



Gunthard Kraus, DG 8 GB

Möglichkeiten und Grenzen der Schaltungssimulation für Funkamateure

Überarbeitete Fassung des Vortrags
zur UKW-Tagung 2009 in Bensheim

1. Sinn und Zweck der Schaltungssimulation

Jeder, der bereits Schaltungen selber entwickelt hat kennt die Fragen, die einen bei dieser Arbeit quälen:

- Wird das überhaupt so funktionieren, wie ich mir das vorstelle?
- Schaffe ich die geforderten Daten?
- Wie finde ich das Optimum?
- Welche Effekte - an die ich noch gar nicht denke! - können das Ergebnis beeinflussen?
- Wird die Schaltung „sauber“ arbeiten und z.B. nicht schwingen?
- Wie wirken sich Bauteile- und Fertigungstoleranzen aus?
- Welche Alternativen habe ich, wenn der vorgesehene Weg in eine Sackgasse führt?
- usw.

Die Schaltungssimulation soll und kann

die meisten dieser gestellten Fragen beantworten, bevor überhaupt ein Prototyp gefertigt oder ein Lötkolben angefasst wird. Allerdings sind genaue Kenntnisse des gewünschten Schaltungsverhaltens unbedingt erforderlich. Das Arbeiten mit der Simulation sowie irgendwelche Änderungen hängen **IMMER** am Entwickler - kein Programm kann ihn und sein Fachwissen ersetzen (selbst wenn es die Werbung manchmal so suggeriert)!

Der Einsatz von Schaltungssimulation bedeutet immer, dass damit kostbare Arbeitszeit gespart, eine alternative Idee sehr schnell geprüft oder das Optimum aus einem Entwurf herausgeholt werden soll.

2. Einteilung der Simulator-Programme

Man unterscheidet:

- a.) Simulatoren, die im *Zeitbereich* (= in der Time Domain) arbeiten. Das sind

grundsätzlich alle SPICE und PSPICE-Programme. „SPICE“ selbst steht für „Simulation Program with Integrated Circuit Emphasis“ und „PSPICE“ ist dann einfach „SPICE for the PC“.

Sie liefern aufgrund ihres Prinzips (= Zerlegung des ansteuernden Signals in sehr kurze, aber praktisch lineare Teilstücke) auch alle Informationen über Anschwing- und Startvorgänge sowie über das Verhalten beim Anlegen von Signalen mit beliebiger Kurvenform. Zusätzlich werden alle Nichtlinearitäten in der Schaltung erfasst.

Weitere Optionen sind der AC-Sweep (= das Wobbeln zur Bestimmung eines Frequenzganges), die Rauschsimulation und die FFT (= Fast Fourier Transformation) zur Analyse eines Signals in der „Frequency Domain“ („Spektraldarstellung“). Damit bieten sie die umfassendsten Möglichkeiten, aber die Simulationsgenauigkeit hängt stets von der Qualität der „SPICE-Modelle“ ab. Diese werden entweder mit der Software mitgeliefert oder von Bauteil-Produzenten zur Verfügung gestellt. Letztlich legen sie fest, wie genau Simulation und Wirklichkeit übereinstimmen!

b.) Simulatoren, die im *Frequenzbereich* arbeiten, verwenden oft die S-Parameter (...manche beispielsweise auch die Y-Parameter und rechnen dann auf S-Parameter um). Dadurch können zur Ergebniskontrolle - aber auch zur Eingabe von Bauteil-Eigenschaften in die Simulation - die bekannten vektoriellen Netzwerk-Analysatoren eingesetzt werden. Da hierbei oft keine durch Formeln beschriebenen Modelle, sondern Dateien mit GEMESSENEN Eigenschaften verwendet werden, erreicht man damit im HF- und

Mikrowellengebiet (bis über 100 GHz) eine gute Übereinstimmung zwischen Simulation und gefertigtem Prototyp. Ein Nachteil ist hierbei, dass es sich grundsätzlich um eine lineare Simulation (sozusagen im Kleinsignalbetrieb) handelt und Informationen über Verzerrungen bei größerer Aussteuerung meist nur mit patentierten Tricks des Programmanbieters gewonnen werden können (z.B. mit „Harmonic Balance“).

c.) *EM-Simulatoren* gehen auf die Form, die Verteilung und die Beziehungen der elektrischen und magnetischen Felder los und dienen zur Analyse von „Strukturen“. Sie können in zwei Gruppen eingeteilt werden:

Für Gebilde, die aus *Flächen* bestehen (Patchantennen, Koppler, Stripline-Filter etc.) zerlegt man die Gesamtstruktur in sehr kleine rechteckige oder dreieckige „Zellen“. Für jede Zelle berechnet man die Einzelfelder und addiert am Ende alles auf.

Für *Drahtantennen (oder Draht ähnliche Gebilde)* heißt das Stichwort dagegen „NEC“. Auch hier zerlegt man die Struktur in kleine Elemente - in diesem Fall eben kurze Drahtstücke mit linearer Strom- und Spannungsverteilung. Diese Elemente lassen sich leichter untersuchen - und man erhält das Gesamtverhalten der Anordnung durch Addition aller Einzelberechnungen.

3. SPICE-Simulation für den Amateur

Hier hat sich die Situation sehr entspannt, denn die Einschränkung „für

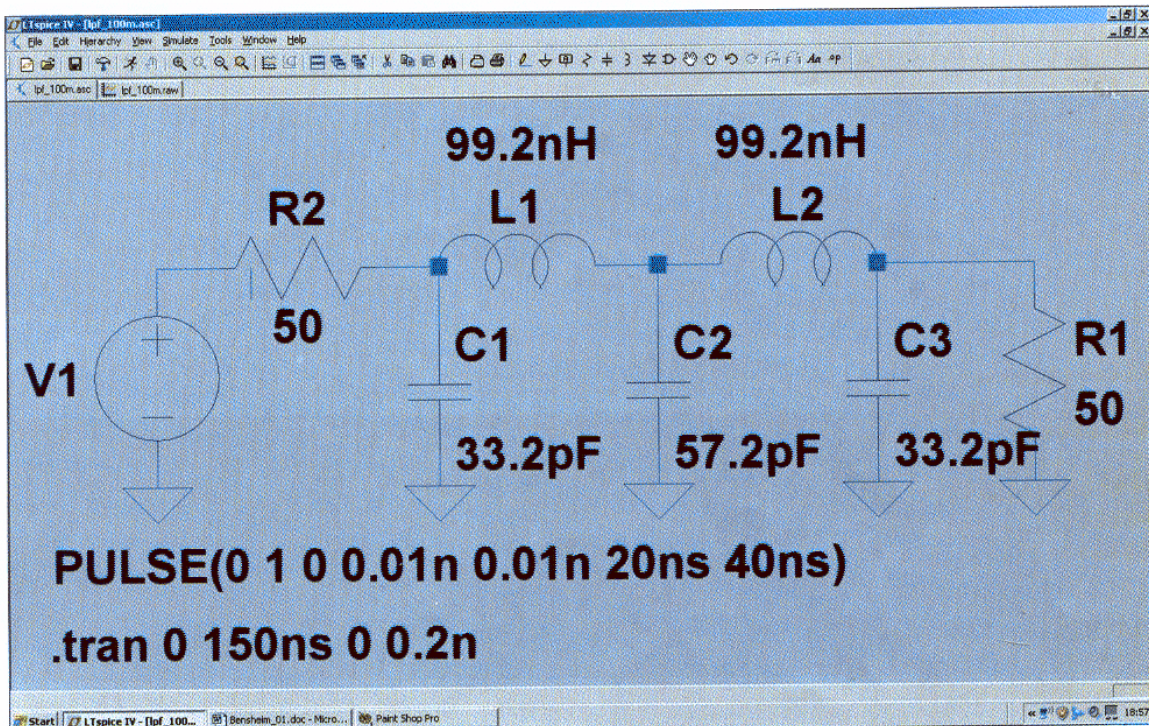


Bild1: So geht es auch: ein Bildschirm ohne Schnörkel, nur mit der Schaltung und den erforderlichen Texteingaben (Properties, Controls und Commands)

den Amateur“ bedeutet meist Einschnitte beim Geldbeutel. Es ist kein Problem, für eine Simulationssoftware bis zu 50.000,- € oder \$ auszugeben - das sind dann aber „riesige Maschinen“, mit denen man gleich die zugehörige Leiterplatte entwerfen und das EMV-Verhalten untersuchen kann usw. Deshalb ersetzt man als Privatmann einfach Geld durch Zeit. In der Praxis bedeutet das einen Verzicht auf Komfort sowie eine Trennung von Simulation und Platinen-CAD.

