



Gunthard Kraus, DG 8 GB

Das interessante Programm:

Heute: qucsStudio

Moderne Schaltungsentwicklung ist heute ohne ein gutes Simulationsprogramm undenkbar. Die speziellen Probleme der HF- und Mikrowellentechnik erfordern jedoch eine „Microwave CAD-Software“. Mit „QucsStudio“ gibt es jetzt eine sehr leistungsfähige und kostenlose Software in deutscher Sprache.

Die weltweit verbreitete Software SPICE (die ja in Form von LTspice als völlig unbegrenzte, sehr leistungsfähige und dazu kostenlose Vollversion im Gebrauch ist) stößt leider sehr schnell an die Grenze der Möglichkeiten, wenn es um die spezifischen Probleme der HF- und Mikrowellentechnik geht. In solchen Fällen benötigt man eine echte „Microwave CAD Software“, die Modelle aller denkbaren und vorkommenden Bauteile und Baugruppen zur Verfügung stellt und nach anderen Prinzipien (z.B. der S-Parameter-Simulation) arbeitet. Speziell hier ist der Markt für kostenlose oder preisgünstige

Versionen sehr dünn. Man konnte sich bisher noch auf die etwas betagten Programme wie den „ANSOFT Designer SV“ oder „PUFF“ stützen.

Nun ist eine solche Software verfügbar - kostenlos, wahlweise in deutscher oder englischer Sprache, sehr leistungsfähig, auf den HF-Bereich einschließlich aller Übertragungssysteme abgestimmt und durch ein stetig wachsendes deutsches Tutorial (augenblicklicher Stand: knapp 200 Seiten) unterstützt. Da lohnt sich der Ein- oder Umstieg!

1. Ein kurzer Steckbrief

Charakterisierung:

Simulator in der Time- und in der Frequency Domain mit folgenden Optionen:

- DC-Simulation
- AC-Simulation
- Transienten-Simulation
- S-Parameter-Simulation

- Harmonic Balance
- Parameter-Simulation
- Digitale Simulation
- Systemsimulation
- Optimierung
- Monte Carlo-Simulationen

Dazu stehen noch zur Verfügung:

- ein Spulen-Designer
- ein Filtercalculator
- ein Leitungscalculator
- eine Dämpfungsglied-Synthese
- ein Tool zur Dimensionierung von Anpass-Schaltungen
- eine GPIB-Steuerung für einige hp-Messgeräte
- ein Gerber-Viewer, um Platinen-Layouts sichtbar zu machen

Das ist Spitze:

Viele Simulationsmöglichkeiten und Tools.

Nichtlineare Simulation sowohl über Transienten-Simulation wie auch über Harmonic Balance möglich

Keinerlei Begrenzungen.

Alle Informationen, Bezeichnungen und Menü-Führungen in Deutsch

Kostenlos

Wird regelmäßig gepflegt

Keine Registrierung unter Windows. Dadurch kann das Programm sogar von einem USB-Stick über die „qucs.exe“-Datei gestartet und betrieben werden. Auch das Löschen hinterlässt keine Spuren im System und Verschieben an einen beliebigen anderen Ort läuft ohne jedes Problem.

Alle wichtigen HF- und Mikrowellen-Bauteile sowie die Baugruppen der Übertragungstechnik sind als Modelle vorhanden.

Alle üblichen Diagrammarten (von Kartesisch über Smith bis Eye Pattern) stehen zur Verfügung.

Sehr gute Fast Fourier (FFT)-Rechenmaschine zur Spektrumsbestimmung (kann es jederzeit mit der ebenso hervorragenden Maschine in LTspice aufnehmen).

Große Liebe zu VERILOG

Das umfangreiche, kostenlose und stetig wachsende deutsche Tutorial [1] mit vielen HF-Projekten ermöglicht einen leichten Einstieg und animiert zur Weiterbildung.

Daran muss man sich erst gewöhnen:

Manchmal recht eigenwillige bis irritierende Organisation der Bedienung und Menüführung (...darüber stolpert man nur, wenn man viel mit anderen CAD-Programmen arbeitet - normalerweise ähneln sich die Benutzerführungen).

Für jede simulierte Spannung oder Größe muss erst ein eigenes Ergebnis-Diagramm programmiert werden.

Importierte SPICE-Modelle aus dem Internet werden grundsätzlich erst über einen in „qucsStudio“ eingebauten Konverter geschickt und damit für die Simulation verwendbar gemacht, dies funktioniert jedoch einwandfrei.

Betriebssystem:

Jede moderne Windows-Version; getestet mit Windows 7.

Genauigkeit, Zuverlässigkeit und Betriebssicherheit:

Liefert präzise Ergebnisse, die mit anderen CAD-Programmen identisch sind. Läuft sehr zuverlässig, solange man



nicht mal aus Versehen mehrere Tasten gleichzeitig drückt - das kann bisweilen schief gehen. Aber bei einem folgenden eventuellen Crash empfiehlt sich eine Neu-Installation (= eigene Erfahrung), die plötzliche neue und unerwünschte Reaktionen des Programms sicher beiseitigt. Die Installation des Programms geht schnell, da keine Registrierungen unter Windows vorgenommen werden (= Auspacken der gezippten Software reicht schon). Und bitte regelmäßig auf die neueste Version updaten.

Probleme und Verbesserungswünsche:

Die Sache mit der Online-Hilfe klappt nicht auf allen Rechnern. In manchen Fällen wird nur das Inhaltsverzeichnis aufgerufen, aber die Verknüpfungen mit den Themen funktionieren nicht. Das könnte ein Problem der Datei-Verwaltung bzw. der richtigen Einstellungen in Windows sein.

Für die Zukunft:

Es hat sich leider noch niemand an die Programmierung eines kostenlosen EM-Simulators gewagt...

2. QucsStudio - was steckt dahinter?

Zunächst gleich etwas Information zur Verwirrung:

Es gibt „Qucs“ und „QucsStudio“. Deshalb hier zuerst „Qucs“:

„Qucs“ steht für „Quite Universal Circuit Simulator“ und das ist natürlich ein hoher Anspruch. Er entstand im Universitätsbereich, ist kostenlos, ohne jede Beschränkung und leidet deshalb nicht an irgendwelchen Vorgaben, die den Verkauf för-

dern sollen. Die „qucs-Software“ arbeitet vollständig mit deutscher Bedienung, aber etliche Fans haben sich bereits die Mühe gemacht, andere Sprachversionen bereitzustellen.

Es kann in der Time Domain und in der Frequency Domain simulieren. Dass der Mikrowellen-Spezialist sein geliebtes Smith-Diagramm verwenden kann, das ist ebenso selbstverständlich wie die Rauschsimulation oder die Stabilitätskontrolle anhand der S-Parameter oder ein Parameter-Sweep. Was aber den HF-Fachmann sehr zur Einarbeitung reizt, ist speziell die enthaltene „Harmonic Balance Simulation“ zur Ermittlung von Oberwellen-Spektren. Sie steht in keiner anderen Software kostenlos zur Verfügung und das macht einem geradezu den Mund wässrig...

Wenn man sich mit Simulationsprogrammen etwas auskennt, fällt einem gleich die hier herrschende Philosophie auf:

Es geht nicht darum, dem Schaltungsentwickler im Labor ein Werkzeug an die Hand zu geben, mit dem sich gegenüber einem Konkurrenzprodukt wieder einige Sekunden an Arbeitszeit einsparen lassen. Oder dass von der Umsetzung der Idee in eine Schaltung mit nachfolgendem Leiterplattendesign samt einer Prüfung der EMV-Abstrahlung und des Übersprechens alles sofort möglich ist - natürlich mit nur einem Mausklick. Dieses Programm will dagegen möglichst viele Werkzeuge bereitstellen, mit denen man einem Problem gründlich auf den Zahn fühlen kann (und folglich kann damit nicht nur die Krone des Zahns, sondern auch seine Wurzel sehr genau untersucht werden).

