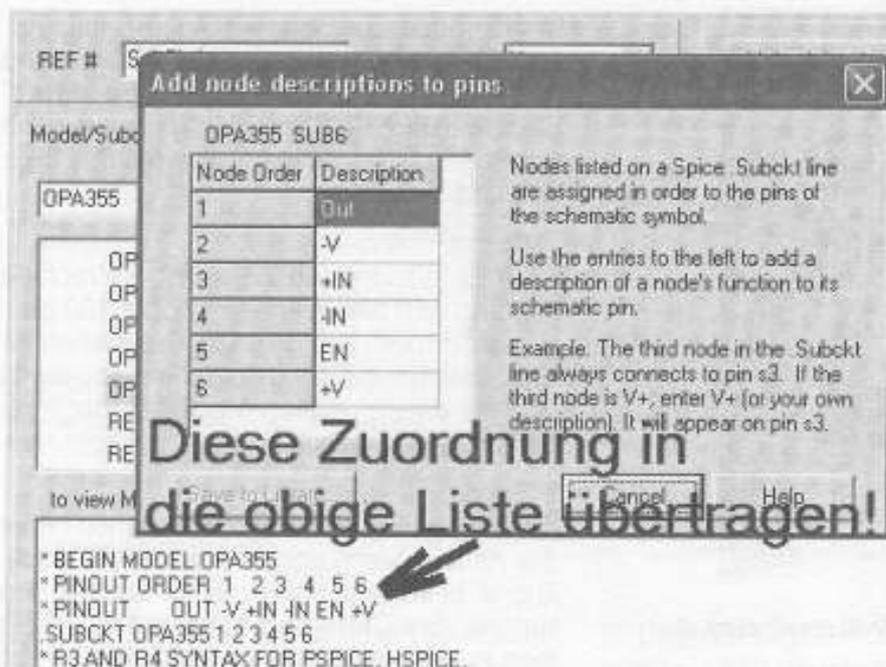


Bild 17:
Nur mit dieser korrekt ausgefüllten Liste kann man das Bauteil in der Schaltung korrekt verdrahten!

3. Schritt

Klickt man auf „Add Pin descriptions“, so braucht man darin nur noch die einzelnen Pin-Funktionen nacheinander in die auftauchende Liste zu übertragen (**Bild 17**).

- Fertig! -

4.3.3. Untersuchung der Schaltung

Jetzt muss man das Simulationsschaltbild entsprechend der Vorlage von Bild 13 erstellen und verdrahten. Anschließend wird mit F8 das Analysis-Menü aufgerufen und zuerst