Für den Platinenentwurf gibt es bereits erschwingliche Lösungen, die gleich die EMV-Simulation und auch eine einfache SPICE-Simulation mit einschließen (Beispiel: „Target“).

Zwar liefern die großen PSPICE-Software-Hersteller auch immer kostenlose Test- oder Demoversionen ihrer Produkte, aber nicht ohne Einschränkung. Wer zum ersten Mal mit den Zähnen knirscht, weil mitten in der Arbeit die Meldung: „alle Bauteile aufgebraucht“ oder „Memory restricted in this version“ erscheint, der lässt sehr schnell die Finger davon. Vor allem deshalb, weil man diesen angefangenen Entwurf leider nie in eine aufwendigere (und teurere) Programmvariante übernehmen kann.

Ein Programm für Amateurzwecke (oder Lehrzwecke!) soll dagegen

- a) möglichst wenig oder gar nichts kosten

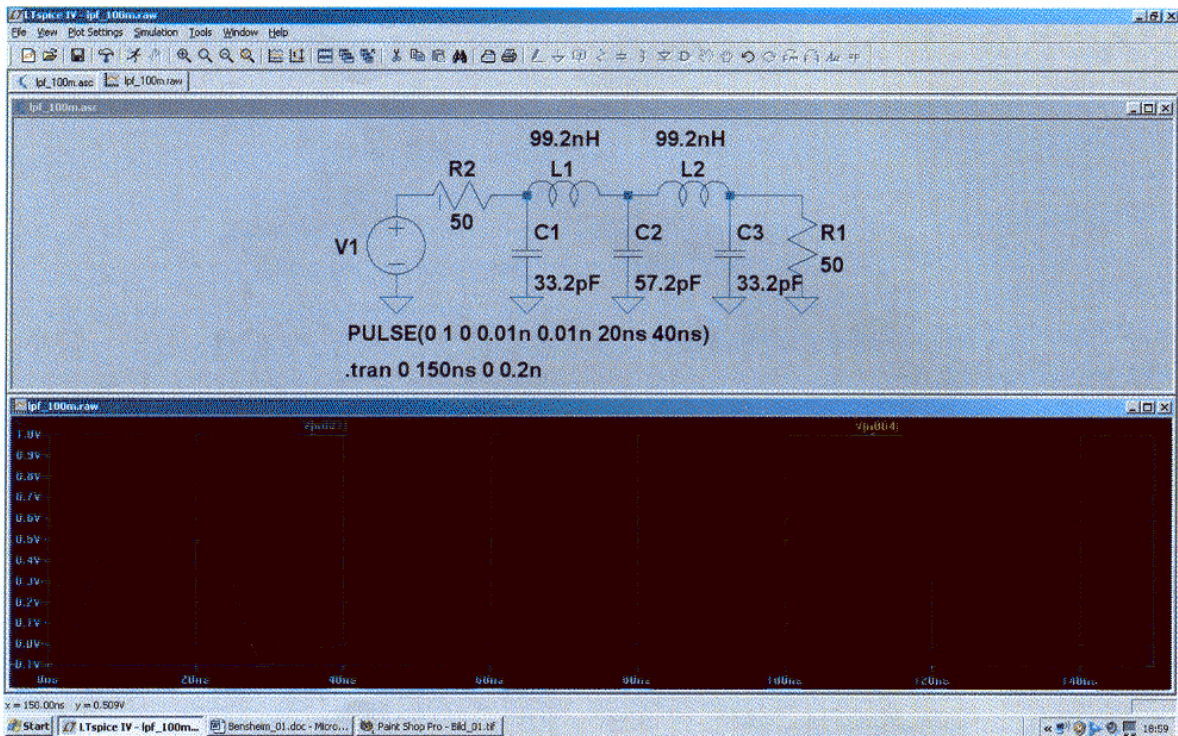


Bild 2: Das Ergebnis der Simulation. Zusätzlich zum Diagramm wird stets die Schaltung eingeblendet und darin wird der gewünschte Messpunkt angeklickt

- b) keine Einschränkungen oder Begrenzungen aufweisen, also voll leistungsfähig sein;
- c) eine problemlose Übernahme neuer Bauelementmodelle aus dem Internet ermöglichen.
- d) Anbindung an eine Fangemeinde (Community) aufweisen, damit man bei Fragen oder Problemen nicht alleingelassen oder auf einen teuren Support angewiesen ist.

Glücklicherweise hat sich hier die Firma Linear Technologies verdient gemacht, da sie für ihre Kunden und deren Entwicklungsingenieure die kostenlose PSPICE-Software „LTSpice IV“ auf ihrer Home-

page [1] zur Verfügung stellt. Da gibt es wirklich keine Begrenzung und sogar die Registrierung vor dem Download kann man verweigern. Die Einbindung neuer Modelle aus dem Internet ist problemlos möglich und es gibt dafür extra einen leicht bedienbaren „Symbol Editor“.

Als einzige Kehrseite hat man eine sehr schlichte Bedieneroberfläche vor sich und die Bedienung ist eine gewöhnungsbedürftige Mischung aus Tastenkommandos, Mausclicks und direkt einzutippenden Anweisungen.

Hinweis: Auf der Homepage des Autors [2] kann für dieses Programm ein Tutorial (wahlweise in Deutsch oder Englisch) kostenlos heruntergeladen werden. Es

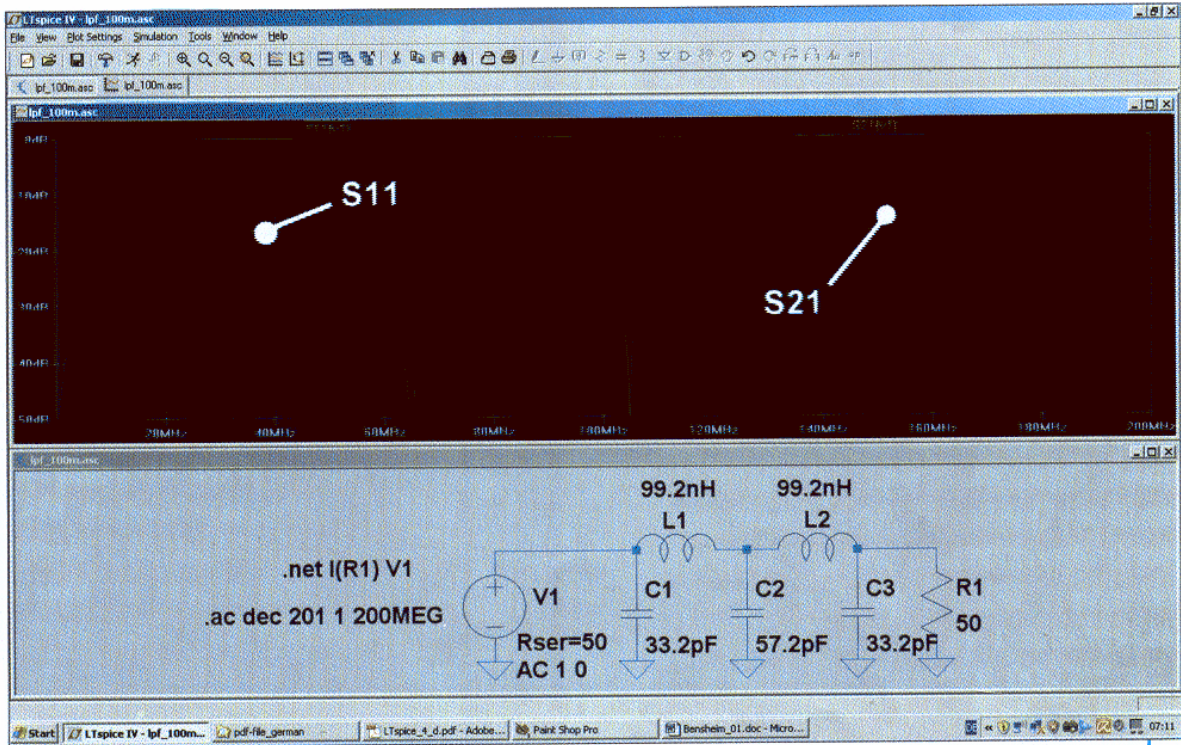


Bild 3: Die Simulation der S-Parameter erfordert eine zusätzliche Spice-Direktive und ein neues „Simulation Command“

enthält eine umfangreiche Sammlung an Praxisprojekten zur Einarbeitung!