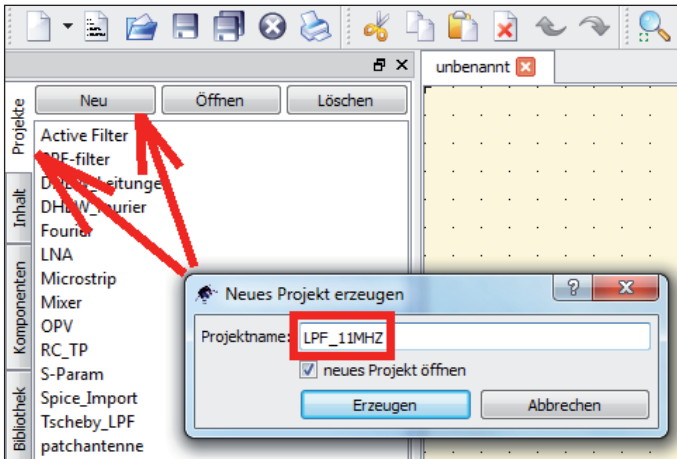


Bild 1:
Damit geht es los:
Zuerst wird das
neue Projekt ange-
legt und gespeichert

Das bedeutet, dass sehr viele Funktionen für die Simulation, aber auch unterschiedlichste Darstellungsmöglichkeiten bei den Ergebnissen bereit stehen. Allerdings will und kann „qucs“ nie verleugnen, dass sein geliebtes Einsatzgebiet die HF- / Nachrichten- und Mikrowellentechnik ist. Muss man wissen...

Daher lässt es sich nicht vermeiden, dass manche Dinge deshalb etwas „leicht gewöhnungsbedürftig“ ausfallen. Auf Grund der Leistungsfähigkeit und der vielen angebotenen Möglichkeiten kann man diese jedoch problemlos ertragen.

Allerdings passierte irgendwann dasselbe wie in Erbgemeinschaften: Man konnte sich nicht mehr einigen und deshalb hat der Chef-Entwickler (Michael Margraf, DD6UM) irgendwann beschlossen, seinen eigenen Weg zu gehen. Er nahm zwar das Konzept mit, schrieb aber die Software neu, entwickelte weiter und nannte sie „QucsStudio“. Deshalb gibt es natürlich bisweilen leichte Unverträg-

lichkeit zwischen beiden Programmen. Man hat aber den Eindruck, dass Weiterentwicklungen nur noch bei „QucsStudio“ erfolgen und deshalb soll es hier nur um diese Programmversion gehen. Vieles, was jetzt folgt, ist auch bei „Qucs“ verwendbar...aber, wie schon gesagt... manchmal kann es dort klemmen.

3. Beschaffung und Installation von „QucsStudio“

Es reicht die Eingabe „qucsStudio“ in die Maske einer Suchmaschine, um auf der Seite „QucsStudio Homepage - DD6UM“ zu landen. Man lädt die aktuellste Version herunter die gepackt ist; bisweilen gibt es beim Entpacken ein Problem: Diese exe-Datei ist mit „7z“ gepackt, was bedeutet, dass man sich erst ein passendes Programm im Internet suchen muss.

Der Rest ist reine Installationsroutine und



läuft ohne Probleme. Für die anschließende Arbeit mit „qucsStudio“ findet sich anschließend auf der Festplatte unter „Benutzer / Administrator“ automatisch ein „qucs“-Ordner, der die bearbeiteten Projekte aufnimmt.

4. Praxis-Beispiel:

Entwurf und Simulation eines Tschebyschef-Tiefpasses mit einer Grenzfrequenz von 11 MHz

Wie sich so ein Programm „anfühlt“, zeigt am besten ein Beispiel. Deshalb folgt hier ein Projekt, das in ähnlicher Form im Tutorial enthalten ist. Es zeigt nicht nur den Umgang mit dem Filtercalculator, sondern auch die Schaltungserstellung, die AC-Simulation und die Simulation im Zeitbereich. Wer Freude daran hat, kann gleich starten und mitmachen...

4.1. Pflichtenheft

Für ein Kommunikationssystem mit $Z = 50 \Omega$ soll ein Tiefpass mit $N = 5$ entworfen werden, der alle Signale oberhalb der in UKW-Empfängern üblichen Zwischenfrequenz von $f = 10,7 \text{ MHz}$ unterdrückt. So etwas beginnt grundsätzlich immer mit dem Pflichtenheft:

Systemwiderstand:	$Z = 50 \Omega$
Filtertyp:	Tschebyschef
Filtergrad (Order):	$N = 5$
Welligkeit (Ripple) im Durchlassbereich:	0,3 dB
3 dB-Grenzfrequenz:	11 MHz
Gemessene Spulengüte:	$Q = 75$ bei 11 MHz

Zum Entwurf wird der in „qucsStudio“ enthaltene Filtercalculator eingesetzt.

4.2. Anlegen eines Projektes und Einsatz des Filtercalculators

Nach dem Programmstart wirft man einen Blick auf die linke obere Seite des Bildschirms (**Bild 1**) und sorgt dafür, dass die Karteikarte „Projekte“ aktiv ist. Ein nachfolgender Klick auf „Neu“ fordert auf, dem neuen Projekt einen Namen zu geben; hier z.B. „LPF_11MHZ“. Wie **Bild 2** zeigt, sollte man auch gleich die Schaltung, die dann auftaucht, (in bekannter Weise über „Datei“ und „Speichern als“) mit einer Bezeichnung versehen; gewählt wurde „LPF_11M_01“.

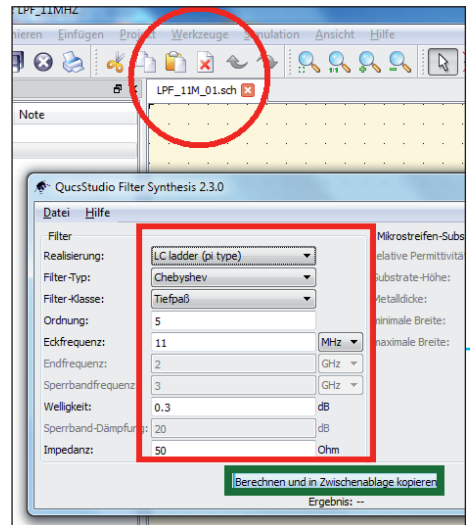


Bild 2: Nach dem Speichern der aktuellen Schaltung übernimmt der Filtercalculator die Entwurfsarbeit für den Tiefpass

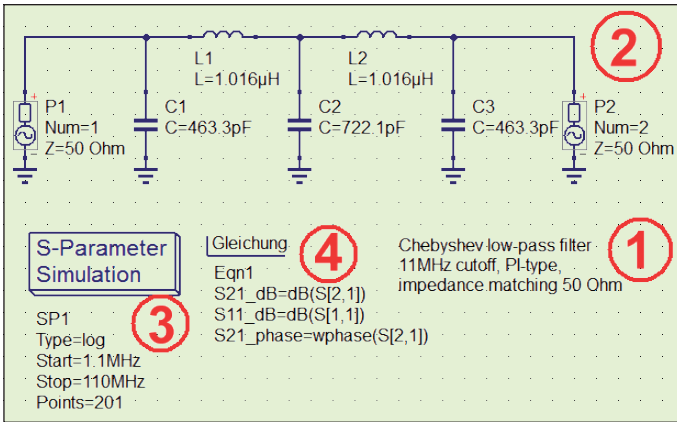


Bild 3:
Mehr kann man nicht verlangen (... und das bieten nur wenige andere Programme): Das Ergebnis ist die bereits komplett für die Simulation vorbereitete Filterschaltung

Doch im roten Kreis in Bild 2 versteckt sich außerdem das Menü „Werkzeuge“ und darin braucht man „Filtersynthese“, also den Filtercalculator. In der auftauchenden Maske gibt man der Reihe nach die Filterdaten aus dem Pflichtenheft ein:

Realisierung:	LC ladder (pi type)
Filtertyp:	Chebyshev
Filterklasse:	Tiefpass
Ordnung:	5
Eckfrequenz:	11 MHz
Welligkeit:	0,3 dB
Impedanz:	50 Ohm

Dann reicht ein Klick auf die grün eingerahmte Taste „Berechnen und in Zwischenablage kopieren“, um den Filtercalculator zu starten. Anschließend löscht man diesen und kehrt zur Schaltung „LPF_11M_01“ zurück und gibt den üblichen Kopierbefehl „<Control> + <v>“ ein. Das Ergebnis sieht man in **Bild 3** wo schon alles automatisch für die Simulation vorbereitet ist. Man findet in „1“ einen Kommentar und eine kurze Beschreibung in „2“ die fertige Schaltung samt bereits

angeschlossenen Ports für die S-Parameter-Simulation

in „3“ die Sweep-Vorgaben (= logarithmischer Sweep von 1,1 bis 110 MHz mit 201 Punkten) und

in „4“ die erforderliche Programmierung der S-Parameter-Berechnung sowie des Phasenverlaufs über eine Sammlung von Gleichungen.

