QucsStudio Tutorial

Teil 1:

Simulationen in der Time Domain und in der Frequency Domain

Tutorial Version 1.7

Autor:
Gunthard Kraus, DG8GB, Oberstudienrat i. R.
Gastdozent an der Dualen Hochschule Baden-Württemberg in Friedrichshafen

(mit Unterstützung durch den Programmautor Michael Margraf, DD6UM)

Email: mail@gunthard-kraus.de
Homepage: www.gunthard-kraus.de

10. März 2019

Inhaltsverzeichnis

Seite

1. Einführung 9

2. Beschaffung und Installation von qucsstudio 10

3. Erstes Projekt: RC-Tiefpass 11
   - 3.1. Simulation in der Frequency Domain: die Übertragungsfunktion 11
     - 3.1.1. Lineare und logarithmische Darstellung des Ergebnisses 11
     - 3.1.2. Was steckt hinter „dB“? 17
     - 3.1.3. „dB“ Darstellung und Phasendarstellung beim Ergebnis 18
     - 3.1.3. Umrechnung eines Spannungswertes in „dBm an 50 Ohm“ 19
   - 3.2. Simulationen in der Time Domain 21
     - 3.2.1. Impulsantwort des RC-Tiefpasses 21
     - 3.2.2. Sprungantwort des Tiefpasses 23
       - 3.2.2.1.Lösung mit Puls-Quelle 23
       - 3.2.2.2. Lösung über ein „PWL“ (= piece wise linear) – Signal 24
     - 3.2.3. Periodische Signale am Eingang 26
       - 3.2.3.1. Speisung mit einer Sinusspannung 26
       - 3.2.3.2. Speisung mit einem Rechteck-Signal 27
       - 3.2.3.3. Speisung mit einem Dreieck-Signal 29
       - 3.2.3.4. Differenz-Messungen 30
       - 3.2.3.5. Strom-Messung 31
   - 3.3. Die FFT (= Fast Fourier Transformation) als Brücke zwischen Zeit- und Frequenzbereich 32
     - 3.3.1. Zusammenfassung der Grundlagen 32
     - 3.3.2. Ein praktisches Beispiel 33
     - 3.3.3. Der Kampf mit den Schaltvorgängen 38
       - **3.3.3.1.** Eine Window-Funktion muss her 38
       - **3.3.3.2. Umrechnung von Spannungswerten in „dBm an 50 Ohm“** **41**

4. Zweites Projekt: Signalformen und Oberwellen = eine kleine FFT-Übung 43
   - 4.1. Grundlagen 43
   - 4.2. Verzerrungen bei periodischen Signalen 44
   - 4.3. Simulation des Spektrums eines einmaligen Pulses 46
   - 4.4. Simulation des Frequenzspektrums bei einem periodischen Pulssignal 49
   - 4.5. Der ideale Sinus 50
   - 4.6. Der unsymmetrisch verzerrte Sinus 51

4.7. Der symmetrisch verzerrte Sinus 52

5. Drittes Projekt: Diodenschaltungen 53
5.1. Einpuls-Gleichrichter mit idealer Diode 53

5.2. Verwendung von Dioden aus dem Bauteil-Vorrat 54

5.3. Eine wichtige Sache: Verwendung von SPICE-Modellen aus dem Internet 55

6. Untersuchung einer Verstärkerstufe mit Transistor 60
6.1. Arbeitspunktbestimmung über eine DC-Analyse 60

6.2. Transienten-Simulation: Ansteuerung mit einem Sinus-Signal 62

6.3. Spektraldarstellung von OUT2 in dB 64

6.4. AC-Sweep zur Bestimmung der Frequenzgänge 66

6.5. Zusammenfassung: Übungs - Beispiel mit dem Transistor BFR92A 67
6.5.1. Beschaffung und Import des SPICE-Modells aus dem Internet 67
6.5.2. Ein Schlusstest 69

7. OPV-Schaltungen 71
7.1. Umkehrende Schaltung 71

7.2. Nicht umkehrende Schaltung 72

7.3. Der OPV als Analog-Addierer 73

7.4.. Import eines OPV-Spice-Models aus dem Internet 75

7.5. Aktive Filterschaltung mit OPV 78

8. Simulation von Digitalschaltungen 83
8.1. Verwendung von Beispielen aus der Qucsstudio-Homepage 83

8.2. Untersuchung des Beispiels „phase_shifter_90deg.sch“ 84

8.3. Ein dreistufiger Frequenzteiler mit D-Flipflops 85

9. Arbeiten mit S-Parametern 86
9.1. Grundlagen der S-Parameter 86

9.2. Praxis-Beispiel: Tschebyschef-Tiefpass mit einer Grenzfrequenz von 11 MHz 87
9.2.1. Zuerst etwas über Filterschaltungen 87
9.2.2. Pflichtenheft für das Projekt 89
9.2.3. Filterentwurf mit qucsstudio: Einsatz des Filtercalculators und der S-Parameter-Simulation 89
9.2.4. Berücksichtigung der Spulenverluste 91
9.2.5. Die Sache mit der Group Delay (= Gruppenlaufzeit) 92

10. Das Smith-Diagramm („smith chart“) 94
10.1. Problemstellung 94
10.2. Wie entsteht ein Smith Chart? 94
10.3. Kleine Tücken bei den Winkelangaben im Smithchart 97
10.4. Wie erhalte ich die Impedanz als Wirk- und Blindwiderstand (= Serienschaltung) bei einem Diagramm-Punkt im Smithchart? 98
10.5. Wie erhalte ich den Wirk- und Blindanteil des Eingangs-Leitwertes (= Parallelschaltung = Admittanz) bei einem Diagramm-Punkt im Smithchart? 100

11. Microstrips (= Streifenleitungen) 101

12. Praxis-Projekt: MMIC-Breitbandverstärker bis 3 GHz 103
12.1. Touchstone-Files (S2P-Files) 103
12.2. Simulation der S-Parameter bis 6 GHz 104
12.3. Untersuchung der praktischen MMIC-Gesamtschaltung mit dem ABA52563 107
12.4. S11 und S22 der MMIC-Schaltung im Smithchart 108
12.5. Praktische Ausführung des Verstärkers 110
12.5.1. Ein Blick auf die Leiterplatte 110
12.5.2. Entwurf des „**Grounded Coplanar Waveguides**“ für die Platine 112
12.6. Das Verhalten der Gesamtschaltung 114
12.6.1. Simulation der S-Parameter 114
12.6.2. Ermittlung des **Rollet Stabilitätsfaktors „k**“ 115
12.6.3. Die Stunde der Wahrheit: Messergebnisse des **Vektoriellen Network-Analyzers** 117

13. Entwicklung eines Low Noise Amplifiers für 1GHz..... 1,7 GHz mit einer maximalen Rauschzahl von 0,4dB 119
13.1. Überblick 119
13.2. Ausgangs-Schaltung 120
13.3. Weiterentwicklung der Schaltung 122
13.4.Der Prototyp 125
13.5. S-Parameter-Messungen am Prototyp ..126
13.6. Das Rauschen 127

14. RF-Mixer (= HF-Mischer) 130
14.1. Einführung 130
14.2. Simulation eines echten Multiplizieres 132
12.2.1. Der Multiplizierer als **Switched Single Balanced Mixer** 133
12.2.2. Der Multiplizierer als **Switched Double Balanced Mixer** 136

14.3. Simulation eines echten Double Balanced Mixers (= Ringmodulator) aus einem 137MHz-Wettersatelliten-Empfangskonverter 139

14.4. Der IP3-Punkt - sein Zweck und seine Ermittlung 142

15. Entwicklung eines schmalen 10,7 MHz -Bandpasses (= Narrow Bandpass = Coupled Resonator Type) 145
15.1. Vorbemerkungen 145

15.2. Installation von DOS-Box, DOS-Shell und fds.exe 145

15.3. Pflichtenheft für den 10,7 MHz – Bandpass 146

15.4. Entwicklungs-Strategie 147

15.5. Erforderliche Filterschaltung und Spulengüte 147

15.6. Der Kampf mit den Spulen 152

15.7. Der erste Test: Filterprototyp mit N = 3 154

15.8. Das endgültige Ziel: die fünf-polige Version 157

15.9. Zum Vergleich: Entwurf mit dem neuen Filterdesigner von qucsstudio (ab Version 2.4.0) 160

16. Entwicklung einer WLAN (= 2,45 GHz) – Patchantennne für lineare Polarisation 163
16.1. Zuerst etwas Antennen-Grundlagen 163

16.2. Grundlagen der Patchantennen 164

16.3. Messung der verschiedenen Antenneneigenschaften 166

16.4. Entwurf einer WLAN-Antenne für lineare Polarisation 168
16.4.1. Patch-Dimensionierung mit qucsstudio 168
16.4.2. Anpassung der Antenne an 50 Ω 172

17. Ein Ausflug in die Modulation 177

17.1. Einstieg 177

17.2. Die Amplitudenmodulation 180
17.2.1. Hintergrund 180
17.2.2. Spektrum des AM-Signals 183
17.2.3. Demodulation eines AM-Signals 185
17.2.3.1. Der klassische AM-Diodendemodulator 185
17.2.3.2. AM-Demodulation mit einem Produktdetektor 186

17.3. SSB = Single Side Band Betrieb 187

17.4. Reale, imaginäre und komplexe Signale ..188

17.5. Analytic Pairs 190

17.6. Half-Complex Mixer zur Erzeugung eines SSB-Signals 190

17.7. Demodulation von amplitudenmodulierten IQ-Signalen 191

17.8. Demodulation von SSB-Signalen 192

17.9. Amplitude Shift Keying = ASK 193

17.10. Die Phasenmodulation 194
17.10.1. Erzeugung einer Phasenmodulation 194
17.10.2. Ergebnis bei sinusförmiger Informationsspanung 196
17.10.3. Das Frequenzspektrum bei Phasen- und Frequenzmodulation 197
17.10.4. Einige wichtige Begriffe zur Frequenz und Phasenmodulation 198
(Frequenzhub, Phasenhub, Kanalbreite)