Der Einsatz von LTSpice soll am Beispiel eines 110 MHz-Tschebyschef-Tiefpasses für einen Systemwiderstand von 50 Ω gezeigt werden, dem ein 25 MHz-Rechtecksignal zugeführt wird. Dazu muss der Anwender erst den Bildschirm nach **Bild 1** hinbekommen.

Neben der Zeichenarbeit gilt es zusätzliche Textzeilen zu erstellen, wozu eine Erläuterungen nötig ist:

Die Zeile „Pulse...“ unterhalb der Schaltung programmiert bei der Spannungsquelle ein Rechtecksignal mit $U_{\min} = 0V$ und $U_{\max} = 1V$. Die Anstiegs- und Abfall-

zeiten betragen je 0,01 ns. Hier wird mit einer Pulslänge von 20 ns und einer Periodendauer von 40 ns gearbeitet. Das Signal beginnt sofort beim Zeitpunkt $t = 0 s$.

Die Zeile „tran“ schreibt vor, dass in der „Time Domain“ der Zeitabschnitt 0 bis 150 ns simuliert wird, die Daten sofort ohne Verzögerung weitergegeben werden und der größte zugelassene „Zeitschritt“ (= Time Step) 0,2 ns betragen darf.

Nach dem Start der Simulation wird der Bildschirm geteilt und es erscheint zusätzlich ein leeres Ergebnisdiagramm für die vorgesehene Simulationszeit (**Bild 2**). Nun klickt man mit der Maus an den Punkt der Schaltung, bei dem der Spannungs-

oder Stromverlauf interessiert und sofort wird diese Kurve im Diagramm erscheinen (hier sind es der Eingangs- und Ausgangspunkt der Schaltung). Durch verschiedene weitere Mausklicks und Untermenüs kann man sich dann das Ergebnis so „hintrimmen“, wie es einem am besten gefällt (z.B. mehrere Einzeldiagramme, andere Wertebereiche, andere Farben für die Kurven...)

Noch ein Blick auf die S-Parameter-Simulation: Dazu muss erst der Innenwiderstand der Quelle (= $50\ \Omega$) in ihr „Property“-Menü übernommen und ein „AC-Sweep“ mit einem „SPICE Simulation Command“ programmiert werden (= Wobbeln von 1 Hz bis 200 MHz mit 201 Punkten pro Dekade, dekadischer Sweep). Die Spice-Direktive „net I(R1) V1“ bewirkt die Bereitstellung der S-Parameter. Das Endergebnis ist in **Bild 3** gezeigt.

4. S-Parameter-Simulationen für den Funkamateuer

4.1. Etwas Grundwissen über S-Parameter

Die Systembeschreibung und -berechnung geht ab etwa 1 MHz von Größen aus, die auch bei höchsten Frequenzen (heute mehr als 100 GHz) leicht messbar sind und man benützt deshalb ein völlig anderes Denkmodell:

Überall im System wird derselbe „Systemwiderstand“ verwendet (üblich sind $75\ \Omega$ in der Unterhaltungselektronik und Videotechnik, dagegen $50\ \Omega$ bei den meis-

ten anderen Gebieten). Er gilt für den Innenwiderstand der Speisespannungsquellen, die Ein- und Ausgangswiderstände der verwendeten Baugruppen, den Wellenwiderstand aller Verbindungskabel und für alle Abschlusswiderstände.

Der Kern dieser Sache ist also der Versuch der perfekten Leistungsanpassung (mit $R_i = R_o$) überall im kompletten System!

Durch geeignete Messgeräte (= Richtkoppler) misst man nun die Abweichungen des Widerstandes von diesem Idealfall am Eingang und Ausgang (Fachausdruck: „Ports“) und drückt sie durch „Reflektionsfaktoren“ aus (= S-Parameter S11 und S22).

Außerdem misst man die Ausgangsleistung, wenn man dem Eingang Leistung zuführt und drückt das durch die „Forward Transmission“ (= S-Parameter S21) aus.

Für die Rückwirkungen in einem Baustein vom Ausgang zurück zum Eingang gibt es noch die „Reverse Transmission“ (= S-Parameter S12).

Bei einem kompletten System sind dann viele Bausteine in Reihe geschaltet und mit Hilfe geeigneter Software lässt sich leicht das Gesamtverhalten bestimmen.

4.2. Einsatz von „PUFF“

Dieses altbekannte DOS-Programm, erhältlich bei [3], war wohl das „Standardprogramm der Funkamateure“ und hat auch nach fast 20 Jahren nicht ausgedient. Niedriger Preis, minimaler Speicherplatzbedarf, hohe Stabilität und Simulationsgenauigkeit, eine einfache

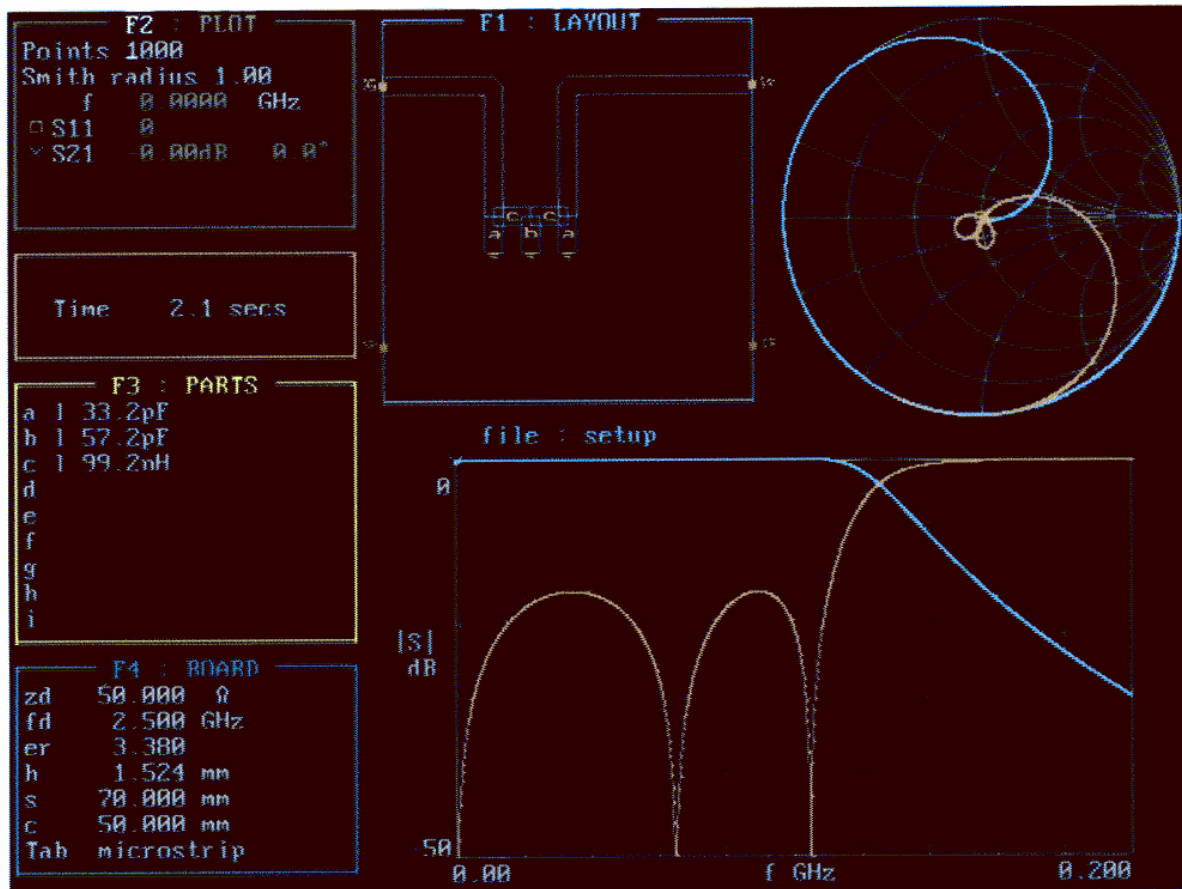


Bild 4: Der PUFF-Bildschirm bietet Eingaben, Werkstoffdaten und Simulationsergebnisse auf einen Blick

und einprägsame Bedienung über Tastenkommandos, das Vorhandensein der wichtigsten Grundmodelle und ein klarer übersichtlicher Bildschirm machen es zum „HF-Taschenrechner des Entwicklers“, mit dem man schnell neue Ideen austesten kann. Dazu wurde es immer wieder „aufgepäppelt“, gepflegt und sogar an Windows-XP angepasst.