Nach der Simulation (= Klick auf das Zahnradchen kurz vor dem Ende der Windows-Menüleiste) wird man wie üblich von einem leeren Ergebnis-Bildschirm empfangen und deshalb zieht man sich ein kartesisches Diagramm (= links oben im Diagramm-Vorrat) auf die Zeichenfläche. Dabei öffnet sich automatisch das „Eigenschafts-Menü“ und entsprechend **Bild 4** klickt man nur auf S21_dB in der linken Liste, um diesen Verlauf darzustellen. Allerdings sollte man sich selbst einen Gefallen tun und die Linienbreite noch auf „2“ umstellen, bevor man auf „Anwenden“ und OK klickt. In **Bild 5** ist das Ergebnis zu sehen (...und das am Cur-

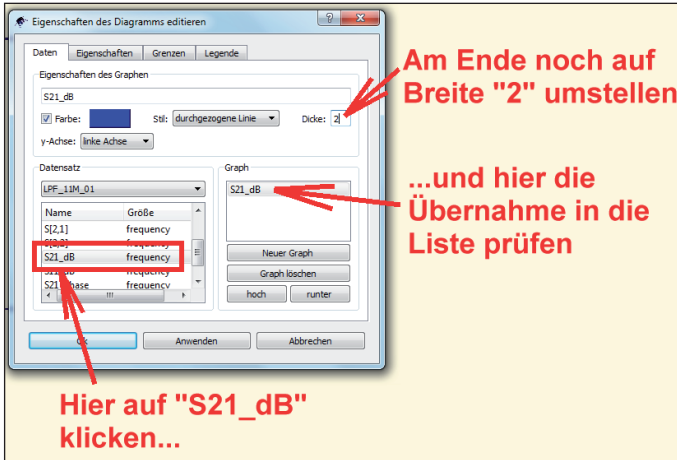


Bild 4:
Ohne Fleiß
kein Preis!
Das Ergebnis-
Diagramm muss
man sich selbst
zusammenstricken
(siehe Text)

sor hängende nächste neue Diagramm wird mit „Escape“ beseitigt).

Allerdings lassen sich die Achsenbezeichnungen noch verbessern. Also klickt man mit der rechten Maustaste auf das Diagramm, gefolgt vom Aufruf „Eigenschaften editieren“.

Nach einem Wechsel auf die Karteikarte „Eigenschaften“ trägt man unter „x-Achsen-Beschriftung“ das Wort „Frequenz“ ein.

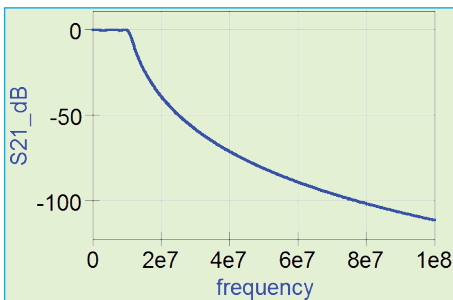


Bild 5: Bei diesem Ergebnis kommt die erste Freude auf - der S21-Verlauf stimmt genau

Weiter geht es auf der nächsten Karteikarte „Grenzen“. Bei der x-Achse wird bei „manuell“ ein Häkchen gesetzt und anschließend der Frequenzbereich von Null MHz bis 100 MHz mit einem Tick von 20 MHz in der Weise vorgegeben, wie es **Bild 6** zeigt.

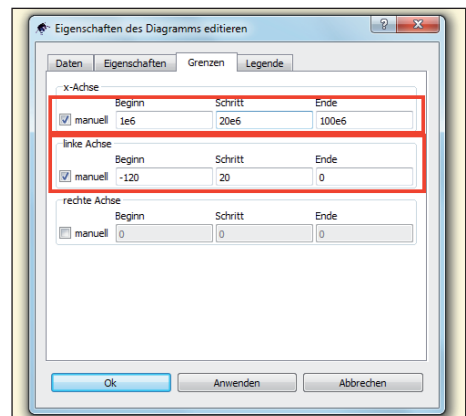


Bild 6: Allerdings ist die Darstellung noch verbesserungswürdig, deshalb wird die Teilung der senkrechten Achse verfeinert....

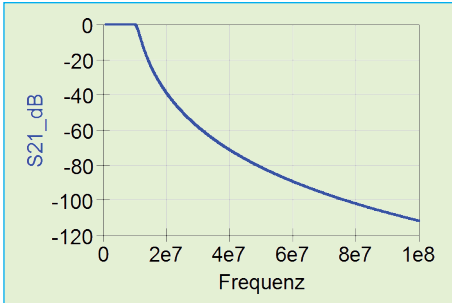


Bild 7: ...und so hat man an S21 mehr Freude

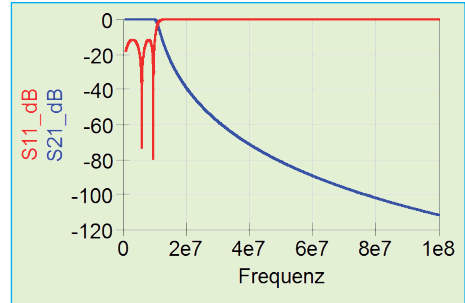


Bild 8: Richtig rund wird die Sache aber erst mit der gemeinsamen Darstellung von S21 und S11

Auf der Karteikarte „Grenzen“ setzt man bei „linke Achse“ im Feld „manuell“ ebenfalls ein Häkchen. Auch hier empfiehlt sich nochmals ein Blick auf Bild 6, wie die Eingabe „zeige den Bereich von -120 bis Null dB mit einem Tick von 20 dB“ vorzunehmen ist.

Und nun wird es interessant:

Ein Klick auf „Anwenden“ reicht, um alle Änderungen sofort im Diagramm sichtbar zu machen (**Bild 7**). Ein Klick auf OK schließt das Eigenschaften-Menü.

Gleich ein Tipp: Mit einem Rechtsklick auf Bauteile, Diagramm oder Quellen usw., gefolgt von „Eigenschaften editieren“, kann man immer ihr Eigenschaften-Menü öffnen und darin Korrekturen, Änderungen oder Erweiterungen vornehmen.

Der S21-Verlauf in Bild 7 bietet in dieser Form ja wirklich einen schönen Anblick und so kann man gleich das eben Erlernte anwenden, um auch noch S11 im gleichen Diagramm unterzubringen. Das geht so:

1. Eigenschaften-Menü öffnen

2. Prüfen, ob „S21_dB“ korrekt in der rechten Graphenliste aufgeführt ist. Dann in der linken Liste auf „S11-dB“ klicken. Dieser Wunsch sollte jetzt korrekt in der Liste dazu gefügt sein.