17.11. FSK = Frequency Shift Keying = von der Phasen- zur Frequenzmodulation 199

17.12. QAM = Quadrature Amplitude Modulation 201
16.12.1. Erzeugung einer 4-QAM 201
16.12.2. Erzeugung einer 16-QAM 203
16.12.3. Demodulation von QAM-Signalen 205

18. Rauschen 206

18.1. Erste Bekanntschaft mit Rauschsignalen 2066
18.2. „Rauschen“ -- woher kommt das? 207

18.3. Weitere Rauschquellen 210

18.4. Weißes Rauschen zur Messung einer Übertragungs – Funktion 211
18.4.1. Zur Erinnerung: die S-Parameter 211
18.4.2. Ermittlung von S21 mit weißem Rauschen in der Simulation 213

18.5. Erzeugung eines Rauschsignals im Zeitbereich mit qucsstudio 215

18.6. Analyse einer OPV-Schaltung (einschließlich Noise-Simulation) 217

19. Untersuchung eines Sinus-Oszillators 220

19.1. Oszillator-Prinzip ..220

19.2. Simulierte Schaltung 221

1. Einführung

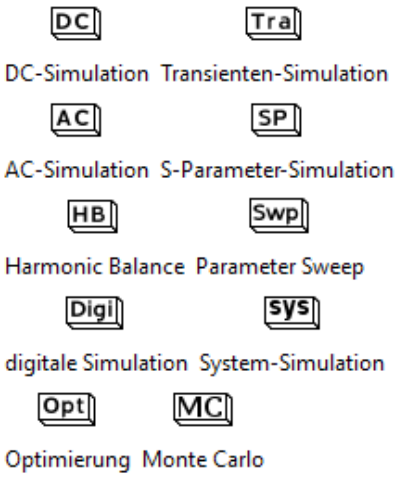
Zunächst etwas zur Verwirrung:

Es gibt „Qucs“ und „Qucs-Studio“.

Deshalb nehmen wir uns zuerst „Qucs“ vor.
„**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 fördern sollen. **Die Software arbeitet wahlweise entweder mit deutscher oder englischer Bediener-Oberfläche.**

(...Umstellung unter „Datei / Applikations-Einstellungen / Sprache).

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 der Parameter-Sweep. **Was aber einen 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...**
Da macht das nebenstehende Bild mit den Simulationsmöglichkeiten geradezu den Mund wässrig…



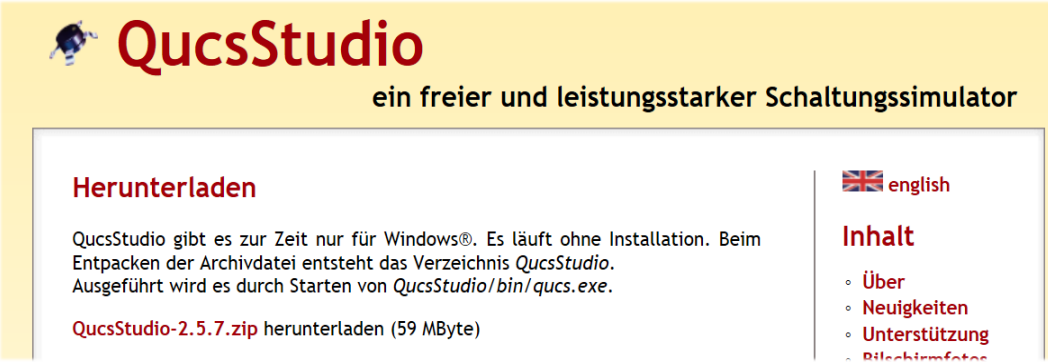
Wenn man sich mit Schaltungs - Simulations - Programmen etwas auskennt, dann fällt einem gleich die hier herrschende Philosophie auf:
Es geht bei diesem Programm nicht darum, dem Schaltungsentwickler im Labor ein Werkzeug an die Hand zu geben, mit dem sich gegenüber einem Konkurrenzprodukt wieder einige 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 einer Schaltung gründlich auf den Zahn gefühlt werden kann (und folglich kann damit nicht nur die Krone des Zahns, sondern auch seine Wurzel sehr genau untersucht werden).
Das bedeutet, dass sehr viele Funktionen für die Simulation, aber auch unterschiedlichste Darstellungsmöglichkeiten bei den Ergebnissen bereit stehen und die Bedienung recht einfach ist, denn die Arbeit soll Freude machen und der Neugier wie auch der Fantasie Raum lassen.

Allerdings will und kann es nie verleugnen, dass sein geliebtes Einsatzgebiet die HF- / Nachrichten- / Mikrowellen- und Übertragungstechnik ist. Muss man wissen…

Dass manche Dinge deshalb etwas -- Na ja, einigen wir uns auf „leicht gewöhnungsbedürftig“ -- ausfallen, lässt sich nicht vermeiden, aber sie können wegen der Leistungsfähigkeit und der vielen gebotenen Möglichkeiten problemlos in Liebe ertragen werden.
Allerdings passierte irgendwann dasselbe wie in Erbengemeinschaften: 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 dann „**QucsStudio**“. Deshalb gibt es natürlich bisweilen leichte Unverträglichkeiten zwischen beiden Programmen. Man hat aber den Eindruck, dass Weiterentwicklungen nur noch beim „QucsStudio“ passieren und deshalb befassen wir uns mit dieser Programmversion. Vieles, was jetzt folgt, ist auch bei Qucs verwendbar…aber, wie schon gesagt….manchmal kann es klemmen…..

Diese Tutorial-Version 1.7. basiert auf der Software-Version 2.5.7. Darin sind einige Dinge verbessert, erweitert oder neu dazu gekommen!

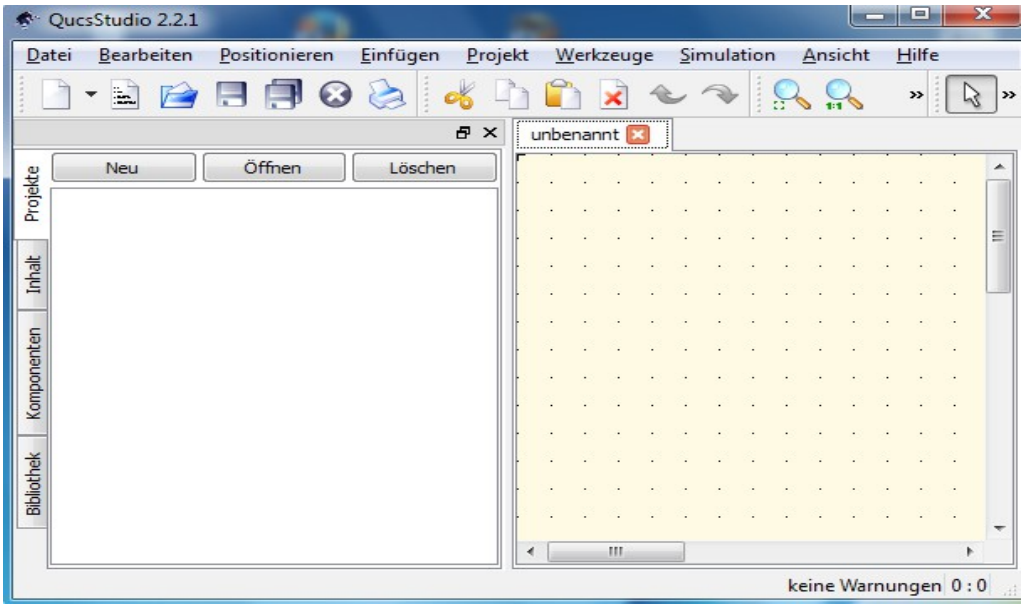
2. Beschaffung und Installation von QucsStudio



Da reicht die Eingabe „**qucsstudio**“ in einer Suchmaschine, um auf der Seite „**QucsStudio Homepage – DD6UM**“ zu landen.

Die aktuellste Version (…hier ist es 2.5.7) wird heruntergeladen und entzippt.

Der Rest ist reine Installationsroutine und läuft ohne Probleme.
Holt man sich anschließend den nebenstehenden Icon auf den Bildschirm und klickt darauf, dann sollte folgender Bildschirm erscheinen:



Man sollte jedoch nicht versäumen, sich **zusätzlich in der Homepage umzusehen:**
da gibt es da Bildschirmfotos und Videos, Beispiele, häufige Fragen usw. Und das hilft sehr oft weiter.
Noch eine Anmerkung:
Das Programm erzeugt keine Einträge in Windows (...also in der Registry) und kann deshalb ganz einfach durch Markieren seines Programm-Ordners, gefolgt von „Delete“, wieder von der Festplatte gelöscht werden.

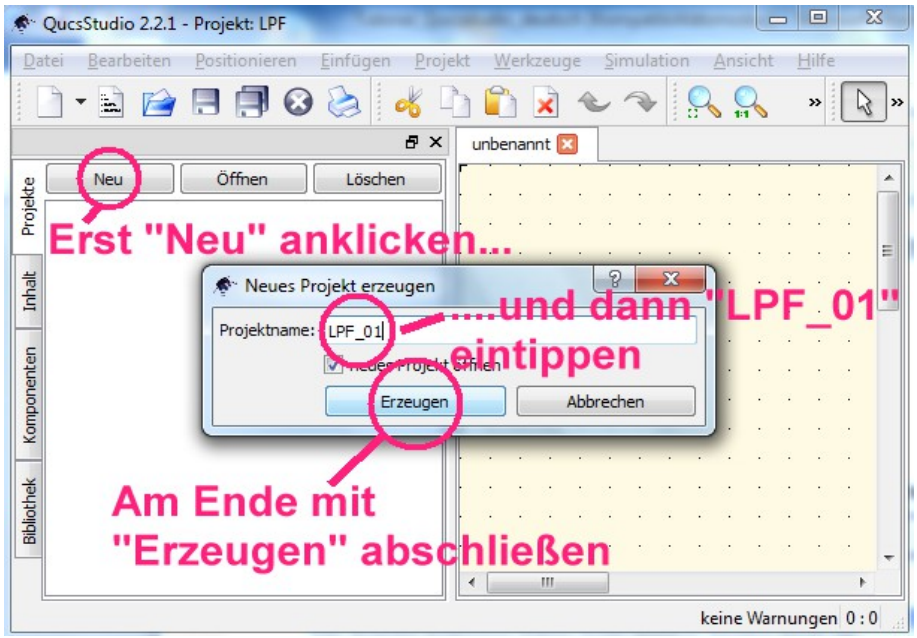
Persönliche Bemerkung des Autors:
Nach Programm-Abstürzen (..gibt es halt mal beim Experimentieren und Tüfteln...) empfiehlt sich unbedingt dieses Löschen, gefolgt von einer Neu-Installation. Ich habe nämlich nach solchen Abstürzen oft Stunden lang mit irgendwelchen eigenartigen Effekten und Fehlern an ganz anderen Ecken gekämpft, die nach der Neu-Installation wie durch Zauberhand wieder verschwunden waren. Muss man auch erst herausfinden...
Deshalb sollte man auch einmal im Monat grundsätzlich Alles löschen und die neueste Programmversion aus der qucsstudio – Homepage installieren.