Hinweis: die Homepage des Autors [2] enthält hierzu alle Unterlagen und Hilfen (einschließlich eines ausführlichen Katalogs der Bedienkommandos mit Erläuterungen und einer Anleitung bzw. einem

„PUFF_XP-Servicepack“ zum störungsfreien Betrieb unter Windows-XP).

Den gesamten Arbeitsplatz des PUFF-Anwenders bei der Simulation des 110 MHz-Tiefpasses sieht man in **Bild 4**:

Bauteil-Eingabe, Ergebnis-Ausgabe per Smithchart oder kartesischem Diagramm, Platinen-Layout und die wichtigsten Werkstoffdaten werden komplett auf EINEM Bildschirm präsentiert!

Die Nachteile seien nicht verschwiegen: sowohl die Eingabe, als auch die Grafikausgabe sind sehr in die Jahre gekom-

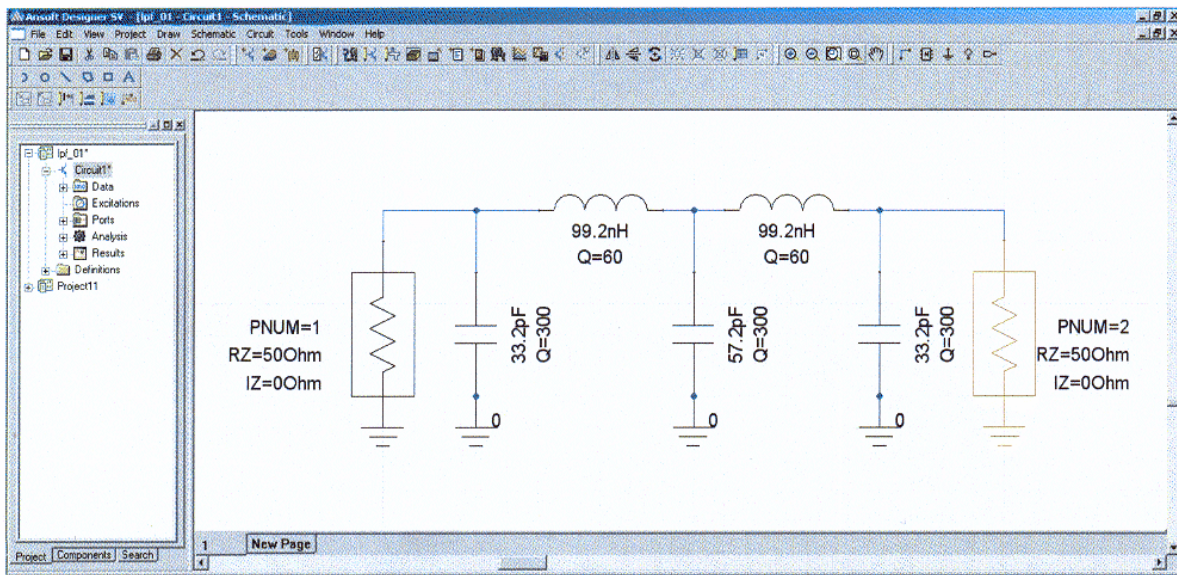


Bild 5: Der Ansoft-Designer erfordert schon mehr Bedieneraufwand, bis man das zu sehen bekommt!

men (besonders aber das Ausdrucken). Man behilft sich da über eine zusätzlich installierte und aufgerufene PCX-Ausgabe des Bildschirm-Screenshots. Es fehlen wichtige Modelle für Störstellen z.B. bei Mikrostreifenleitung usw - und es wird sicher nicht mehr auf eine Windows-Bedieneroberfläche umgestellt. Trotzdem: für schnelles Testen einer Idee von der Geschwindigkeit und der Bedienung her unschlagbar - bei gleicher Genauigkeit wie große, teure Maschinen! Und die Beschränkungen muss man einfach hinnehmen.

4.3. Der Ansoft Designer SV

Das ist natürlich modernster Stand der Simulationstechnik: ein Windows-Programm vom Feinsten und mit allen nur denkbaren Möglichkeiten ausgestattet. Hunderte von Bauteil- und Störstel-

lenmodellen decken alle denkbaren Anwendungsfälle bis hin zum Hohlleiter ab, ein integrierter „Filtercalculator“ erspart die Suche nach einem geeigneten externen Filter-Berechnungsprogramm. Die gesamte lineare Simulation ist ohne Einschränkungen freigegeben - wo liegt das Problem, wenn ANSOFT eine solch teure Software für Studienzwecke (SV bedeutet: Studentenversion...) kostenlos freigibt und zum Download [4] bereitstellt?

Nun, da stecken natürlich sehr raffinierte Marketing-Überlegungen dahinter: wenn nämlich Studenten ein solches Programm kennen und lieben lernen, werden sie mit großer Sicherheit am späteren Arbeitsplatz dasselbe verwenden wollen. Also setzt man die Begrenzungen exakt da an, wo es einer Firma weh tut, die heimlich mit dieser Version arbeiten möchte: man sperrt die direkte Übernahme eines Entwurfs in einen Platinent-

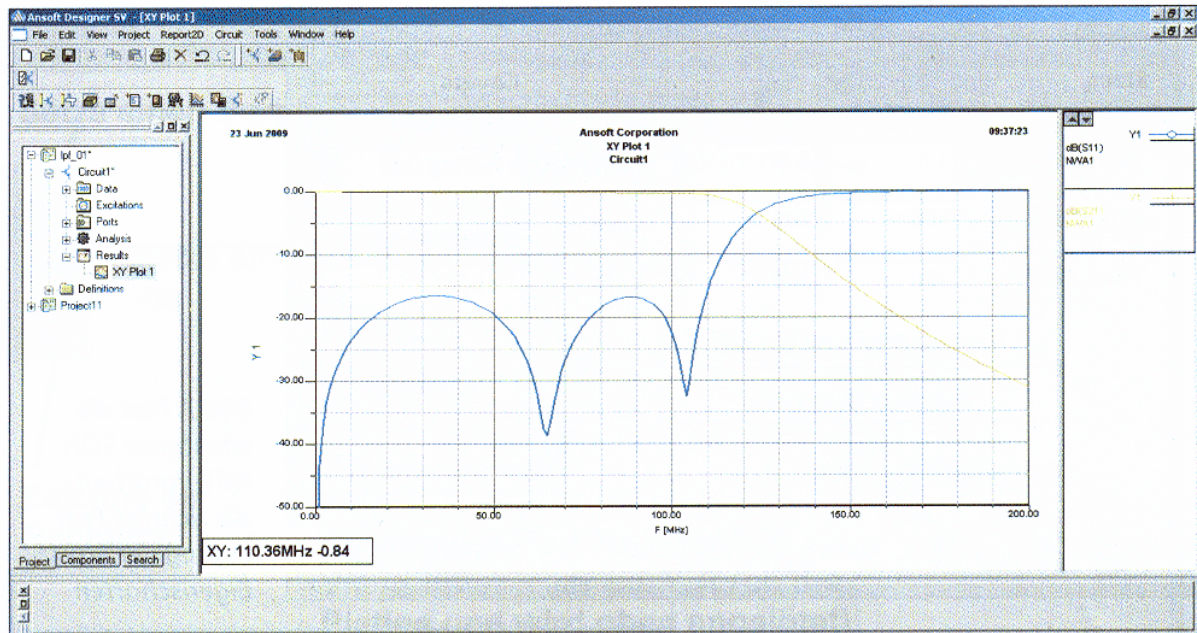


Bild 6: Der Anwender legt selbst nach der Simulation fest, was gezeigt werden soll - und in welcher Form

wurf, man sperrt die (sehr gute, selbst ausgetüftelte und patentierte) nichtlineare Simulation, man reduziert sehr stark die Möglichkeiten der automatischen Optimierung usw.