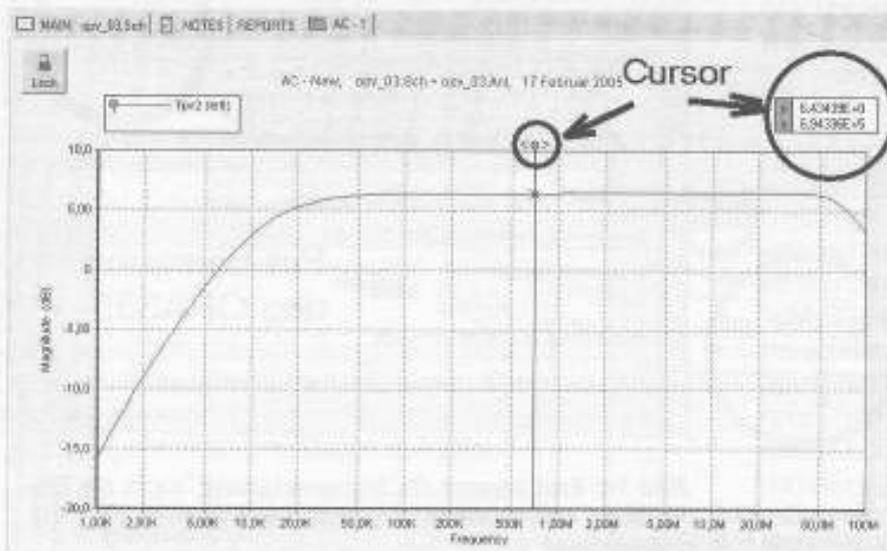


Bild 18:
Der AC-Sweep bis 100 MHz ist doch „Sehr zufriedenstellend!“

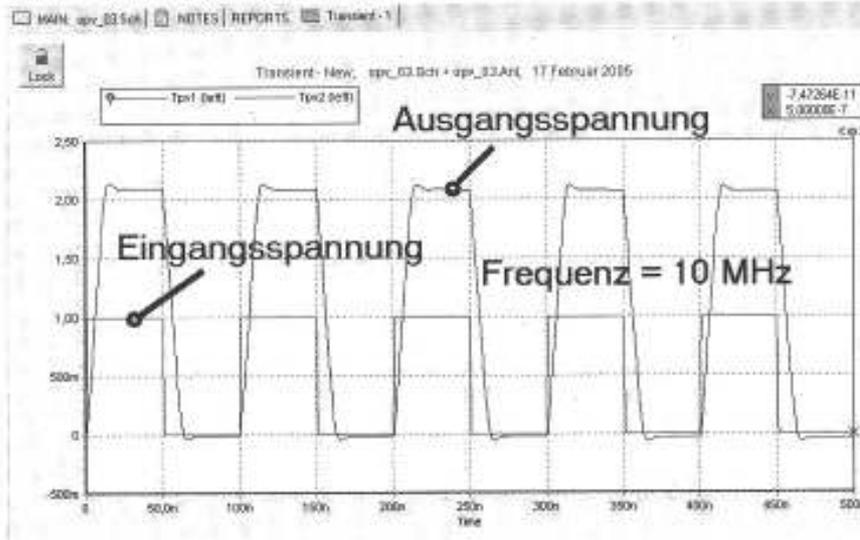


Bild 19:
Dieses Rechteck-
signal mit 1 Volt
Spitzenwert wird
noch korrekt
verarbeitet....

einmal ein AC-Sweep für den Frequenzbereich von 1 kHz bis 100 MHz programmiert (Siehe Kapitel 4.2.2.). Das Ergebnis ist in **Bild 18** zu sehen und man ist geneigt zu sagen: „gar nicht so übel!“.

Interessant ist nun die Reaktion auf ein rechteckförmiges Eingangssignal mit $U_{min} = 0$ V,

$U_{max} = 1$ V, Frequenz = 10 MHz (Vorgehen: Siehe Kapitel 4.2.3.).

Das Verhalten des Gainblocks zeigt **Bild 19**; durchaus zufriedenstellend. Erst wenn man die Eingangsspannungsdoppelte (**Bild 20**) und hinterher die Frequenz verdreifacht (**Bild 21**), geht der Schaltung deutlich „die Luft aus“.

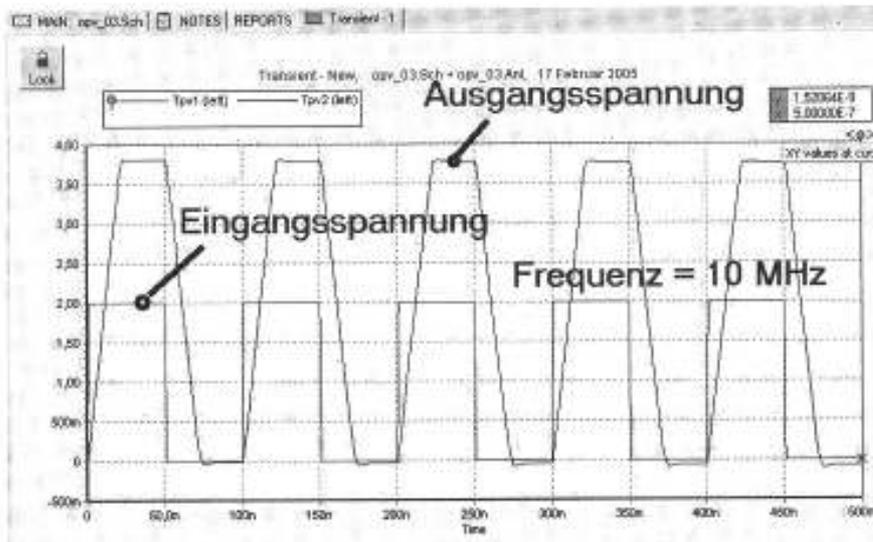


Bild 20:
...aber bei 2 V
Spitzenwert wird
es schon eng...