3. Erstes Projekt: RC-Tiefpass

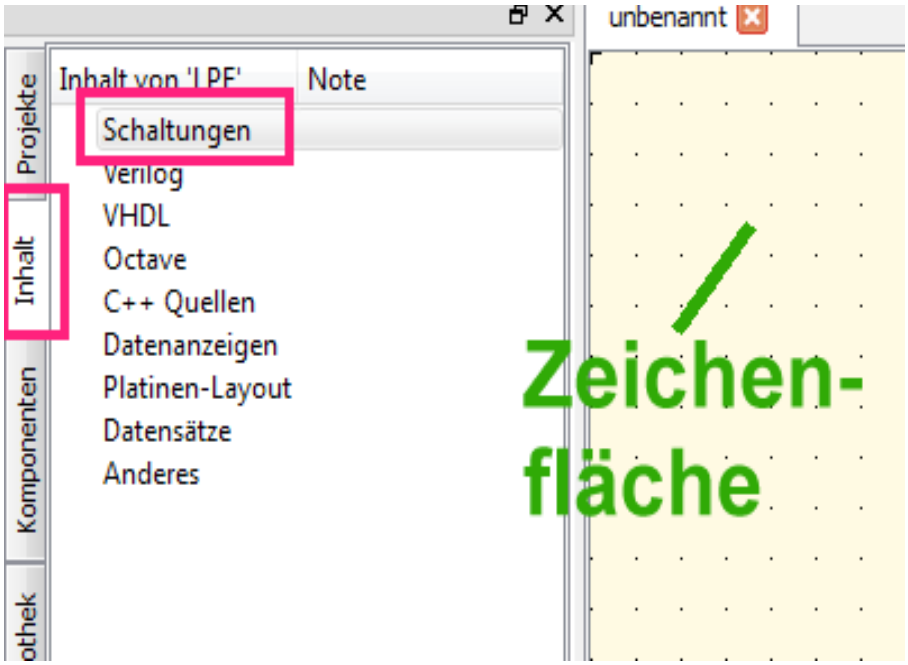
3.1. Simulation in der Frequency Domain: die Übertragungsfunktion

3.1.1. Lineare und logarithmische Darstellung des Simulations-ergebnisses

Da nehmen wir uns etwas Einfaches, aber von den Simulationsmöglichkeiten doch sehr Vielseitiges vor, nämlich einen **RC-Tiefpass**. Wir interessieren uns zunächst für den „**Frequenzgang der Übertragungsfunktion**“, also das Verhältnis von Ausgangsspannung zu Eingangsspannung bei unterschiedlichen Frequenzen des sinusförmigen Eingangssignals.



1. Schritt:
Das Programm wird gestartet und nach der nebenstehenden Reihenfolge das Projekt angelegt.

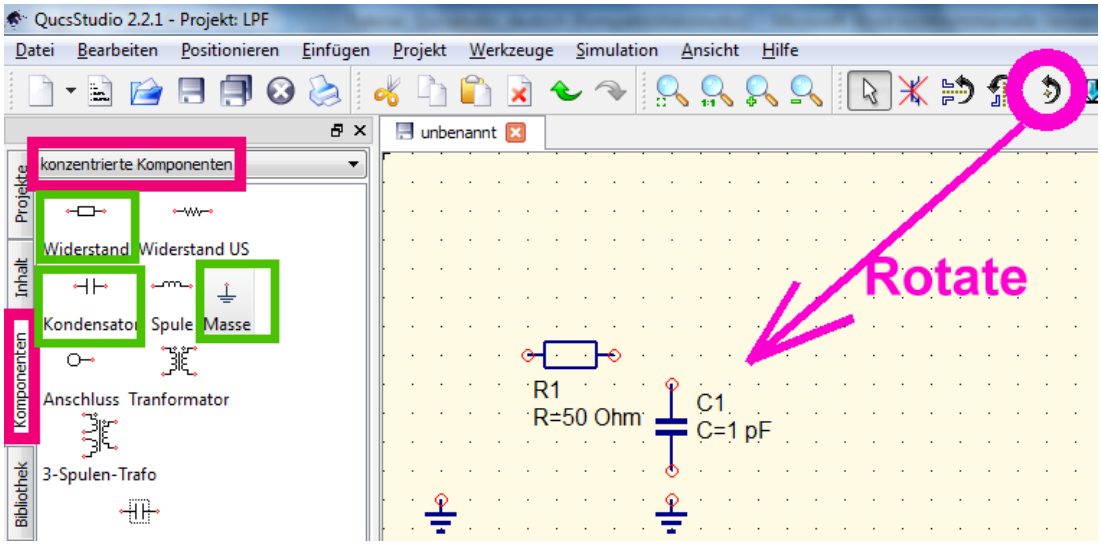


Noch ein kurzer Kontrollblick: Wurde automatisch auf „**Inhalt**“ umgeschaltet und die Option „**Schaltungen**“ gewählt? **Dann kann man diese neue Schaltung speichern, z.B. unter dem Namen**

LPF_01.sch

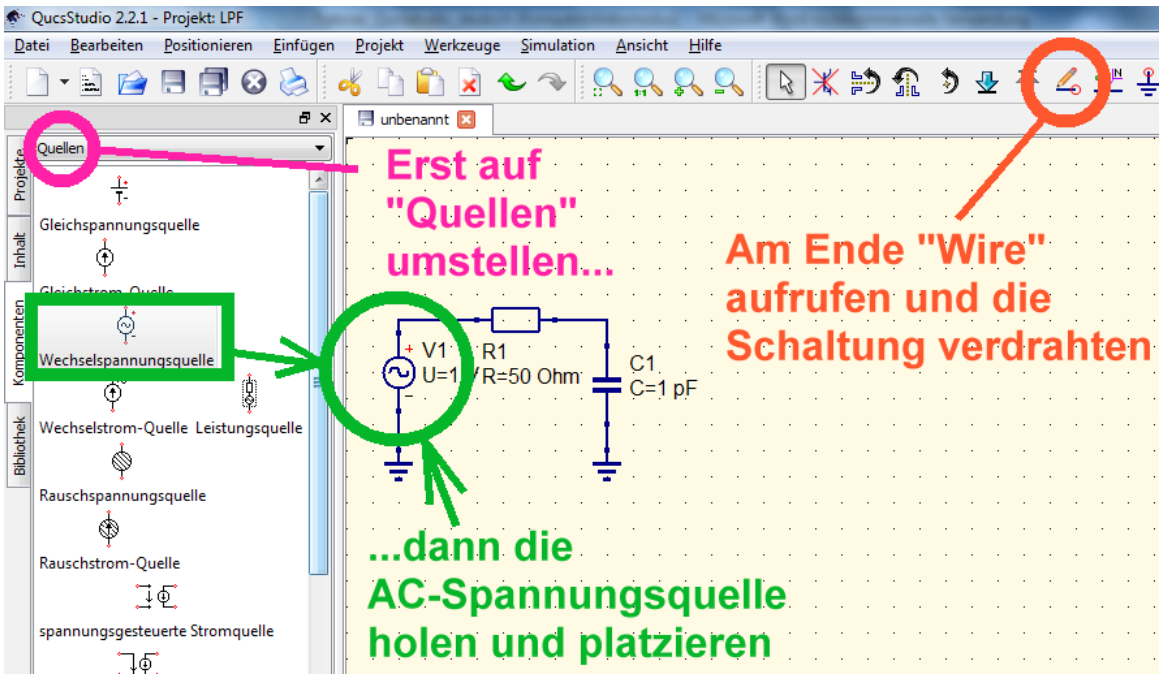
(Das ist kein Problem, denn beim Aufruf von „Datei speichern“ im Windows – Menü wird sofort der richtige Pfad ins Projekt und die richtige Datei-Endung angeboten).

2.Schritt:
Jetzt geht es an die Bauteile und dafür brauchen wir zuerst das Menü „**konzentrierte Komponenten**“ auf der „**Komponenten**“-Karteikarte.