Für Amateurfunk-Anwendungen bleibt das trotzdem ideal und perfekt, aber gegenüber PUFF kommt doch kurze Wehmut auf: ein solch aufwendiges Programm mit so vielen fantastischen Möglichkeiten erfordert einen deutlich größeren Einarbeitungs- und Bedieneraufwand. So ist die Eingabe samt Simulation von der Ergebnisausgabe völlig getrennt, da man dem Programm hinterher erst sagen muss, was man gerne sehen möchte. Dafür gibt es nichts, was man sich nicht zeigen lassen kann: seien es Stabilitätswerte, S-Parameter, Rauschzahlen usw...

Hinweis: Auf der Homepage des Autors [2] kann dazu ein umfangreiches Tutorial (in Deutsch oder Englisch) kostenlos heruntergeladen werden. Es enthält viele Einsteigerprojekte mit aktiven oder passiven Schaltungen.

Der erste Teil der Arbeit ist in **Bild 5** zu sehen, nämlich die Eingabe der Schaltung für die Simulation.

Und ein Bildschirm gemäß **Bild 6** erscheint, wenn man nach der Simulation unter „Results“ die Ausgabe von S11 und S22 in kartesischen Koordinaten verlangt.

5. EM-Simulationen für den Funkamateur

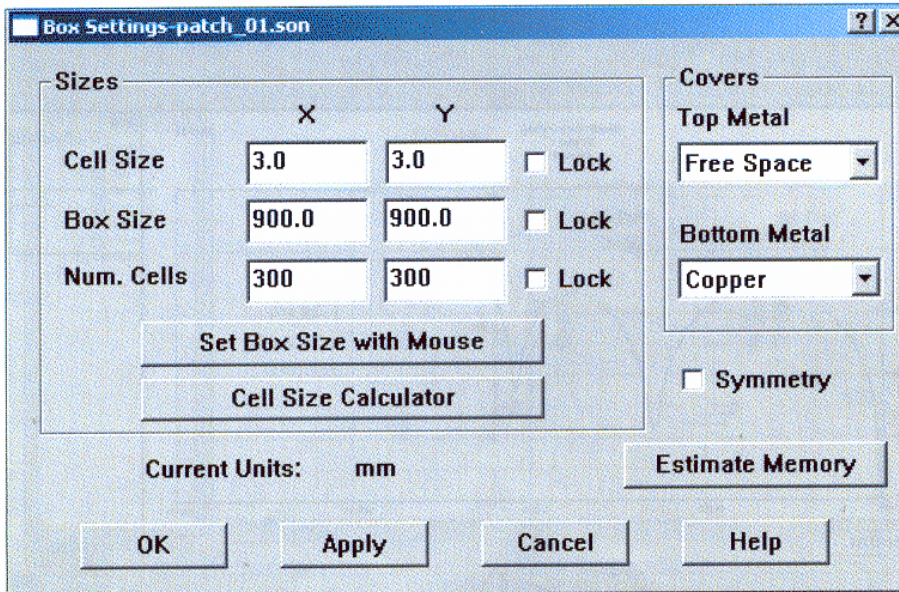


Bild 7: Fast die wichtigste SONNET-Vorarbeit: die richtige Festlegung der Box-Eigenschaften

5.1. Simulation von flächigen Strukturen mit Sonnet Lite

Auch hier gilt, wie im vorigen ANSOFT-Kapitel die Bemerkung über dieselbe gute Marketing-Strategie der Firma SONNET. Zwar wird bei diesem Programm eine echte Begrenzung über den maximal belegbaren Speicherplatz vorgenommen, aber im Endeffekt kann man gut damit leben und muss höchstens eine etwas höhere Ungenauigkeit beim Simulationsergebnis akzeptieren.

Dabei gilt:

- a.) wer lediglich das Programm (kostenlos) herunterlädt und installiert, bekommt 1 Megabyte Speicherplatz genehmigt;
- b.) wer sich anschließend online registrieren lässt und eine Lizenz beantragt, kann plötzlich mit 16 Megabyte arbeiten.

Da ständig Verbesserungen zum Einsparen von Speicherplatz in den Simulator einfließen, wird die Situation für die „Lite-Version“ immer günstiger...

Natürlich ist die Einarbeitung in diese Materie deutlich schwieriger, als der Umgang mit PSPICE oder PUFF. Das beginnt schon mit der erwähnten Aufteilung der Flächen in einzelne kleine Zellen und die Einbindung der verschiedenen verwendeten Materialien. Die Signaleinspeisung muss korrekt vorgenommen und immer wieder der bereits verbrauchte Speicherplatz kontrolliert werden.

Hinweis: Auf der Homepage des Autors [2] gibt es eine Rubrik „Meine Veröffentlichungen in der Zeitschrift UKW-Berichte“. Darin findet man den Artikel „Das interessante Programm - Heute: SonnetLite 9.51“.

Er enthält eine genaue Gebrauchsanleitung zur Bedienung des Programms und

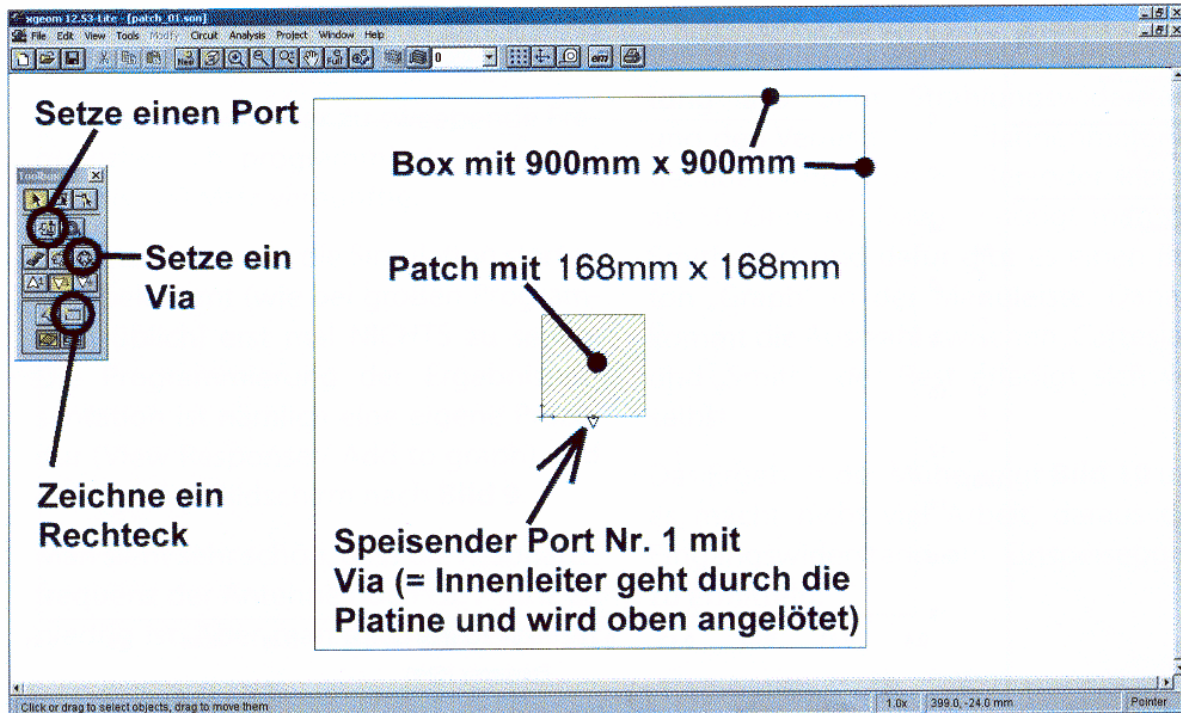


Bild 8: Wenn das zu sehen ist, hat man es geschafft: die Antennenstruktur samt Speiseport und „Via“ sind in der Box untergebracht

zwei unterschiedliche Patchantennen-Simulationen als Anwendungsbeispiele. Diese Anleitung kann auch noch für die modernste Programmversion verwendet werden.