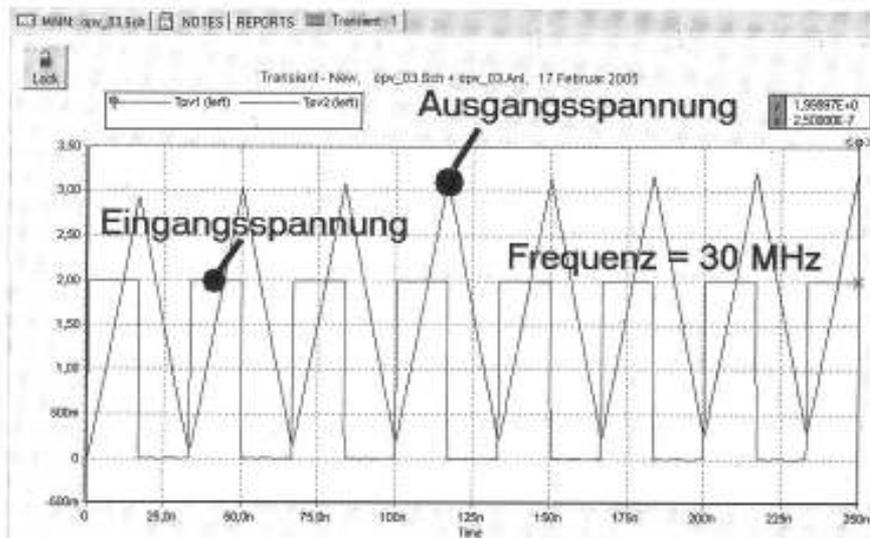


Bild 21:
...und bei Erhöhung
der Frequenz auf
30 MHz schwächt
der OP doch schon
deutlich!

An dieser Stelle kann man nur empfehlen: bitte selber noch etwas weiterprobieren!

5. Zusammenfassung

PSpICE ist heute ein unverzichtbares Werkzeug am Arbeitsplatz des professionellen Schaltungsentwicklers, aber leider bewegen sich auch die Kosten für die Software in solchen Regionen, die nur Entwicklungsabteilungen aufbringen können. So soll dieser Artikel ermutigen, auch bei kleiner Hobbykasse den eigenen Projekten immer zuerst mit diesem modernen Hilfsmittel genau auf den Zahn zu fühlen, bevor überhaupt der Lötkolben angefasst wird.

Der Autor hat deshalb in seiner Homepage (www.elektronikschule.de/~krausg) eine gezippte „5Spice-CD“ zum kostenlosen Download bereitgestellt. Sie enthält

- den letzten Stand der 5Spice-Software;
- das erwähnte selbstverfasste 50-seitige Einsteiger-Tutorial mit noch ausführlicherer

Einführung und vielen Simulationsbeispielen;

c. ein originales „Berkeley-Spice-Reference-Manual“ aus dem Internet;

d. die übliche Sammlung der "FAQ-Fragen" aus der 5Spice-Homepage;

e. die in den letzten Jahren zusammengetragene private Sammlung an SPICE-Modellen (Vorsicht: ungezippt sind das schon mehr als 120 Megabyte an Daten....).

Die verschiedenen aufgeführten Sachen können natürlich auch einzeln angesehen oder auf den heimischen Rechner geholt werden. Damit sind nun alle Ausreden nichtig, die bisher gegen den Einsatz von PSpICE sprachen.

Viel Erfolg!