3. Die Linienbreite auf „2“ umstellen und prüfen, ob das Diagramm nun in roter Farbe dargestellt wird.

4. Auf „Anwenden“ klicken und das Ergebnis (**Bild 8**) bewundern.

5. Mit OK abschließen.

Auch hier wieder ein Tipp: Möchte man die Farbe einer Linie ändern, öffnet man das Eigenschaften-Menü des Diagramms und klickt in der rechten Graphenliste auf den gewünschten Verlauf. Dann entfernt man kurz das Häkchen bei „Farbe“ und setzt es anschließend sofort wieder. Das Farbfeld wird nun plötzlich schwarz sein, aber ein Mausklick auf dieses schwarze Feld öffnet die Farbpalette. Hier sucht man sich heraus, welche Farbe man lieber hätte, klickt auf die gewünschte Farbe und bestätigt mit OK, fertig! („aber natürlich erst nach „Anwenden“ und OK...“).

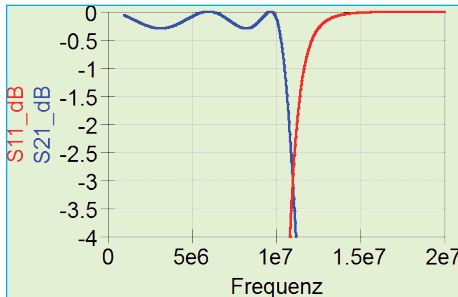


Bild 9: Auch die Darstellung der Tschebyschef-Wellen funktioniert ohne Probleme

4.2. Verfeinerung der Darstellung

Wer sich für die „Tschebyschef-Wellen im Durchlassbereich“ interessiert, der gehe folgendermaßen vor:

Erst ein rechter Mausklick auf das Diagramm, dann „Eigenschaften editieren“.

Jetzt auf die Karteikarte „Grenzen“ wechseln und

- bei der x-Achse einen Bereich von 0 MHz bis 20 MHz mit einem Tick von 5 MHz einstellen.
- Bei der linken Achse „zeige den Bereich von -4 dB bis Null dB mit einem Tick von 0.5 dB“ programmieren.

Wie das aussehen kann, zeigt **Bild 9**. Der Schnittpunkt von S11 und S21 bei „-3 dB“ ist gut zu sehen - hier findet man die vorgegebene Grenzfrequenz von 11 MHz.

Wer mit der Ablesegenauigkeit der Kurven nicht so recht zufrieden ist, der setze einfach einen „Frequenzmarker“ und schiebe den z.B. auf diesen Punkt von 11 MHz. In **Bild 10** finden sich alle nötigen Informationen dazu.

Wer immer noch nicht zufrieden ist und mit der Ablesegenauigkeit - auch mit

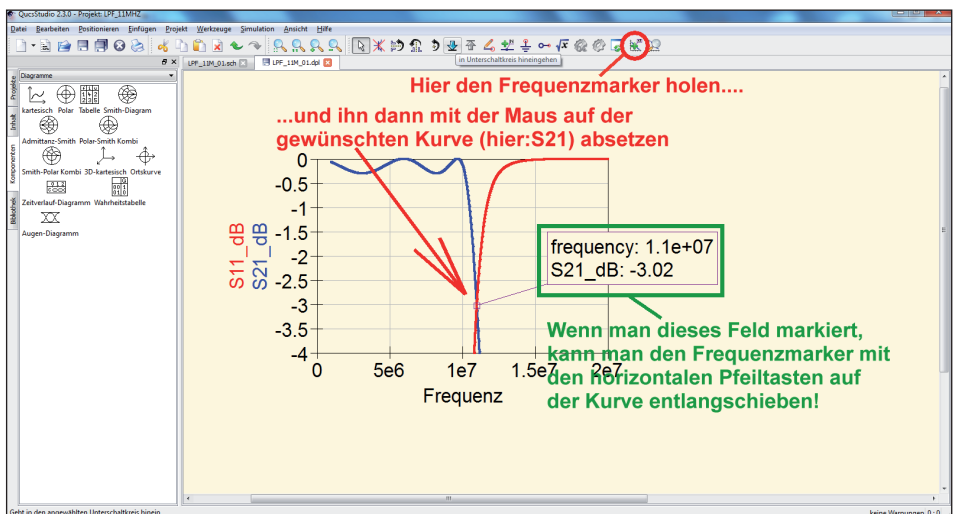


Bild 10: Aber erst mit dem Einsatz des Frequenzmarkers ist eine Detailauflösung möglich (siehe Text)

S-Parameter Simulation

SP1
Type=lin
Start=1MHz
Stop=101MHz
Points=10001

Bild 11: Möchte man nur in kleinen Schritten weitergehen muss man einen neuen Sweep mit hoher Auflösung programmieren ...

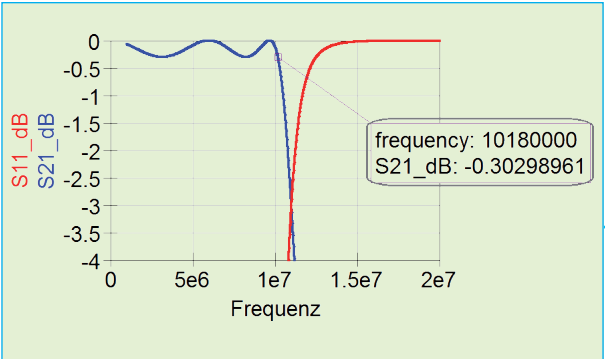


Bild 12: ...aber so richtig Spaß macht das erst, wenn man die Markeranzeige auf 8 Stellen umgestellt hat (siehe Text)

Marker! - hadert, der muss die Zahl der Punkte bei den Simulationsvorgaben im Schaltplan erhöhen. Ein brauchbarer Vorschlag für eine Schrittweite von 10 kHz ist in **Bild 11** zu sehen.

Dazu benötigt man aber auch anschließend eine genauere Marker-Anzeige des Ergebnisses. Also wird (wieder mal) „mit Rechts“ auf das Markerfeld geklickt, „Eigenschaften editieren“ gewählt und die Genauigkeit auf 8 Stellen erhöht (default = 3). **Bild 12** liefert als Beispiel für diese Methode die genaue „Ripple-Grenzfrequenz“ zu 10,018 MHz. Dort übersteigt die Dämpfung zum ersten Mal den Wert des vorgegebenen Ripples mit 0,3 dB.

5. Wiederholung des Projektes ohne den Filtercalculator

Nur zur Übung: Das soll keine Quälerei sein, sondern man lernt dadurch die

Erstellung eines Simulations-Schaltbildes, die Suche nach Bauteilen und der passenden Quelle, die Erstellung von Simulationsvorgaben, den Umgang mit Gleichungen...

Geht man von einemn kleinen Zettel mit einer Skizze der Tiefpass-Schaltung samt den nötigen Bauteilwerten (...findet sich alles in Kapitel 4.2.)

Schritt 1:

Ein neues Projekt wird gestartet und benannt, ein neuer Schaltplan angelegt (Siehe den Anfang von Kapitel 4.2.)

Schritt 2:

Entsprechend **Bild 13** tastet man sich an das Versteck von Spule und Kondensator (in „Komponenten / konzentrierte Komponenten“) heran und zieht 2 Induktivitäten und 3 Kapazitäten in die Arbeitsfläche.

Schritt 3:

Wird ein Bauteil markiert, lässt es sich mit der Maus verschieben und mit „<Control> + <R>“ drehen. Ein rechter Maus-

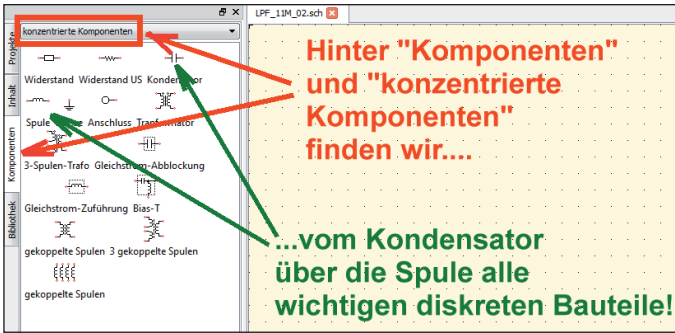


Bild 13:
Selbst ist der Mann!
Also zeichnet man die komplette Schaltung von Hand (siehe Text), wozu man verschiedene Bauteile benötigt

klick und „Eigenschaften editieren“ öffnet (wieder) den Zugang zum Menü, in dem man den korrekten Bauteilwert einträgt. Diese Werte holt man sich, wie erwähnt, aus Kapitel 4.2.