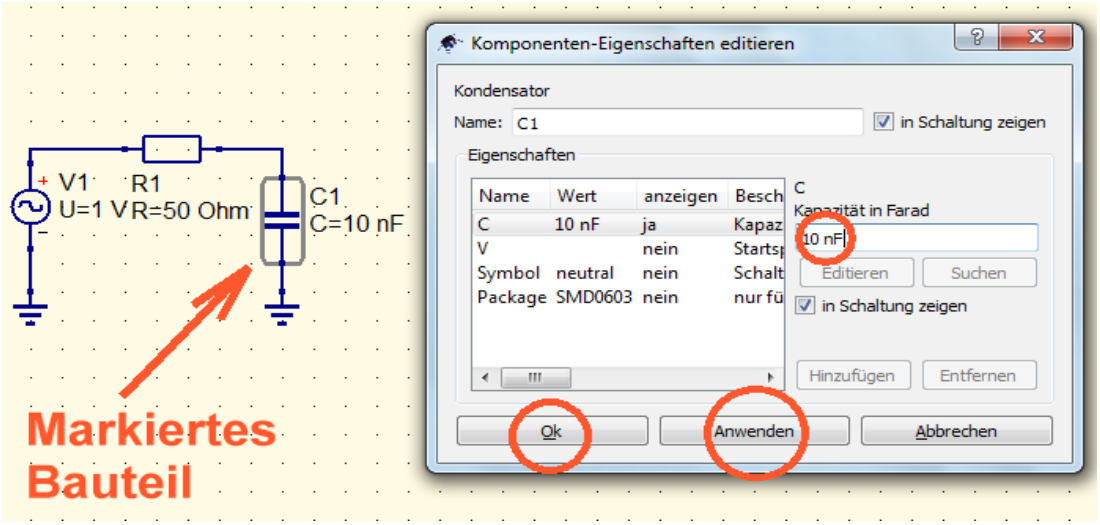
Klickt man nun auf das Widerstandssymbol im Vorrat, dann hängt dieses Bauteil am Cursor und kann mit einem linken Mausklick auf der Zeichenfläche des Editors platziert werden. Mit der „**Escape**“ – Taste befreit man sich nun wieder von diesem Anhängsel.
Ebenso verfährt man mit dem Kondensator. Dass der zunächst waagrecht im Schaltbild erscheint, stört nicht. Sobald man nämlich auf „Esc“ gedrückt hat, kann das Symbol (nach dem „Markieren“ durch Anklicken) mit dem „**Rotate – Button**“ (oder mit der Tastenkombination „**<Control + R>**“) passend zurechtgedreht werden. Den Abschluss bilden zwei **Massesymbole**. Verschieben lassen sich alle Symbol in bekannter Weise durch Anklicken mit der linken Maustaste (= „Markierung“), dadurch erscheint ein grauer Rahmen um das Symbol. Anschließend kann es durch „Ziehen bei gedrückter Maustaste“ an den richtigen Platz verschoben werden.

3. Schritt:



Nachdem das Auswahlmenü auf „Quellen“ umgestellt ist, kann eine **AC-Spannungsquelle** (= „Wechselspannungsquelle“) geholt und im Schaltbild platziert werden. Den Abschluss bildet die Verdrahtung mit einem „**Wire**“ oder „**Draht**“ durch einen Mausklick am Anfang und einem weiteren Mausklick nach dem Verlegen einer Verbindung zwischen zwei Schaltungspunkten.

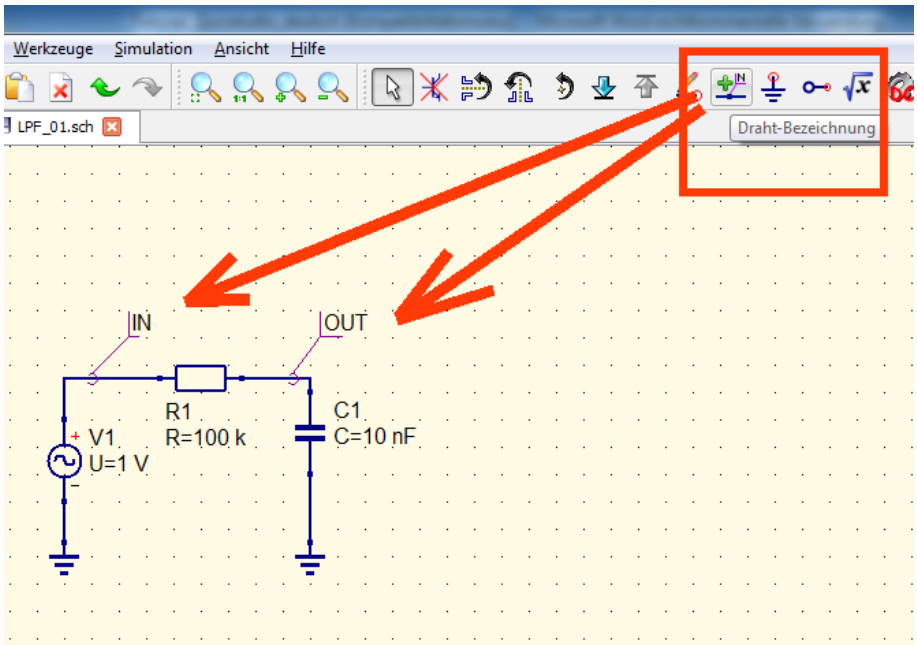
4. Schritt:
Am Kondensator wollen wir uns nun ansehen, wie ein Bauteilwert geändert wird.
Dazu wird erst mit der linken Maustaste zweimal schnell nacheinander auf das Schaltzeichen geklickt -- oder erst mit der rechten Maustaste auf das Symbol geklickt und dann „Eigenschaften editieren“ gewählt. So wird das „**Property Menu**“ geöffnet. Dort wird im entsprechenden Fenster der Wert von 10 nF eingetragen. Anschließend drückt man „**Anwenden**“, gefolgt von OK. Durch einen linken Mausklick irgendwo auf die leere Zeichenfläche verschwindet die Markierung des Bauteils wieder



Genau so verfährt man mit dem Wider-stand. Er erhält einen Wert von **100 kΩ**.

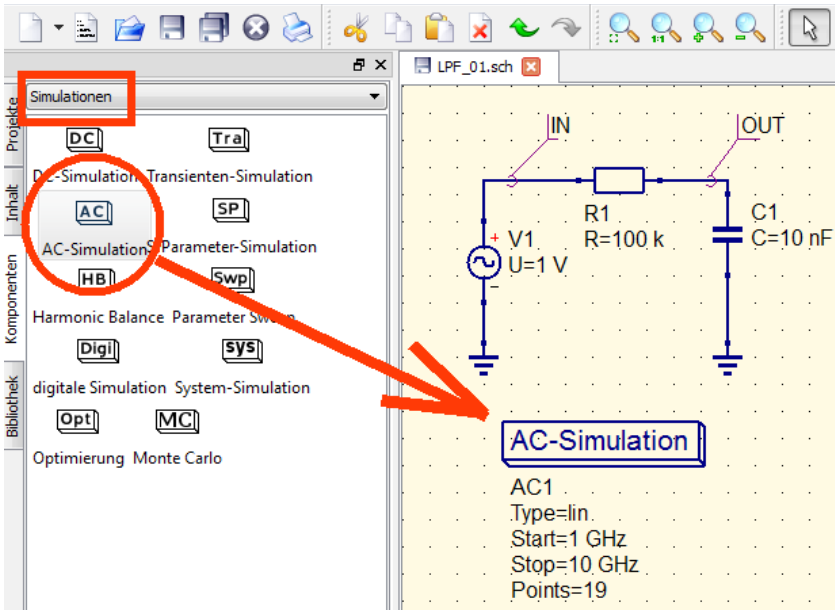
5. Schritt
Bei der **Spannungsquelle** prüfen wir, ob neben dem Schaltzeichen bereits der erforderliche Eintrag **von „U=1V“** zu finden ist. Wenn nicht, dann müssen wir eben auch hier „Eigenschaften editieren“ – öffnen, die gerade beschriebene Prozedur vornehmen und für diese Vorgabe sorgen.

Jetzt fehlen noch „**Labels**“, also passende verständliche Bezeichnungen für wichtige Punkte in der Schaltung. **Ohne diese Labels können keine Ergebnisse angezeigt werden.....hier sind es der Eingang (= IN) und der Ausgang (= OUT).**



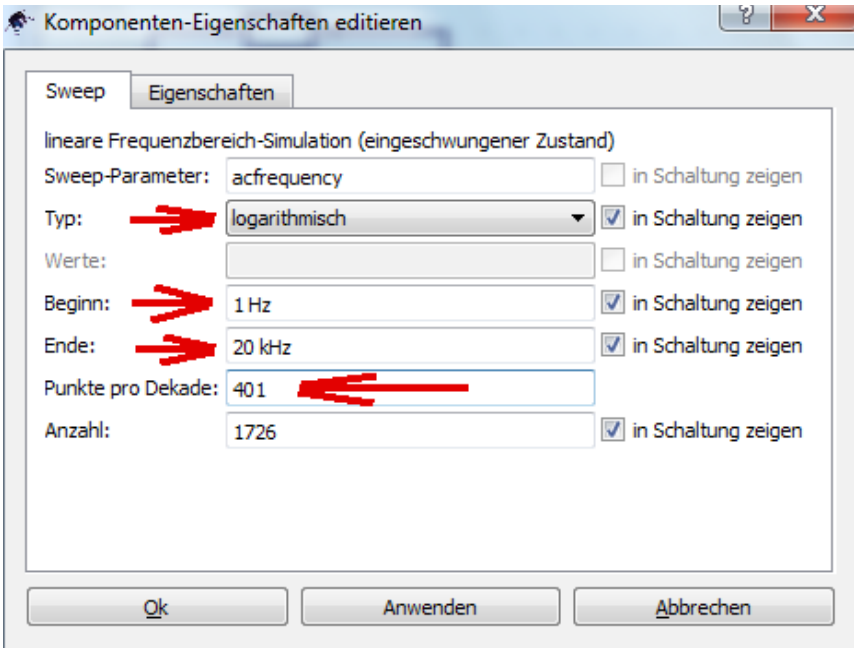
Dazu gibt es einen hübschen Button **“Draht-Bezeichung“** in der Menüleiste (…andere Programme sagen einfach „Label“ dazu…) und wenn man den anklickt, hängt dieses Element am Cursor.

Sobald man nun auf den gewünschten Knoten klickt, muss man anschließend nur noch den passenden Namen im auftauchenden Fenster eintippen. Dann wiederholt man den Vorgang beim nächsten Knoten und verlässt die Option wieder mit „Escape“.



6. Schritt:
Nach einem Menüwechsel auf „Simulationen“ wählen wir die „**AC-Simulation**“, ziehen sie mit der Maus in die Zeichenfläche und setzen sie dort in bekannter Weise ab.

Durch den berühmten Doppelklick auf dieses Symbol „AC-Simulation“ kommen wir an das Eigenschaften-Menü für die **Sweep – Vorgaben** heran.