Zur Einarbeitung und zur Hilfe bei Problemen: es gibt sehr viele Beispiele, Application Notes und Tipps auf der Homepage [5] bzw. im Internet und die FAQ-Abteilung ist wirklich erste Klasse.

Insgesamt ist alles prima, aber eine Sache tut weh: absichtlich ist die Simulation der Strahlungsdiagramme sowie die Antennengewinn-Berechnung gesperrt. Schade, schade...

Doch hier ein Beispiel in konzentrierter Form: Die Simulation einer quadrati-

schen Patchantenne aus FR4-Material für die Frequenz $f = 435$ MHz, durch die der Eingangswiderstand der Antenne an der Patchkante ermittelt werden soll. Dabei muss man Folgendes beachten:

SONNET ist nur ein Analyseprogramm und KEIN Entwurfsprogramm. Es testet nur die fertigen Ideen des Anwenders!

Also muss man den „Grundentwurf“ der Antenne z.B. mit dem kostenlosen DOS-Programm „patch16“ erstellen, das im Internet (nach der Eingabe „patch16.zip“ in die Suchmaschine) leicht zu finden ist und durch seine einfache und leichte Bedienung glänzt. Als Trägermaterial dient eine beidseitig kupferbeschichtete FR4-Platte mit den Abmessungen eines

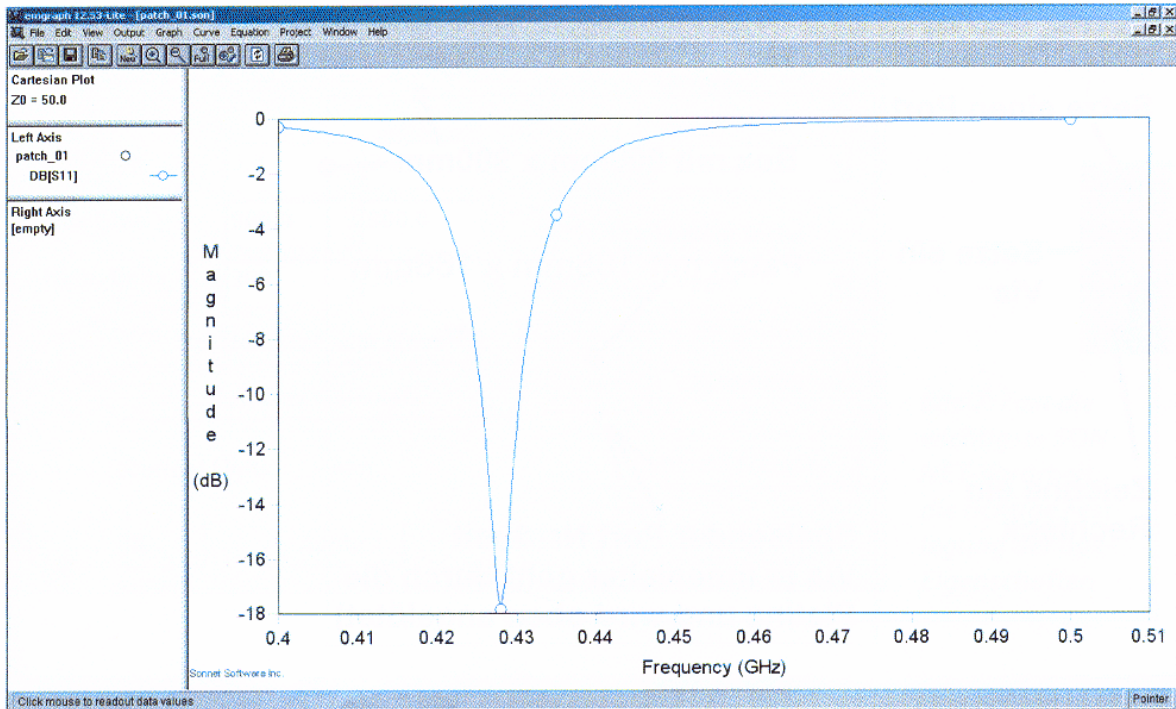


Bild 9: Das wollte man wissen: die Eingangsreflektion S_{11} informiert genau über die Resonanzfrequenz der Antenne

DIN A4-Blattes (210 mm x 297 mm) und einer Dicke von 1,52 mm (= 0,06"). Die Dielektrizitätskonstante liegt bei 4,35 und als Verlustfaktor ist $\tan \delta = 0,01$ zu verwenden. Das Patch-Programm schlägt nun 6,61" x 6,61" = 168 mm x 168 mm als Antennenabmessungen vor und hier soll mit SONNET die Brauchbarkeit des Vorschlags geprüft werden.

Nach der Eingabe des verwendeten Platinenwerkstoffs und der beteiligten Metalle muss man sich intensiv um die „Box“-Daten kümmern (**Bild 7**). SONNET verwendet als Grundlage der Simulation eine rechteckige „Kiste“, in die die zu analysierende Platine hineingelegt wird. Die Feldverteilung in einer solchen Box ist bekannt und berechenbar. Damit muss

der Anwender nun die Details programmieren.

Der erste Schritt ist die Eingabe der Box-Abmessungen (mit einer halben Wellenlänge Abstand der Wände in jeder Richtung von der zu untersuchenden Struktur). Weiter die Länge bzw. Breite sowie die Anzahl der Zellen, in die man seine Struktur zerlegen muss. Mehr als 5 % der Wellenlänge dürfen es bei der Zellenlänge und der Zellenbreite nicht sein, sonst leidet deutlich die Genauigkeit. Viel weniger als 1 % sollen es aus demselben Grund auch wieder nicht sein und so liegt das Optimum bei 1 bis 2 %.

Das Bildschirmfoto (**Bild 8**) zeigt den SONNET-Bildschirm nach Fertigstellung der Eingabearbeit; er sollte sofort unter



einem passenden Namen gespeichert werden.

Als nächstes wird der zu sweepende Frequenzbereich programmiert - hier sind 400 bis 500 MHz vernünftig.

Erst jetzt kann man die Simulation starten und bekommt (wie bei großen Programmen üblich) erst mal NICHTS zu sehen. Die Programmierung der Ergebnispräsentation ist nämlich eine eigene Prozedur (View Response / Add to graph) und erzeugt den Bildschirm nach **Bild 9**.

Man sieht sehr schön, dass die Resonanzfrequenz der Antenne noch ca. 7 MHz zu niedrig ist. Aber man erkennt bei dieser

Darstellung nicht, ob der Eingangswiderstand (der bei Resonanz als Parallelschaltung aus dem Strahlungswiderstand und den Verlusten des Platinenmaterials gebildet wird) nun größer oder kleiner als 50 Ohm ist. Dazu benötigt man das Smithchart und dafür gibt es einen Button „Graph“ in der Menüleiste. Danach kommt die Auswahl zwischen „Cartesian“ und „Smith“, der Rest erledigt sich von selbst.

Das Ergebnis der Mühe zeigt **Bild 10** und es macht nicht viel Arbeit, daraus den Eingangswiderstand am Einspeisepunkt zu ermitteln:

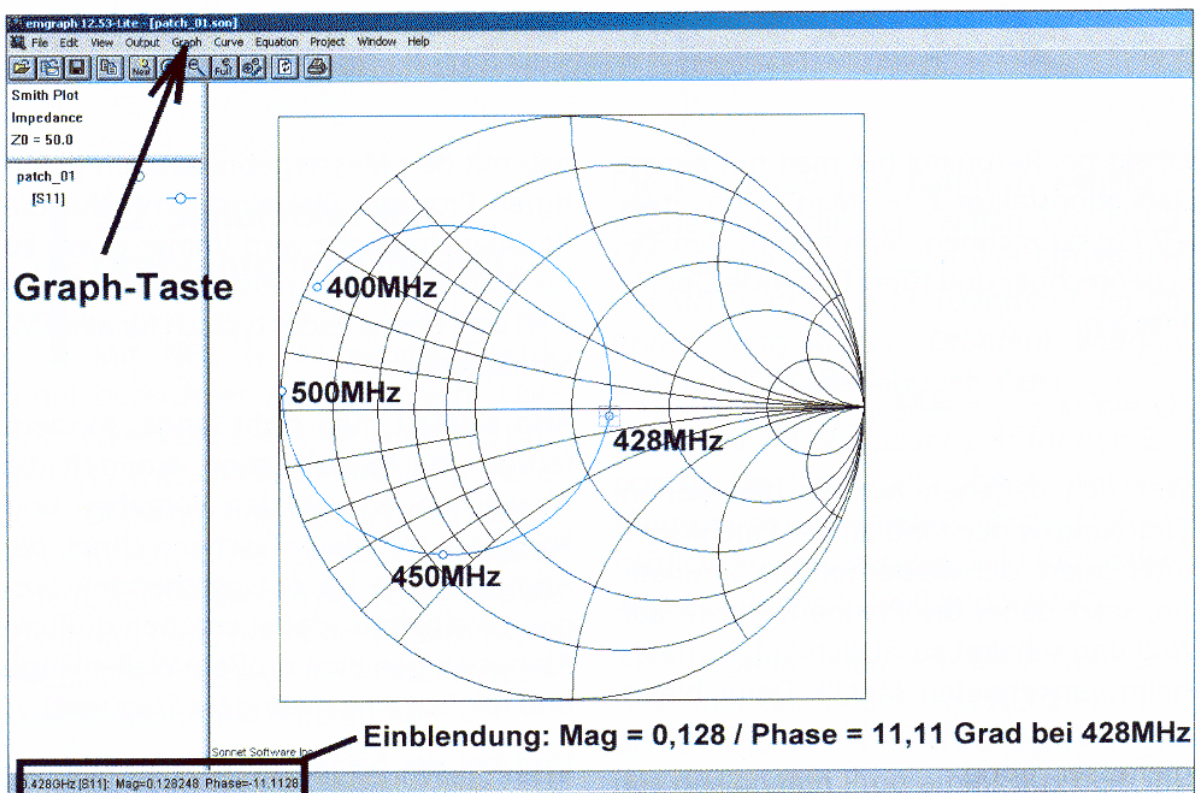
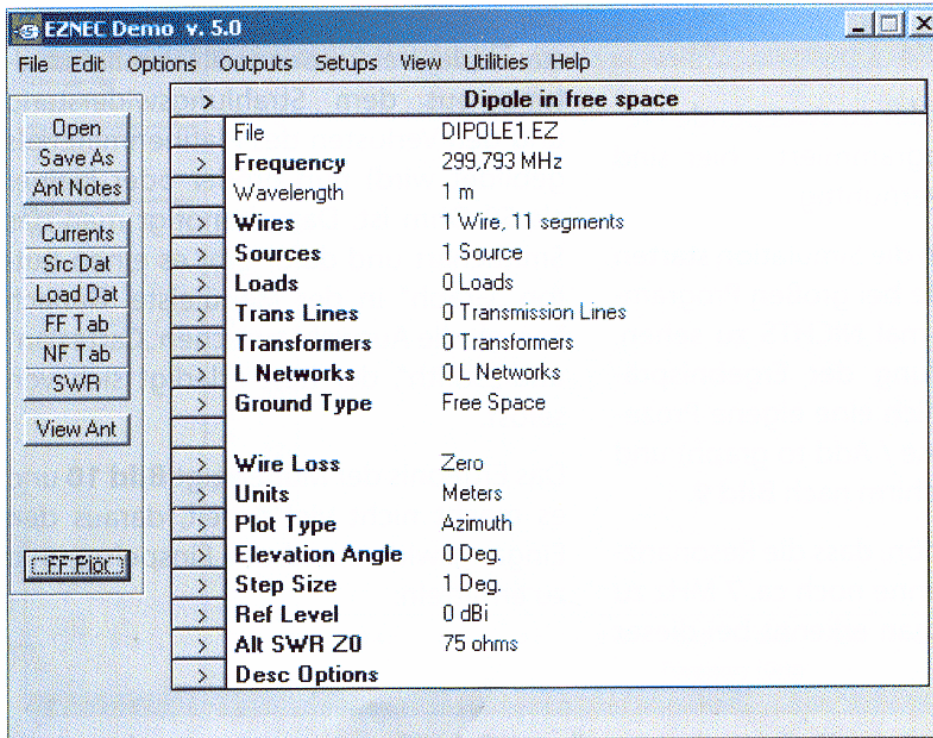


Bild 10: Unverzichtbar zur Berechnung des genauen Eingangswiderstandeswertes: das Smith-Diagramm (siehe Text)



*Bild 11:
So geht es auch:
die EZNEC-
Eingabemaske
mit der auto-
matisch gela-
denen letzten
Anwendung
wird einfach
„nachbear-
beitet“*

Direkt bei Resonanz hat man mit einem Reflektionsfaktor r (= „Mag“) von etwa $+0,128$ zu rechnen. Man nimmt den Taschenrechner und führt folgende Operation aus:

$$R_{IN} = 50\Omega \cdot \frac{1+r}{1-r} = 50\Omega \cdot \frac{1+0,128}{1-0,128} = 64,67\Omega$$

Wer sich dagegen nur für den reinen Strahlungswiderstand an der Patchkante interessiert, der wiederholt die Simulation, setzt dabei die Platinenverluste auf Null und schaltet zusätzlich auf „lossless“ beim verwendeten Metall. Dann bleibt nur noch die Abstrahlung als „Energiefresser“ übrig.

Das ist eine schöne Sache und die Erfahrung zeigt, dass die Simulation recht ge-

nau mit den Messergebnissen am gefertigten Prototyp übereinstimmt. Nur die Resonanzfrequenz wird immer etwas zu hoch ausgegeben (meist zwischen 1 und 3 %) und das ist leider typisch für alle EM-Simulatoren.

Also sinniert man nicht lange, sondern fertigt eine Musterplatine, ermittelt die Abweichungen von den Vorgaben und korrigiert mit dem Taschenrechner die Abmessungen im entsprechenden Frequenzverhältnis (dabei ergeben größere Abmessungen eine größere Wellenlänge und folglich eine niedrigere Frequenz).

Und bei der Sache mit den Strahlungsdiagrammen der Antennen kann man nur hoffen und abwarten...

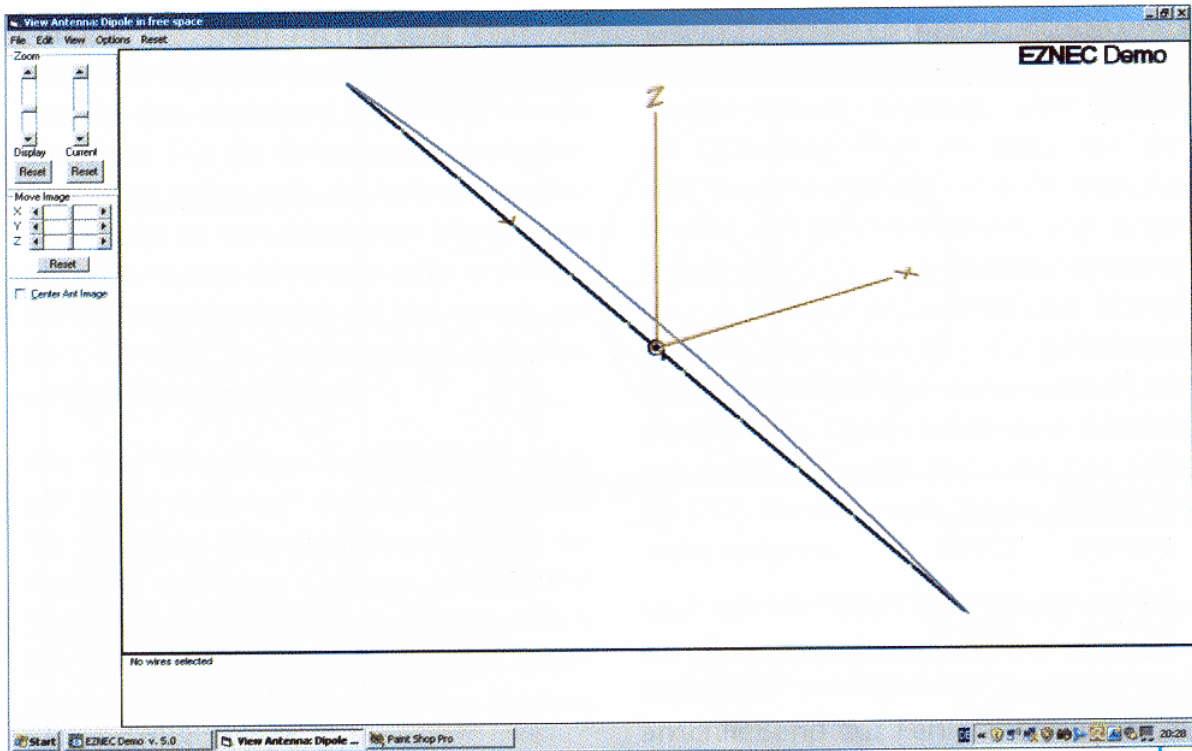


Bild 12: So sieht die zu simulierende Antenne aus...