In der oberen Menüleiste gibt es das Erdungssymbol und den „Wire“ zum Verbinden der Teile. Damit wäre man bei **Bild 14**.

Schritt 4:

Jetzt werden noch 2 Ports benötigt. Dazu wechselt man das Bauteil-Menü, denn diese stecken in „Komponenten / Quellen“ als „Leistungsquelle“. Dieses Symbol

wird angeklickt und der gewünschte Port zwei Mal in der Schaltung abgesetzt. Die Portsymbole werden platziert, mit Massezeichen am unteren Ende versehen und an die Filterschaltung angeschlossen.

Schritt 5:

Wieder wechselt man das Menü, denn nun benötigt man „S-Parameter-Simulationen“ aus „Komponenten / Simulationen“. Nach dem Absetzen öffnet man das Eigenschafts-Menü und trägt dort der Reihe nach die Simulationsvorgaben ein. Hier hält man sich ganz genau an Bild 11 aus dem vorigen Kapitel und sollten nun

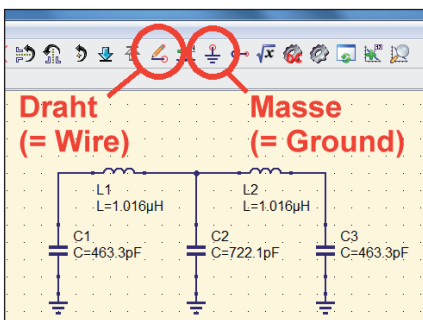


Bild 14: Das sollte der erste Erfolg sein - die nackte Schaltung, noch ohne Ports

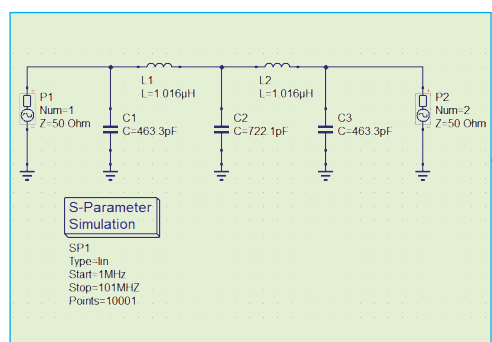


Bild 15: So könnte man schon simulieren,...

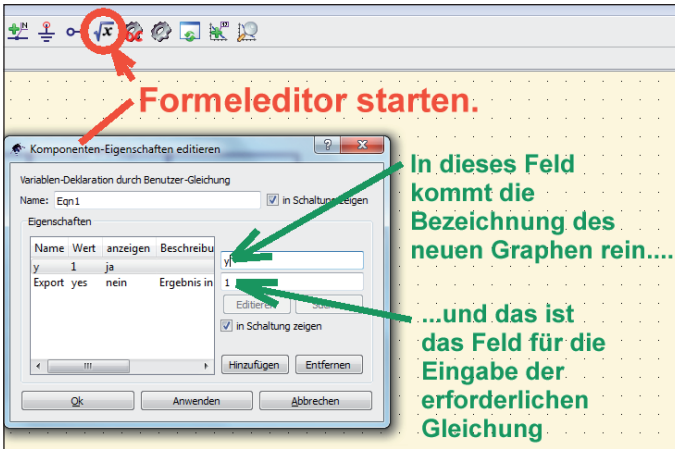


Bild 16:
...wenn man nicht so neugierig wäre und auch noch die dB-Darstellung verlangte

Bild 15 auf dem Bildschirm haben.

Schritt 6:

Jetzt fehlt nur noch die Sammlung an Gleichungen zur S-Parameter-Berechnung. Das entsprechende Vorgehen zeigt **Bild 16**. Erst holt man sich den For-

meleditor auf den Bildschirm (...das Eigenschafts-Menü öffnet sich von selbst). Markiert sind darin die beiden Eingabefelder für den Namen der neuen Größe und die erforderliche Verknüpfung.

Anschließend werden gemäß **Bild 17** folgende drei Formeln programmiert:

$$S11_dB = dB(S[1,1])$$

$$S21_dB = dB(S[2,1])$$

$$S21_phase = wphase(S[2,1])$$

Bevor man abschließt, sollte man noch den ersten Eintrag in der Liste (y = 1) löschen und erst dann „Anwenden“ und OK drücken.

Bild 18 muss nun mit Bild 3 übereinstimmen; es fehlt lediglich der Kommentar aus Bild 3.

Wenn alles vorbereitet ist kann es losgehen - jetzt wird simuliert und dazu auf das Zahnradchen in der Menüleiste geklickt.

Schritt 7:

Er besteht in der Programmierung des Ergebnisdigramms und dazu blättert man

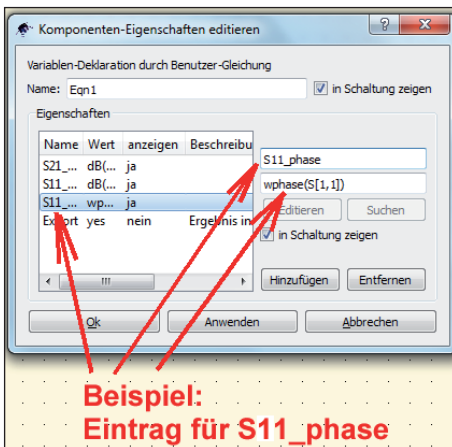


Bild 17: So sollte die Gleichungs-Programmierung der dB- und Phasenanzeige am Ende aussehen

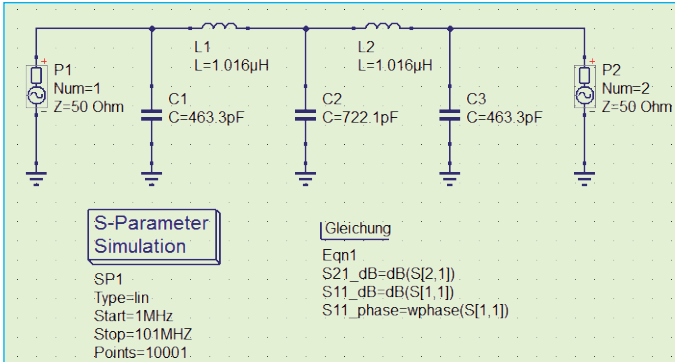


Bild 18:
*Jetzt ist wirklich
alles vorbereitet
und es kann
simuliert werden*

einfach in das vorige Kapitel zurück. Man benötigt die Einstellungen für Bild 8 und so zeigt **Bild 19** exakt denselben Verlauf. Geschäft!

6. Berücksichtigung der Spulenverluste

Beim Aufbau des Musters auf einer Leiterplatte wurden Induktivitäten mit einer gemessenen Güte von $Q = 75$ bei 11 MHz eingesetzt. Dazu gehört ein Reihen-Verlustwiderstand von

$$R_r = X_L / Q = 2 \cdot \pi \cdot f \cdot L / 75 = \\ = 2 \cdot \pi \cdot 11 \text{ MHz} \cdot 1,016 \mu\text{H} / 75 = 0,94 \Omega.$$

Also nimmt man nochmals das Simulations-Schaltbild her, fügt bei jeder Spule einen solchen Widerstand ein (...er stammt aus „Komponenten / konzentrierte Komponenten“) und simuliert nochmals (**Bild 20**).

Für das Ergebnis braucht man das kartesische Diagramm mit passender Dehnung zur Demonstration der Tschebyscheff-Wellen (...siehe vorheriges Kapitel...) und dann sieht man gleich die Bescherung (**Bild 21**):

Die Spulenverluste führen zu einer mit der Frequenz ansteigenden „Grunddämpfung“ im Durchlassbereich (Passband).