Bitte eintragen:

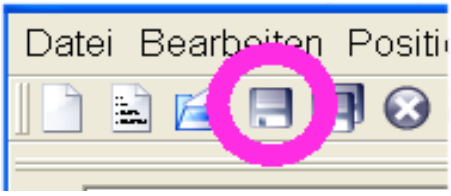
Logarithmisch

Start: 1 Hz

Ende: 20 kHz

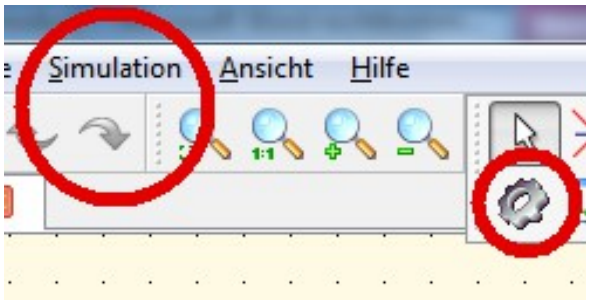
401 Punkte pro Dekade

(Die Punktezahl wird dann automatisch berechnet)



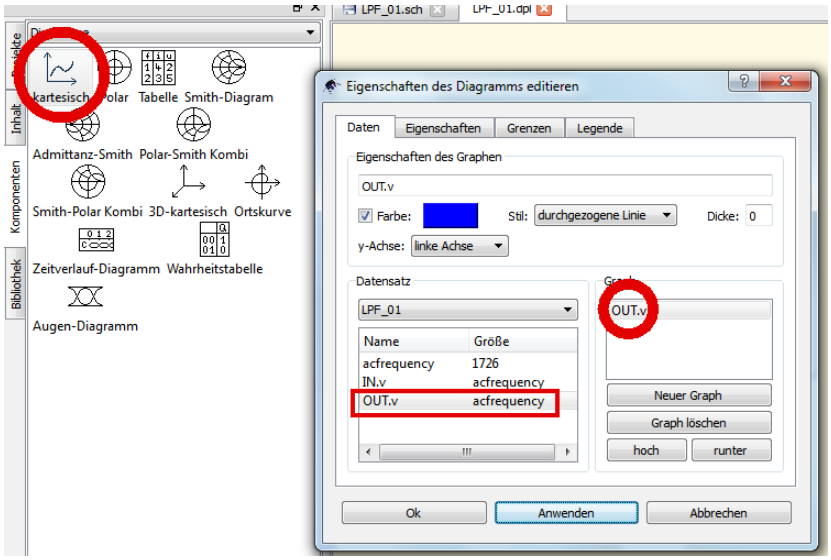
7. Schritt:
Spätestens jetzt muss der Schaltplan mit allen Vorgaben abgespeichert werden.
Bitte dazu den markierten Button drücken, im anschließend auftauchenden Menü z. B. die Schaltplanbezeichnung
LPF_01
eintippen und mit OK abschließen. Pfad und Datei-Endung werden automatisch richtig eingestellt.
War das Schaltbild aber schon vorher angelegt, dann werden durch den Klick auf das markierte Symbol die neuesten und letzten Änderungen gespeichert.

8. Schritt:



Nun wird simuliert!
Dazu sucht man **entweder** nach dem Button mit dem Zahnrädchen **ODER** man öffnet das Menü „Simulation“ und wählt die passende Zeile aus.

Wenn alles stimmt, huschen kurz ein Bildchen und ein grüner Balken über den Bildschirm.
Dann wird automatisch auf den Ergebnis-Bildschirm umgeschaltet.

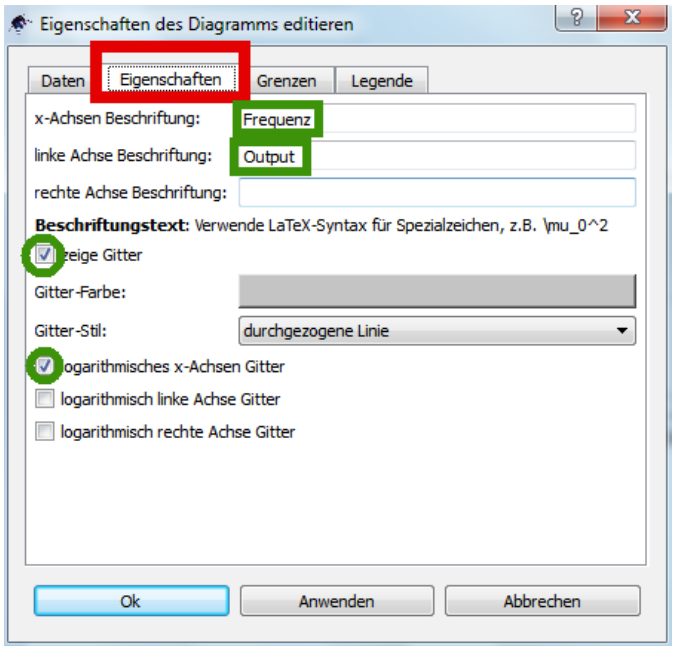


9. Schritt
Wir drücken mit der linken Maustaste auf „**kartesisch“** und ziehen dieses Diagramm auf die Arbeitsfläche. Beim Loslassen der Maustaste öffnet sich sofort das Property-Menü. Darin wird auf

„OUT.v“

geklickt und anschließend kontrolliert, ob diese Auswahl korrekt in das Graph-Fenster übernommen wurde.

Wer möchte, der kann (im oberen Drittel des Menüs) gleich die **Liniendicke** umstellen (… „2“ hat sich da sehr bewährt). Auch die **Farbe** kann man ändern, aber das ist etwas trickreich: **erst das Häkchen bei „Farbe“ entfernen -- dadurch wird das Farbfeld schwarz. Dann das Häkchen wieder setzen und anschließend in das schwarze Farbfeld klicken: damit öffnet sich die Auswahl-Farbpalette**.



Dann wechseln wir auf die Karteikarte „**Eigenschaften**“ und geben folgende Feinheiten vor:

x-Achsen Beschriftung: **„Frequenz“**

linke Achse Beschriftung: **„Output“**

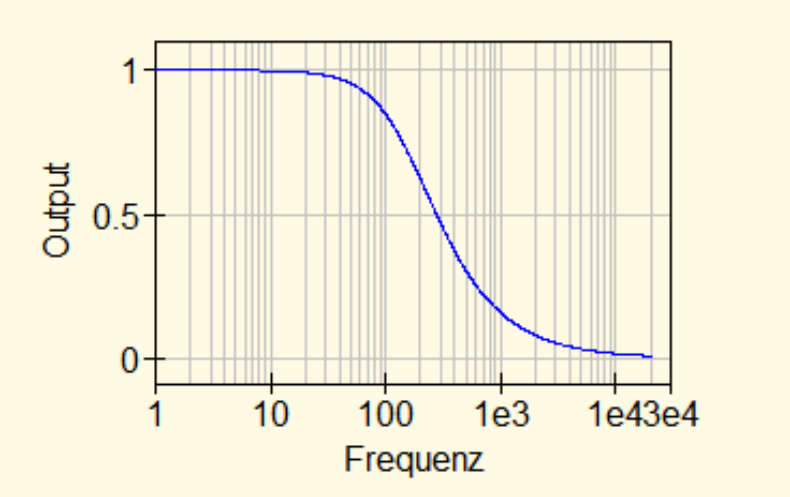
Dazu setzen wir Häkchen bei

„zeige Gitter“

und

logarithmisches x-Achsen-Gitter

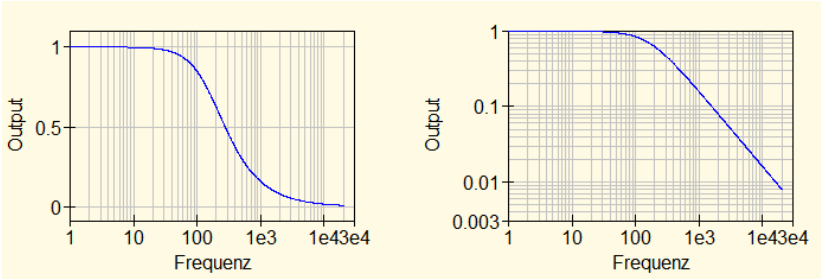
Die Spannung steigt: was passiert nun wohl nach „OK“?



Das wollten wir doch wissen!

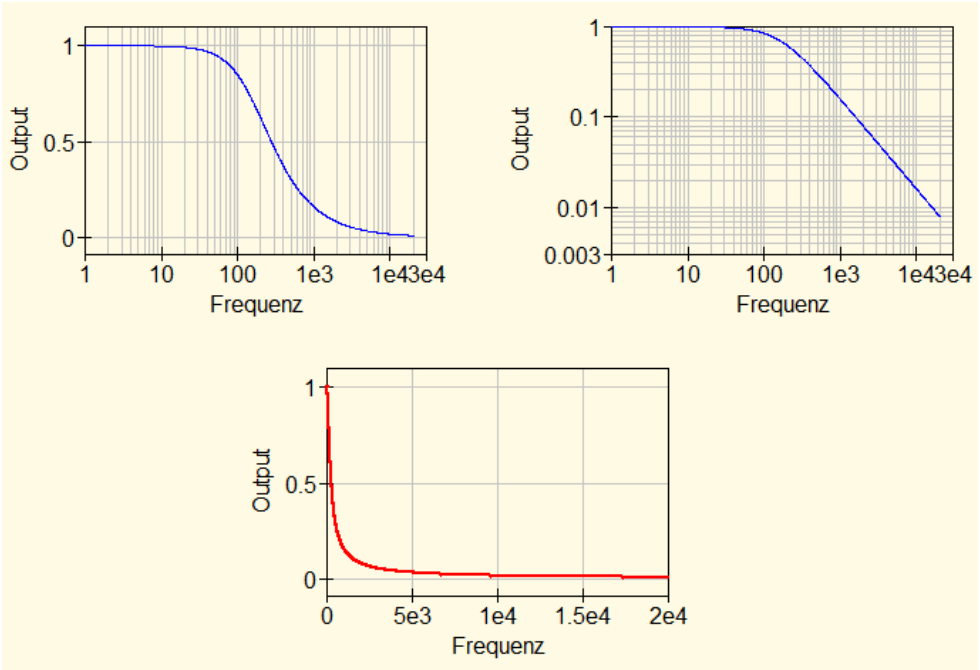
Und da wir bei der Eingangsspannung eine Amplitude von 1V verwendet haben, stellt diese Kurve gleichzeitig den Verlauf der Übertragungsfunktion (OUT / IN) dar!