5.2. Simulation von Drahtantennen mit NEC

Mit NEC (= Numerical Electric Code) bezeichnet man eine vom Lawrence Livermore Labor im Jahre 1981 für die Navy entwickelte Simulationsmethode für Drahtantennen. Dabei wird die Antenne in sehr kurze Stücke = „segments“ zerlegt, auf denen sich der Strom und die Spannung (fast) nur linear ändern (siehe SPICE). Damit lassen sich erstaunlich genaue Simulationen durchführen.

Der Standard - für den es sehr viele, auch kostenlose Anwendungssoftware gibt - ist NEC2.

Die Entwicklung ist natürlich weitergegangen und die Schwächen von NEC2 (z.B. falsche Berechnung von Strukturen mit sich sehr nahe kreuzenden oder im Boden vergrabenen Drähten) wurden erst mit NEC4 ausgebügelt. Aber:

NEC4 war bis vor kurzer Zeit für den Export gesperrt und galt früher als geheim. Heute ist es auch außerhalb der USA erhältlich, kostet aber viel Geld (normalerweise zwischen \$ 2000 und \$ 4000).

Also begnügt man sich als Privatmann (= kleiner Geldbeutel) mit NEC2 und hat dafür die Qual der Wahl bei der Auswahl des geeigneten Programms aus dem Internet. Da ist es sinnvoll, etwas zu wählen, das:

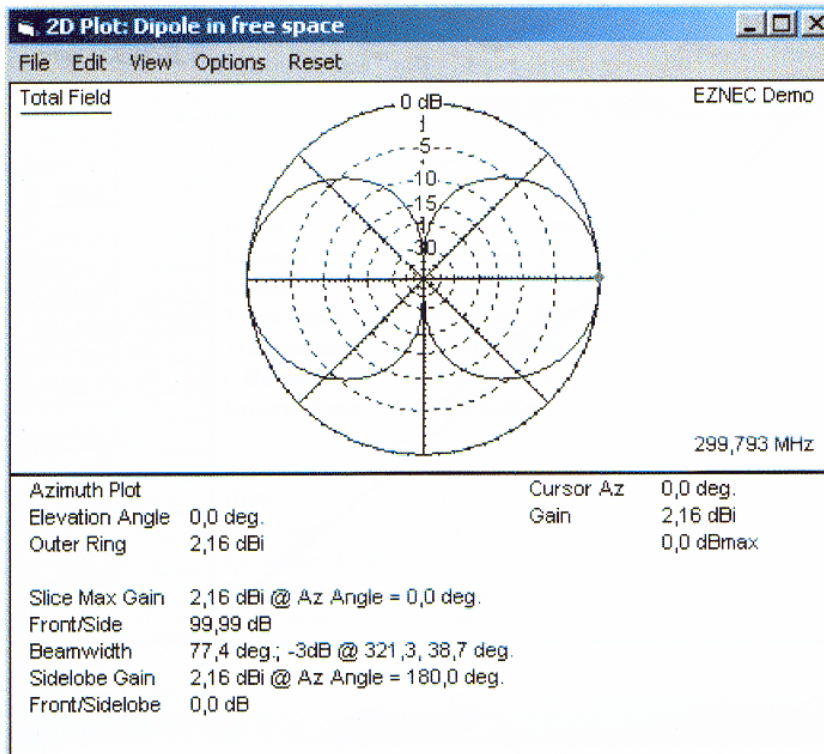


Bild 13:
...und das ist das zugehörige simulierte horizontale Fernfeld-Diagramm

- a) schon weit verbreitet und beliebt ist,
- b) eine ansprechende und leicht zu bedienende Eingabe-Oberfläche besitzt,
- c) eine leicht verständliche Ergebnisausgabe aufweist,
- d) regelmäßig und gut gepflegt wird,
- e) eine Nah- und Fernfeld-Simulation durchführen kann,
- f) von einer Fangemeinde mit vielen Beispielen und Vorlagen und Hilfsbereitschaft unterstützt wird,
- g) ein gutes Handbuch und eine gute Einsteigeranleitung vorweist und...
- h) den Privatmann nicht finanziell ruiniert.

Aus diesen Gründen greifen viele Funk-

amateure und Antennenentwickler seit Jahren zu EZNEC (Autor: Roy Lewallen, [8]); wobei der aktuelle Preis von \$ 89 beim direkten Download und Bezahlung per Kreditkarte ist mehr als fair.

Einen Blick auf die Bedieneroberfläche der neuesten Version 5.0 dieses Programms zeigt **Bild 11**, wenn man aus dem mitgelieferten Beispielvorrat einen einfachen Lambda-Halbe-Dipol für 299,793 MHz auswählt. Hier sieht man nur ein Reihe von Buttons und einen kleinen Bildschirm mit Eingabezeilen und das hat seinen Grund:

EZNEC hat nämlich eine etwas andere Philosophie und zeigt kein „leeres Blatt“ beim Start, sondern immer das letzte geladene Beispiel. Also holt man sich



einfach aus dem Archiv eine beinahe passende Antenne und ändert der Reihe nach in den einzelnen Zeilen des Menüs die Werte. Für die Untersuchung eigener Antennen sollte man deshalb einen großen Vorrat an verschiedenen simulierten Antennentypen sammeln oder erstellen. Diese Vorgehensweise ist mal etwas anders, nämlich „Re-Engineering“, aber deswegen nicht schlechter!

Wie der Bildschirm nach einem Klick auf „View Antenna“ aussieht, zeigt **Bild 12**. In einem Koordinatensystem ist die Antenne mit ihren Drähten (wires) und Segmenten angeordnet und sie lässt sich mit den links auf dem Bildschirm angeordneten Bedienelementen im System verschieben.

Ein Klick auf „FF Plot“ (.....„FF“ steht für „Farfield“) erzeugt das Strahlungsdiagramm (**Bild 13**) und man kann mit dem Cursor die dargestellte Kurve abfahren. Natürlich lässt sich auch ein anderer Elevationswinkel wählen und unter „Setups“ im Hauptmenü findet man sogar die Option „Near Field“.

Tipp: Wer Strahlungsdiagramme lieber in räumlicher (= 3D-) Darstellung und in poppigen Farben sehen möchte, der hole sich mal das Programm „4nec2“ aus dem Internet!

6. Abschließende Hinweise zur Antennensimulation

Es gibt natürlich - wieder im Internet - Antennenberechnungsprogramme, die direkt mit passend zugeschnittenen

Näherungsformeln arbeiten. Das Programm „patch16“ wurde im SONNET-Kapitel bereits erwähnt, aber es gibt ein DOS-Programm im Web, mit dem eine Vielzahl von Antennenformen (von Draht- und Patchantennen, von Antennen-Arrays über Log-Periodic-Antennen bis hin zu Parabolspiegeln und Hornantennen) durch Einträge der geforderten Daten am Bildschirm dimensioniert werden können. Dieser kostenlose Alleskönner heißt „pcaad21.zip“ und hat selbst als DOS-Version noch heute seinen Reiz nicht verloren.

Und nun viel Spaß und Erfolg mit eigenen Projekten, bei denen die Simulation nicht nur zum Entwurf, sondern auch anschließend zur Fehleranalyse und zur Fertigungsentwicklung eingesetzt wird!

7. Literatur und Links

Hier finden Sie weiterführende Links und Literaturhinweise zum Artikel:

- [1] Linear Technologies, Homepage mit „LTSpice IV“:
www.linear.com
- [2] Gunthard Kraus, DG8GB, Homepage unter Elektronischule, Tettang:
www.elektronischule.de/~krausg
- [3] Leserservice der „UKW-Berichte“
www.ukw-berichte.de
- [4] Homepage von ANSOFT:
www.ansoft.com
- [5] www.sonnetsoftware.com
- [8] www.eznec.com