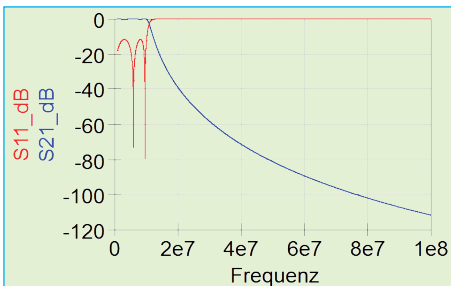


Bild 19: Voll getroffen - dieses Diagramm stimmt mit Bild 8 überein!

7. Die Gruppenlaufzeit (Group Delay)

Bei der Signalübertragung hat man es selten mit einem einzigen rein sinusförmigen Signal zu tun, sondern es sind entweder modulierte Signale (AM, FM, QAM usw.) oder ein Signalgemisch

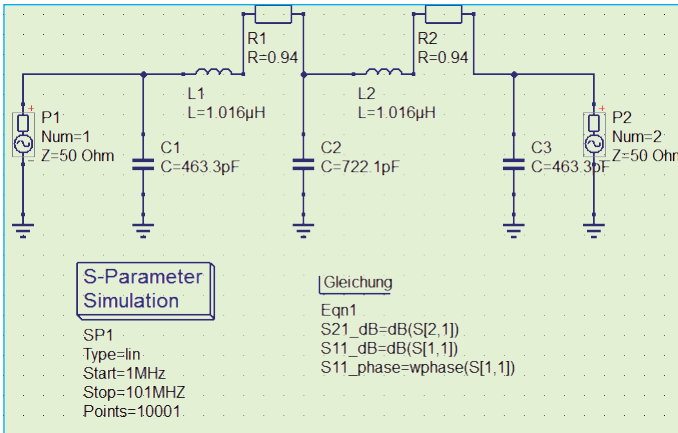


Bild 20:
Mit dieser Schaltungserweiterung sind die Spulenverluste berücksichtigt...

(Sprache, Musik, Videosignale usw.) oder gar impulsförmige Signale.

Zur Erhaltung der Eingangs-Kurvenform ist bei der Übertragung nicht nur die korrekte Übermittlung aller Amplitudenwerte wichtig, sondern auch die Erhaltung der gegenseitigen Phasenlagen aller beteiligten Signale.

Betrachtet man die Simulation in **Bild 22**, bei der die ersten drei Spektrallinien ei-

nes symmetrischen Rechtecksignals addiert werden. Es handelt sich um:

- die Grundwelle mit $U_1 \text{ kHz} = 1 \text{ V}$ und der Frequenz $f = 1 \text{ kHz}$
- die dritte Harmonische mit $U_3 \text{ kHz} = 1/3 \times (U_1 \text{ kHz})$ und der Frequenz $f = 3 \times f_1 = 3 \text{ kHz}$
- die fünfte Harmonische mit $U_5 \text{ kHz} = 1/5 \times (U_1 \text{ kHz})$ und der Frequenz $f = 5 \times f_1 = 5 \text{ kHz}$

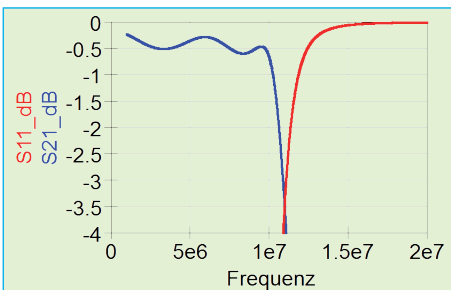


Bild 21: ...und sie führen zu einer zusätzlichen „Grunddämpfung im Durchlassbereich“, die mit der Frequenz ansteigt

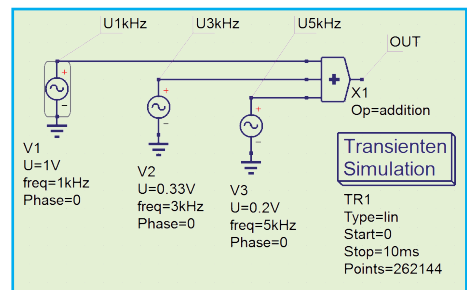


Bild 22: So werden die Grundwelle, die dreifache und die fünffache Frequenz zu einem Rechtecksignal addiert

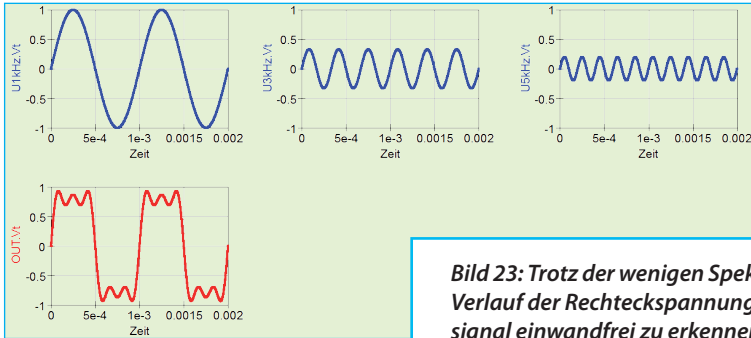


Bild 23: Trotz der wenigen Spektrallinien ist der Verlauf der Rechteckspannung beim Summensignal einwandfrei zu erkennen

Alle Signale weisen dieselbe „Nullphase“ auf.

Hinweis:

Die Addierstufe findet man als „Operation“ unter „Komponenten / Systemkomponenten“. Dann öffnet man ihre Eigenschaften und stellt auf 3 Eingänge um.

Wie **Bild 23** zeigt, ist trotz des geringen Oberwellenanteils der grundsätzliche Kurvenverlauf des Rechtecks bei der Summenspannung eindeutig zu erkennen.

Nun lässt man die Amplituden der Harmonischen unverändert, verschiebt je-

doch ihre Phasenlagen gegeneinander. Dazu lässt man die dritte Harmonische mit „-120 Grad“, die fünfte Harmonische dagegen mit „-200 Grad“ beginnen (**Bild 24**). Das führt zu einer totalen Veränderung der Kurvenform, bei der das ursprüngliche Rechtecksignal in **Bild 25** fast nicht mehr zu erkennen ist - obwohl die Amplituden der beteiligten Spektrallinien beibehalten wurden!

Also heißt die Forderung:

Durchläuft ein solches „Gruppensignal“ einen Baustein oder eine Übertragungsstrecke, dann dürfen zwar Zeitverzögerungen auftreten, aber die ursprünglichen Phasenlagen (der Einzelsignale untereinander) müssen am Ausgang noch korrekt stimmen. Die korrekte Formulierung lautet damit:

Eine Signalübertragung ohne Verformung des Kurvenverlaufs erhält man nur bei konstanter „Gruppenlaufzeit = Group Delay“ des Systems.

Für Spezialisten:

Dazu muss die Phasenverschiebung zwischen Eingang und Ausgang streng linear mit der Frequenz zunehmen.

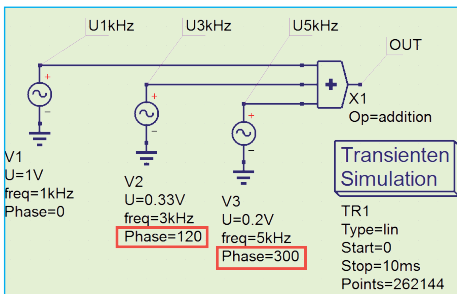


Bild 24: Aber wehe, man verschiebt die Phasenlagen der Anteile gegeneinander....