Wiederholen wir diese Prozedur und holen uns (erneut durch einen Mausklick auf „Kartesisch“ im Diagramm – Menü) ein weiteres Diagramm auf unseren Schirm. Diesmal sollen **beide Achsen logarithmisch geteilt** sein. Also füllen wir die „Daten“-Karte wie vorher aus, setzen aber auf der Karteikarte „**Eigenschaften**“ zusätzlich ein Häkchen bei
„logarithmische linke Achse Gitter.



So sollte die Zeichenfläche auf dem Bildschirm nun aussehen:

Zum Abschluss des Kapitels wollen wir noch **beide Achsen mit linearer Teilung** versehen. Auch das dürfte kein Problem sein, denn da muss man nur die beiden Häkchen für die logarithmischen Teilungen entfernen. Aber stellen wir doch zusätzlich auf der ersten Karteikarte „**Daten**“ die Ergebniskurve auf „**rote Farbe**“ und „**Liniendicke = 2“** um:



Na also…

3.1.2. Was steckt hinter „dB“?

Die „dBs“ sind sozusagen das handwerkszeug der Nachrichtentechnik und der Hintergrund ist nicht kompliziert:
Sehr oft hat man es mit Spannungsverhältnissen bzw. Leistungsverhältnissen wie Verstärkung oder Dämpfung zu tun.Schaltet man nun mehrere Bausteine hintereinander, dann muss man zur Ermittlung des Gesamtverhaltens z. B. alle Verstärkungsfaktoren miteinander multiplizieren. Das wird kompliziert und man hat sich da eine große Hilfe ausgedacht:

Geht man auf logarithmische Darstellung über, dann wird die Multiplikation durch eine Addition von dB-Werten ersetzt (...die man oft im Kopf erledigen kann....).

Dann gilt

Pout / Pin in dB = 10 * log(Pout / Pin)

Und da sich die Spannnungen quadratisch mit der Leistung ändern,gilt

Uout / Uin in dB = 2o * log(Uout / Uin)

Übrigens:
„dB“ ist die Abkürzung für „Dezi-Bel“, also 0,1 Bel. Damit erreicht man in den meisten Fällen, dass der anzugebende dB-Wert größer als 1 wird.

Die folgende Tabelle sollte man aber irgendwann im Kopf haben:

dB-wert	Spannungsverhältnis	Leistungsverhältnis
0	1	1
6	2	4
9,5	3	9
12 = 6+6	4 = 2*2	16
14 = 20 -6	5 = 10 / 2	25
15,5 = 9,5 + 6	6 = 2*3	36
16,9	7	49
18 = 6 +6 +6	8 = 2*2*2	64
19 = 9,4 + 9,5	9= 3*3	81
20	10	100
26 = 20 + 6	20 = 2*10	400
32 = 20 + 12	40 = 4*10	1600
40 = 20 + 20	100 = 10*10	10 000
60	1000 =10*10*10	1 000 000

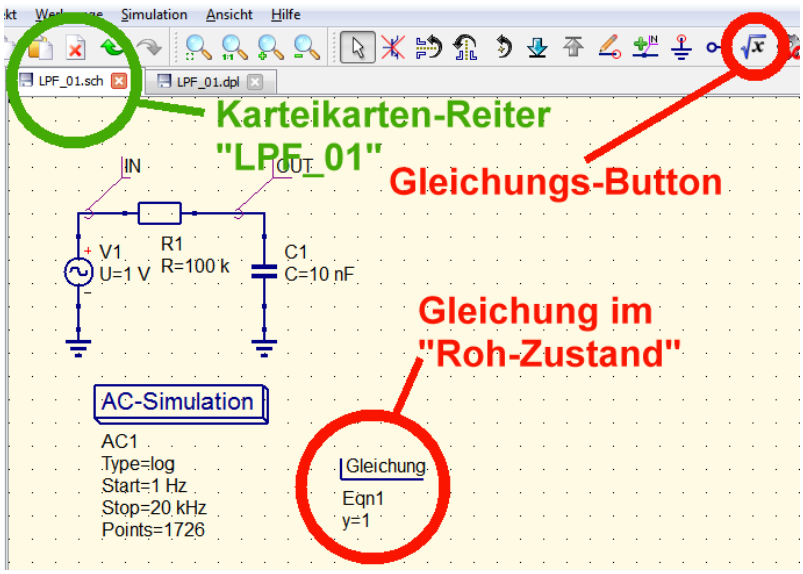
usw.

3.1.3. „dB“ Darstellung und Phasendarstellung beim Ergebnis

Da gibt es grundsätzlich keinen eigenen Button,aber zwei Möglichkeiten.

Möglichkeit 1:

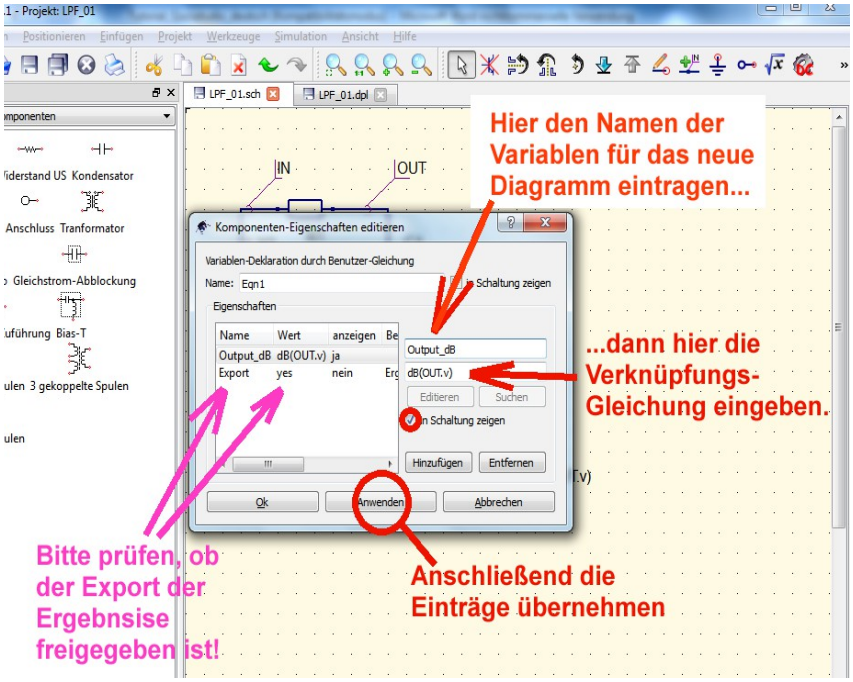
Hierfür müssen wir zum ersten Mal bereits vor der Simulation eine Gleichung bereitstellen, damit dieses Ergebnis gleich nach der Simulation für die Anzeige zur Verfügung steht!



Also kehren wir zu unserem Schaltbild durch Anklicken des zugehörigen

„**Karteikarten-Reiters LPF_01.sch**“

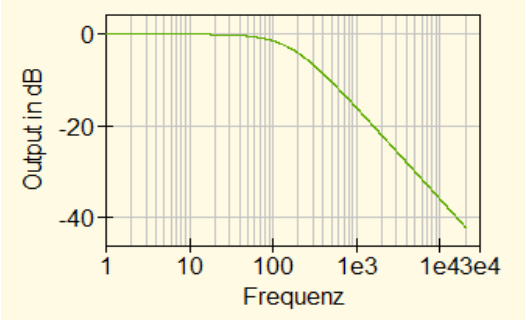
zurück und drücken auf den **Gleichungs-Button** in der Menüleiste. Dann hängt die „**Gleichung im Rohzustand**“ am Cursor und kann im Schaltbild abgesetzt werden.
Auf das Wort „Gleichung“ wird nun **doppelt geklickt**, um die erforderlichen Einträge im Menü vorzunehmen.



Wie man sieht, wurde hier als Name für die dB-Darstellung „**Output_dB**“ gewählt.
Die dB-Berechnung steht als Funktion zur Verfügung und deshalb wird durch den Eintrag „**dB(OUT.v)**“ in der zweiten Zeile die Gleichung

Output_dB=dB(OUT.v)

programmiert (…wobei “**OUT.v**” den Spannungsverlauf am Ausgang bildet).



Nach einer erneuten Simulation erhalten wir dieses Ergebnis, wenn wir ein neues Kartesisches Diagramm aufrufen und dann im Menu „**Output_dB**“ auswählen. **Außerdem wurde eine logarithmisch geteilte Frequenzachse sowie grüne Farbe für die Ergebnislinie eingestellt.**

Aufgabe:
Sehen Sie bei der Y-Achse eine 10dB-Teilung vor.

Möglichkeit 2:

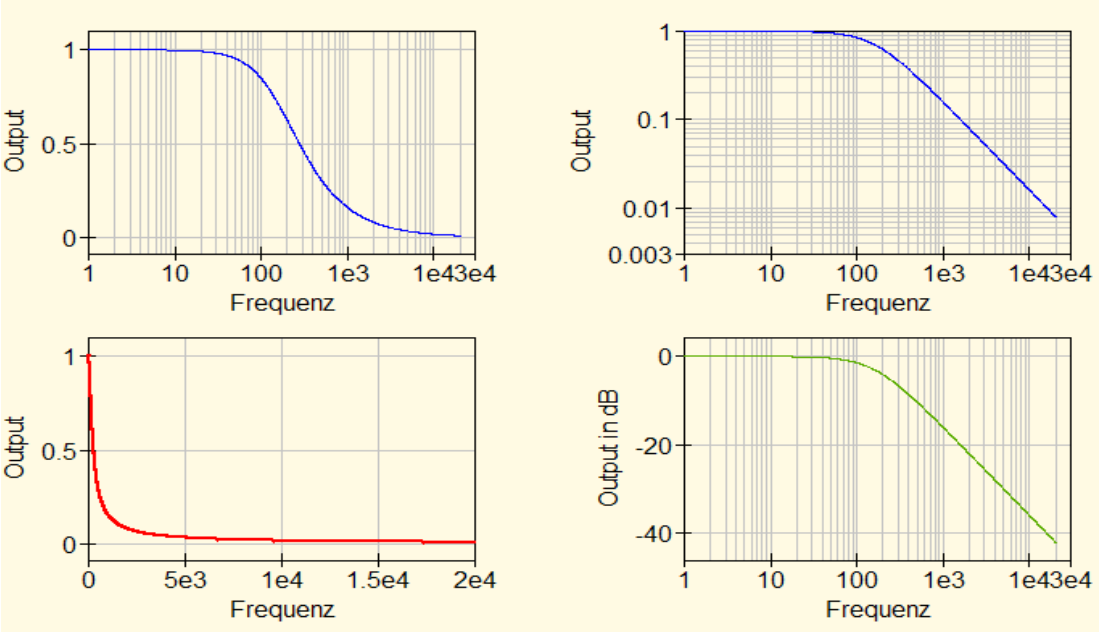
Hat man dagegen bereits etwas fertig simuliert und dann fällt es Einem ein, schnell das Ergebnis auch in „dB“ anzusehen – das verwendet man dieses Verfahren:

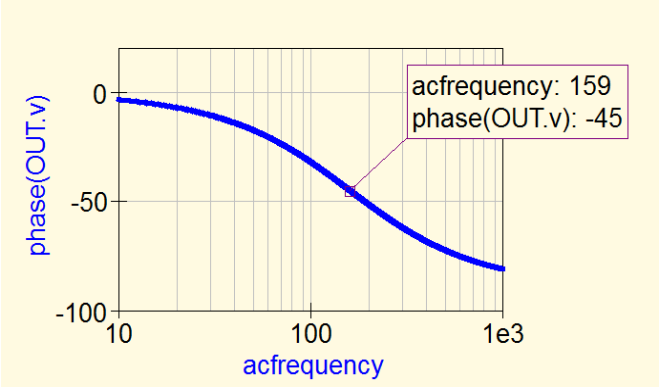
1. Schritt:
Wir öffnen nach der Simulation ein kartesisches Diagramm und seine Eigenschaften. Das kann ein bereits verwendetes oder ein neues sein. Darin wird **zum Beispiel die Ausgangsspannung „OUT.Vt“** ausgewählt und markiert.

2. Schritt:
Sobald wir nun im Dataset (= linkes unteres Fenster) auf „OUT.Vt“ klicken, erscheint diese Variable zwei Mal: einmal rechts **in der Mitte unter „Graph“** und **ganz oben in der Zeile „Graph Properties“.**
In **diese obere Zeile** (Graph Properties) stellt man ganz links den Cursor und ändert den Eintrag in

dB(OUT.Vt)

Mit „Anwenden“, „OK“ und einer logarithmisch geteilten Frequenzachse bekommen wir dann dasselbe Ergebnis wie oben und haben damit insgesamt folgende Sammlung beieinander:





Wer sich (z. B. Als Regelungstechniker) auch für den **Phasenverlauf über der Frequenz** interessiert, der öffne ein neues kartesisches Diagramm, wähle im Property-Menü die Ausgangsspannung (z. B. "Out") für die Anzeige und trage in der Zeile "**Graph Properties**" folgenden Wunsch ein:

"phase(Out.v)"

Er wird nach einem Klick auf "Anwenden" und "OK" sofort erfüllt.
Und wenn man zusätzlich einen Marker setzt (= zweiter Button von rechts in der Menüleiste), kann man ihn nach dem Markieren mit den Pfeiltasten genau auf die Grenzfrequenz schieben

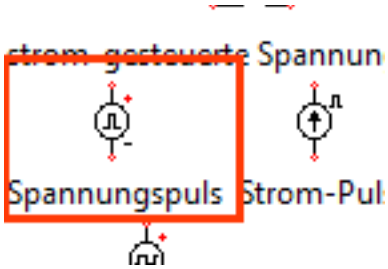
3.2. Simulationen in der Time Domain

3.2.1. Impulsantwort des RC-Tiefpasses

Eine wichtige Sache, die immer mehr an Bedeutung gewonnen hat! Denn aus der **Impulsantwort** kann mittels der FFT (Fast Fourier Transformation) sehr schnell das Verhalten im Frequenzbereich (= **Frequenzgang der Übertragungsfunktion**) bestimmt werden.
Man speist dazu die Schaltung mit einem „Dirac"-Impuls, der die Fläche „1" besitzt. Dabei ist die Pulslänge theoretisch unendlich kurz, die Amplitude geht dafür gegen Unendlich. Das ist für die Praxis fast eine Schnapsidee, aber man kann sich da mit einer einfachen Näherung helfen:

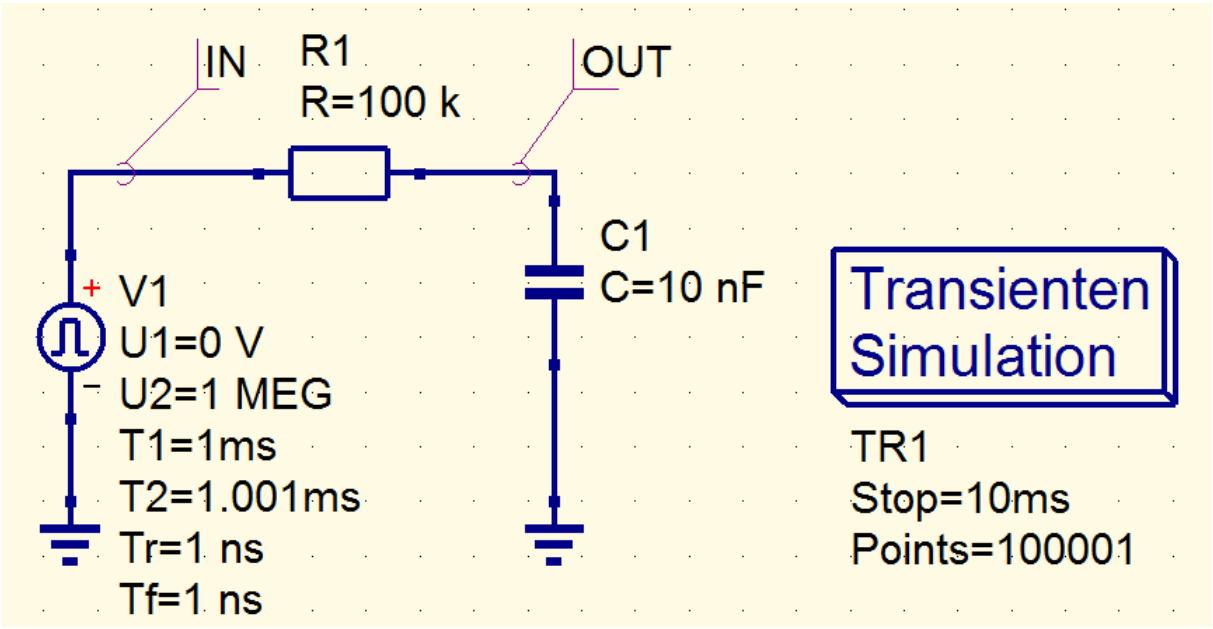
Macht man die Pulslänge wenigstens um den Faktor 100….1000 kleiner als die Zeitkonstante des untersuchten Systems und erhöht dafür um denselben Faktor die Amplitude, dann bleibt die Fläche gleich. Aber man erhält dieselben Ergebnisse wie bei der Speisung mit dem echten Dirac-Impuls.

Nehmen wir unseren Tiefpass aus dem vorigen Kapitel nochmals her und legen ein zusätzliches neues Projekt „**LPF_TD**" an. Darin kann man in dem neuen geöffneten Schaltplan entweder die Schaltung nochmals zeichnen oder sie aus dem vorigen Kapitel in die Zwischenablage kopieren und dann mit <CTRL + v> einfügen.