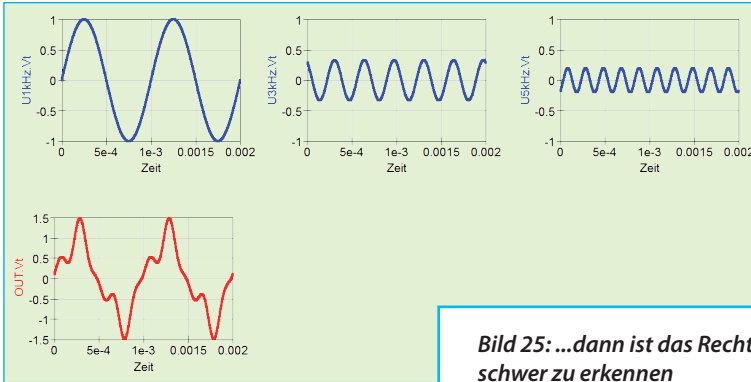


Bild 25: ...dann ist das Rechteck nur noch schwer zu erkennen

Nun soll wieder vorliegendem Tiefpass mit Verlusten (aus dem vorherigen Kapitel) auf den Zahn geföhlt und das eben Besprochene an diesem Beispiel demonstrieren werden. Entscheidend ist, dass bereits die Gleichung für die Bestimmung der Phase von S21

$$S21_phase = wphase(S[2,1])$$

vorhanden ist, denn aus der Phase wird nach der Simulation die Gruppenlaufzeit berechnet.

Dazu holt man sich ein kartesisches Diagramm auf die Arbeitsfläche und trägt bei den Eigenschaften des neuen Graphen `gdelay(S[2,1])` ein;

Nach einem Klick auf „Anwenden“ erhält man **Bild 26**.

Man erkennt, dass die Gruppenlaufzeit etwa bis zu 75 % der Durchlass-Grenze nur um ca. 10 ns um den Mittelwert von 70 ns herum schwankt, ab da steigt sie massiv an.

Wie wirkt sich das in der Praxis aus? Dazu speist man den Tiefpass (ohne Spulenverluste) mit einem symmetrischen

Rechtecksignal, das eine Periodendauer von 500 ns aufweist (...das ergibt $f_{\text{pulse}} = 2$ MHz). Dazu löscht man die beiden Ports sowie die Aufforderung zur S-Parameter-Simulation aus dem Schaltbild und ändert:

Port 1 wird durch die Quelle „Rechteck-Spannung“ (aus „Komponenten / Quellen“) ersetzt und diese gemäß **Bild 27** programmiert (das ergibt ein Signal mit -1 V / +1 Volt und 2 MHz)

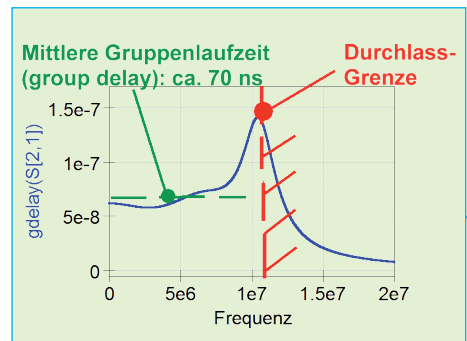


Bild 26: Das Gruppenlaufzeit-Verhalten des Tiefpasses; bis ca. 8 MHz ist es gar nicht so schlecht, aber dann...

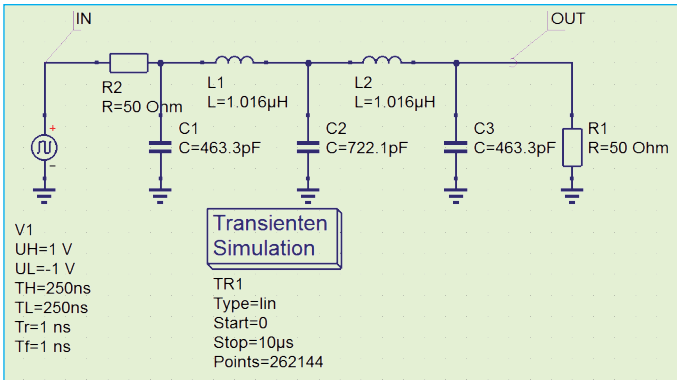


Bild 27:
*So speist man
die Schaltung
mit einem 2 MHz-
Rechteck-Signal...*

Port 2 wird durch einen Abschlusswiderstand mit 50 Ω ersetzt, der den fehlenden Innenwiderstand von 50 Ω bei der Quelle in der Schaltung ergänzt.

An der Quelle und am Abschlusswiderstand werden passende Labels angebracht.

Jetzt braucht man noch eine Transienten-Simulation von 0 bis 10 Mikrosekunden und 262 144 Punkten, die Details sieht man ebenfalls in Bild 27.

Achtung:

Wenn man das Menü für die Transienten-Simulation geöffnet hat, sollte man noch einen Blick auf die darin enthaltene Karteikarte „Eigenschaften“ werfen. Bei der Integrationsmethode sollte nämlich „gear 2“ eingestellt werden, sonst kann es Probleme und einen Simulationsabbruch geben.

Aber jetzt wird simuliert und das Ergebnis in **Bild 28** für $t = 0$ bis 2 μs ist fast selbst erklärend. Der Einfluss der Gruppenlaufzeit-Verzerrungen ist deutlich als Überschwinger(= „Geklingel“) zu erkennen.

8. Ein Blick auf die FFT und die Simulation eines Spektrums

Das ist eine recht einfache Übung, denn man kann die eben simulierte Tiefpass-Schaltung (mit idealen Spulen) ohne Änderungen weiterverwenden und muss nicht einmal eine Gleichung im Schaltbild hinzufügen! Nach der Simulation im Zeitbereich kann man sich eine Freude machen und das Diagramm

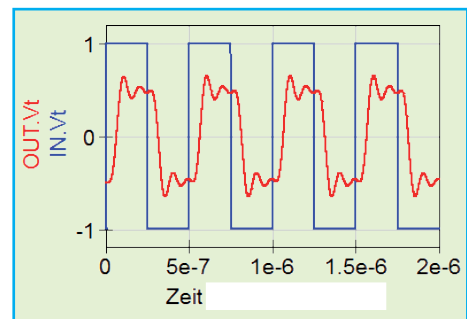


Bild 28: ...und so reagiert der Tiefpass darauf

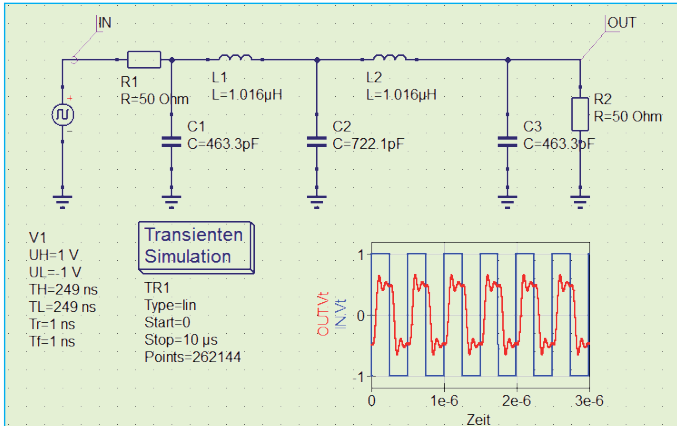


Bild 29:
Das ist einfach
wunderbar: Das
Ergebnis-Diagramm
kann man einfach
neben den Strom-
laufplan kopieren

mit dem Verlauf von Eingangs- und Ausgangsspannung auch auf die Schaltplan-Seite kopieren (**Bild 29**).

Für eine FFT reicht beim Ergebnis-Bildschirm das Aufrufen eines neuen kartesischen Diagramms, das nach Eintippen folgender Beziehung im Feld „Eigenschaften des Graphen“, gefolgt von „Neuer Graph“, zur Darstellung des Spektrums am Eingang überredet wird:

`time2freq(IN.Vt)`

Ebenso verfährt man mit dem Ausgangssignal, bei dem in einem neuen Diagramm die Formel

`time2freq(OUT.Vt)` erforderlich ist.

Wenn man dann noch den Frequenzbereich auf 0 bis 20 MHz einstellt, sieht man in **Bild 30** sehr schön die Filterwirkung des Tiefpasses beim Vergleich von Eingang und Ausgang.

Der auswertbare Amplitudenumfang ist leider wegen der linearen Darstellung sehr begrenzt und den „Amplitudengang“ bekommt man erst so richtig schön mit

einer dB-Skala zu sehen. Also holt man nacheinander nochmals zwei kartesische Diagramme auf die Arbeitsfläche und in ihre Felder „Eigenschaften des Graphen“ müssen folgende Einträge:

`dB(time2freq(IN.Vt))`

für das Eingangsspektrum und

`dB(time2freq(OUT.Vt))`

für das Ausgangsspektrum.

Die überzeugende Wirkung dieser Maßnahme zeigt **Bild 31**, die Wirkung des Tiefpasses für einen Frequenzbereich von 0 bis 40 MHz und einen Amplitudenbereich von -60 dB ... Null dB. Aber eine Sache ist immer ein Problem bei der FFT:

Der Signalverlauf in der Time Domain beginnt abrupt bei Null und endet ebenso abrupt bei $t = 10 \mu\text{s}$. Diese abrupten „Schaltvorgänge“ sorgen leider für einmalige zusätzliche Spektren, die den gesamten Frequenzbereich „mit Energie ausfüllen“ und die dargestellte Dynamik verkleinern.

Eine bekannte Abhilfe besteht darin, über

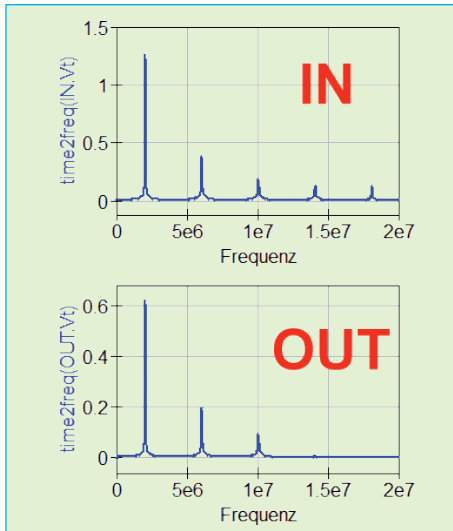


Bild 30: So eine FFT ist einfach schön - aber die lineare Darstellung zeigt we-
nig Dynamik

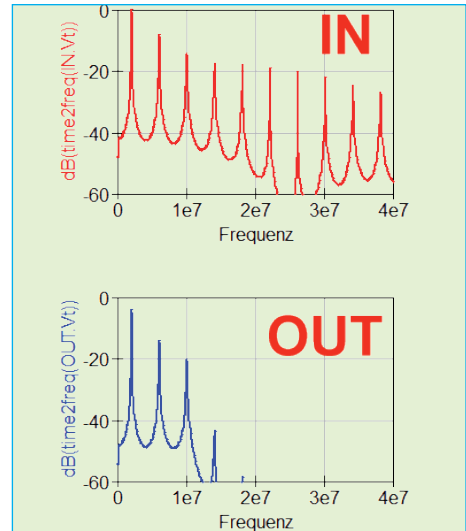


Bild 31: Der Umstieg auf die dB-Darstellung behebt dieses Dynamikproblem, aber die Schaltvorgänge machen noch Ärger (siehe Text)

den zeitlichen Verlauf eine „Fernsterfunktion“ zu legen. Damit wird der Zeitverlauf multipliziert und bei den meisten „Window“-Funktionen werden dazu die Samples am Anfang und Ende zum Teil bis Null abgeschwächt. Sie steigen jetzt nur noch sanft auf ihren tatsächlichen Wert an bzw. klingen ebenso sanft aus - und auf diese Weise hat man die Schaltvorgänge fast eliminiert. Bezahlt wird das mit einer geringen Anzahl an „voll wirksamen Samples“ und das ist gleichbedeutend mit einer kürzeren Simulationszeit. Und da folgende Beziehung gilt:

$$\text{Frequenzauflösung} = \text{Linienbreite} = \text{Startfrequenz des simulierten Spektrums} = 1 / \text{Simulationszeit}$$

bekommt man nun plötzlich breitere Spektrallinien zu sehen. Keine Rose ohne

Dornen...

Bei „qucsStudio“ stehen fünf Fensterfunktionen zur Verfügung und nach meiner Erfahrung ist Funktion Nr. 3 (Hanning) nicht die Schlechteste. Also wiederholt man nochmals die FFT, aber die erforderlichen Gleichungen enthalten nun den erforderlichen Hinweis auf „Window Nr. 3“: Man schreibt beim Ergebnis-Diagramm für den Eingang:

$$\begin{aligned} &\text{dB}(\text{time2freq}(\text{IN.Vt}, [3])) \\ &\text{oder} \\ &\text{dB}(\text{time2freq}(\text{IN.Vt}, 3)) \end{aligned}$$

Und das Ausgangsspektrum erhält man mit der Gleichung

$$\begin{aligned} &\text{dB}(\text{time2freq}(\text{OUT.Vt}, [3])) \\ &\text{oder} \\ &\text{dB}(\text{time2freq}(\text{OUT.Vt}, 3)). \end{aligned}$$

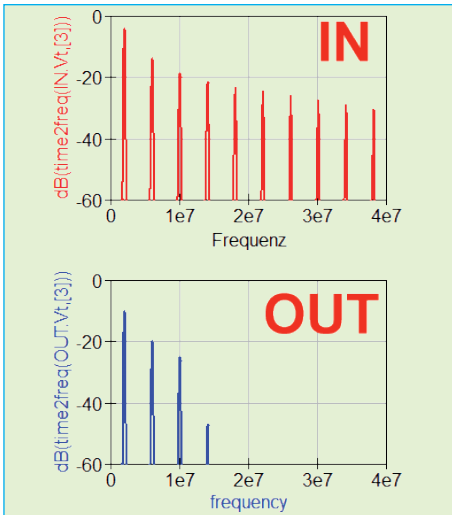


Bild 32: Da hilft die „Fensterung“ mit einem Hanning-Window. Aber dadurch klemmt es an einer anderen Stelle (siehe Text)

Die Verbesserung ist beim Vergleich von **Bild 32** mit dem vorigen Simulationsergebnis (Bild 31) deutlich zu erkennen.

Allerdings hat diese Rose ebenfalls wieder Dornen: Durch die Fensterfunktion und den dadurch geänderten Amplitudenverlauf im Zeitbereich stimmen zwar die Amplitudenabstände der einzelnen Linien zueinander und die Dynamik wurde größer.

ABER:

Konnte man in Bild 31 noch die absoluten Amplituden aller Linien „in Volt“ herausholen (...die Nulllinie der vertikalen Achse entspricht dort „Null dBVolt“ mit einer Bezugsspannung von 1 Volt für Null dB), so ist nun diese „Eichung“ in Bild 32 durch die „Fensterung“ verloren gegangen.

9. Abschließende Bemerkung

Was man jetzt glücklich geschafft hat, ist der Zugang zum Programm und zur Bedienung. Aber schon dieser kurze Spaziergang zeigt die vielen Möglichkeiten und wer sich nun die Mühe macht, die ganzen Komponenten-Menüs zu durchforsten (und das erwähnte Tutorial durch zu arbeiten), dem wird schnell schwindelig. Man ahnt: die komplette Beherrschung des Programms grenzt fast an ein Lebenswerk - aber sie bietet auch genau die Möglichkeiten, die das Leben eines Entwicklers leichter und faszinierender machen. Die etwas andersartige Bedienung hält man irgendwann locker aus und das Tutorial dürfte beim Einarbeiten eine echte Hilfe sein.

Aber das Wichtigste zum Schluss:

Tausend Dank an Michael Margraf, DD6UM, für die unendliche Mühe und Arbeit, eine solche Software zu entwickeln und zu pflegen. Wir können es würdigen!

10. Literatur:

[1] Gunthard Kraus, Tutorial zur Schaltungssimulation mit qucsStudio. Kostenlos von der Homepage www.gunthard-kraus.de herunterzuladen

Beschaffung der neuesten Software-Version, Informationen über sonstige Neuigkeiten, weitere Anwendungsbeispiele usw.:

Nach der Eingabe von „qucsStudio download“ in eine Suchmaschine wird man sofort zur Homepage des Programmautors Michael Margraf, DD6UM, geleitet. Dort findet man alles - auch Updates ...