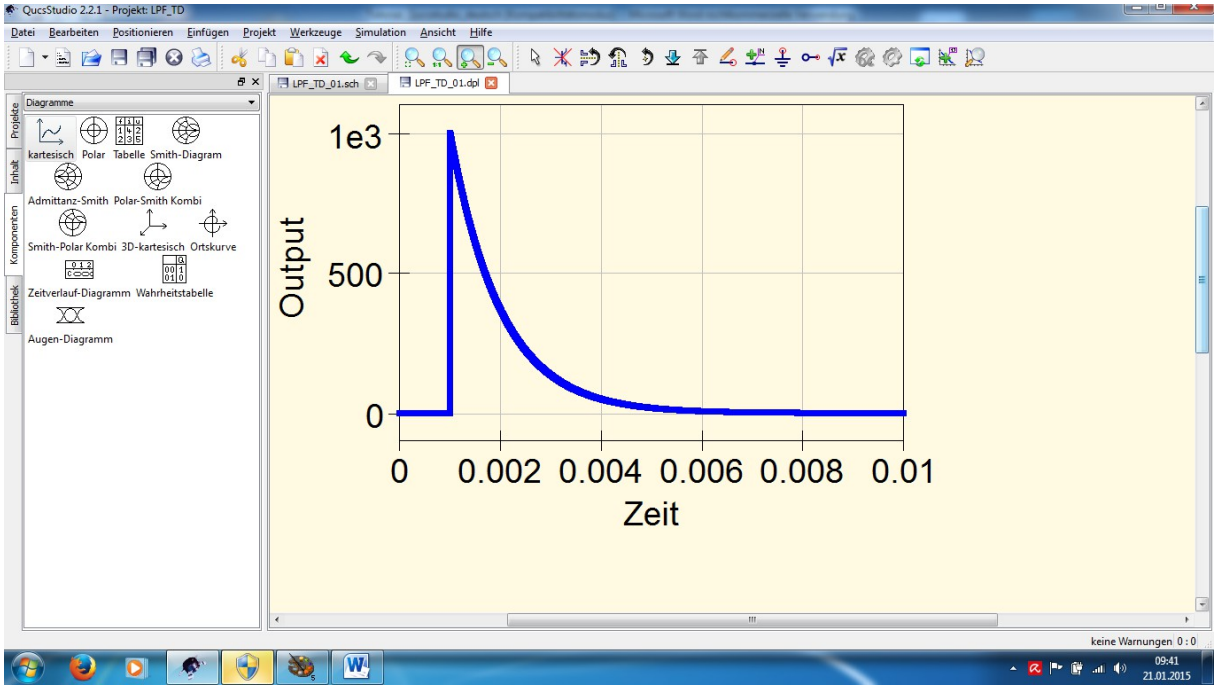


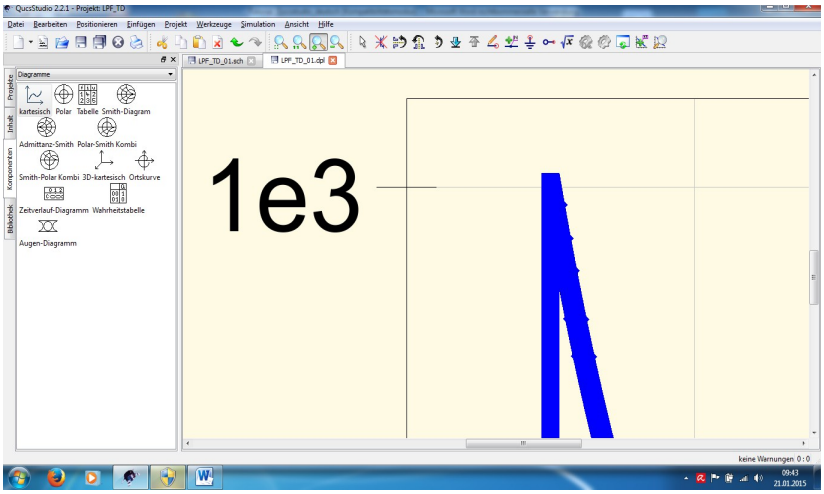
Wir brauchen nun eine **Puls-Spannungsquelle** und die finden wir in der Kartei „**Komponenten / Quellen**". Wir arbeiten mit einer **Pulsamplitude von 1 MegaVolt und einer Pulslänge von 1 Mikrosekunde. Die Startverzögerung sei 1 Millisekunde, Anstiegs- und Abfallzeit je 1 Nanosekunde**.

Dann holen wir uns aus der Kartei „**Simulationen**" die „**Transienten-Simulation**" und setzen sie im Schaltbild ab. **Eingestellt wird eine Simulationszeit von 10 Millisekunden mit 100001 Punkten.**
So muss der Bildschirm dann aussehen:



Dann wird simuliert und das kartesische Diagramm gewählt. Ein rechter .Mausklick auf das Diagramm, gefolgt von „**Eigenschaften editieren**“, ermöglicht die Übernahme von „**OUT.Vt**“ in die Graphenliste, eine passende Achsenbeschriftung und eine größere Linienbreite. Mit etwas Zoom sieht man dieses Ergebnis:





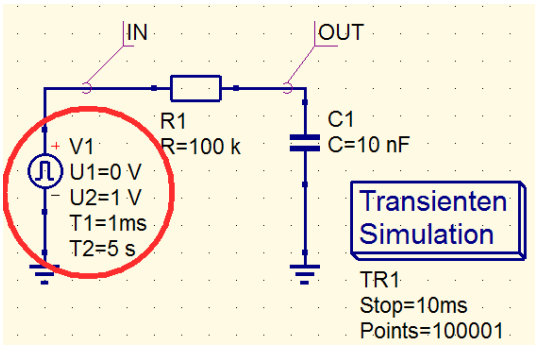
Selbstverständlich ist es nun möglich, jedes beliebige Detail des Kurvenverlaufs über den Zoom heraus zu holen.

Das zeigt dieses Beispiel.

3.2.2. Sprungantwort des Tiefpasses

Da haben wir zwar mehrere Möglichkeiten. Fangen wir mit der einfacheren Methode an.

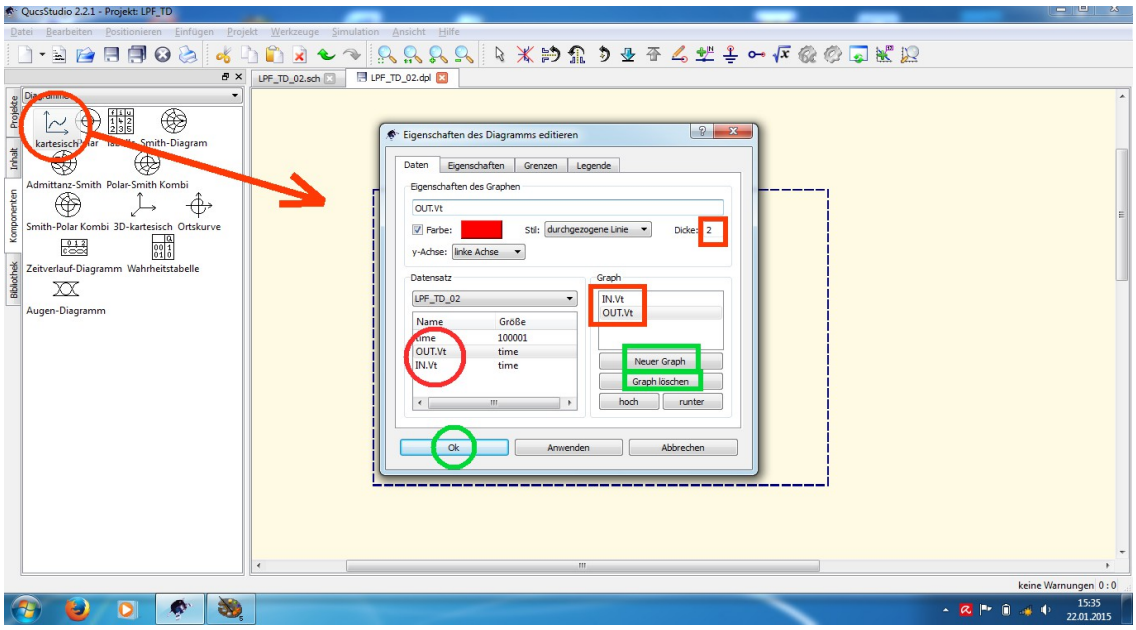
3.2.2.1.Lösung mit Puls-Quelle



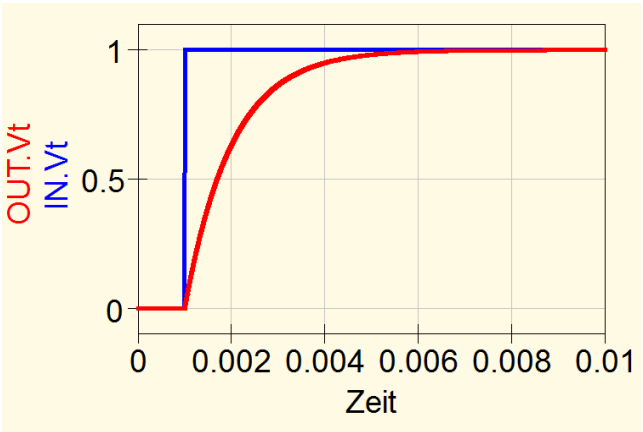
Wir benützen die vorige Schaltung und ändern nur die **Programmierung der Pulsquelle**. Nach einer Zeitverzögerung von 1 Millisekunde startet der Puls (mit einer Amplitude von 1V) und dauert **5 Sekunden (= also viel länger als die Simulationszeit von 10 Millisekunden).** Die Anstiegs- und Abfallzeit bleibt bei der vorgegebenen Einstellung von 1 ns.

Wenn wir nun mit einer Schrittweite von 100 Nanosekunden für 10 ms simulieren, entspricht das Ergebnis genau der gesuchten Sprungantwort.

Nach der Simulation wählen wir wieder das kartesische Diagramm und füllen das auftauchende Menü entsprechend dem folgenden Bild aus.



Im linken Datensatz klicken wir zuerst auf „**IN.vt**“, dann sollte das in die rechte Graph-Liste übernommen werden. Oben rechts wählen wir die Linienbreite zu „2“ und klicken anschließend auf „**Neuer Graph“**. Da taucht leider plötzlich in der Graph-Liste nochmals die vorige „IN.vt“ auf, **aber die beseitigen wir sofort mit „Graph löschen**“. Dann wiederholt sich das Spiel:



Nun wird aus dem linken Datensatz „**OUT.Vt**“ durch Anklicken in die Graph-Liste übernommen und geprüft, ob die Linienbreite von „2“ auch für diese Kurve übernommen wurde.
Wenn ja, dann ruft man die Karteikarte „Eigenschaften“ auf und trägt „Zeit“ als Bezeichnung der X-Achse ein. Mit „Anwenden“ und OK sollte man dann dieses Bild sehen.

3.2.2.2. Lösung über ein „PWL“ (= piece wise linear) - Signal

Diese Möglichkeit, sich einen bestimmten Spannungsverlauf durch eine Datei mit vielen Wertepaaren (= Zeitpunkt + Amplitude bei einem bestimmten Zeitpunkt) zu erzeugen, gibt es in allen Simulationsprogrammen und das ist eine überaus praktische Sache. Dabei werden bei der Simulation die einzelnen „Samples“ durch gerade Linien miteinander verbunden und der gewünschte Verlauf angenähert.

Allerdings muss man beim qucsstudio auf einige Feinheiten achten:

a) Der Aufruf dieser Quelle erfolgt wieder über „Komponenten / Quellen“. Aber die PWL-Quelle sucht man vergeblich, denn hier heißt sie „**dateibasierte Spannungsquelle“.**

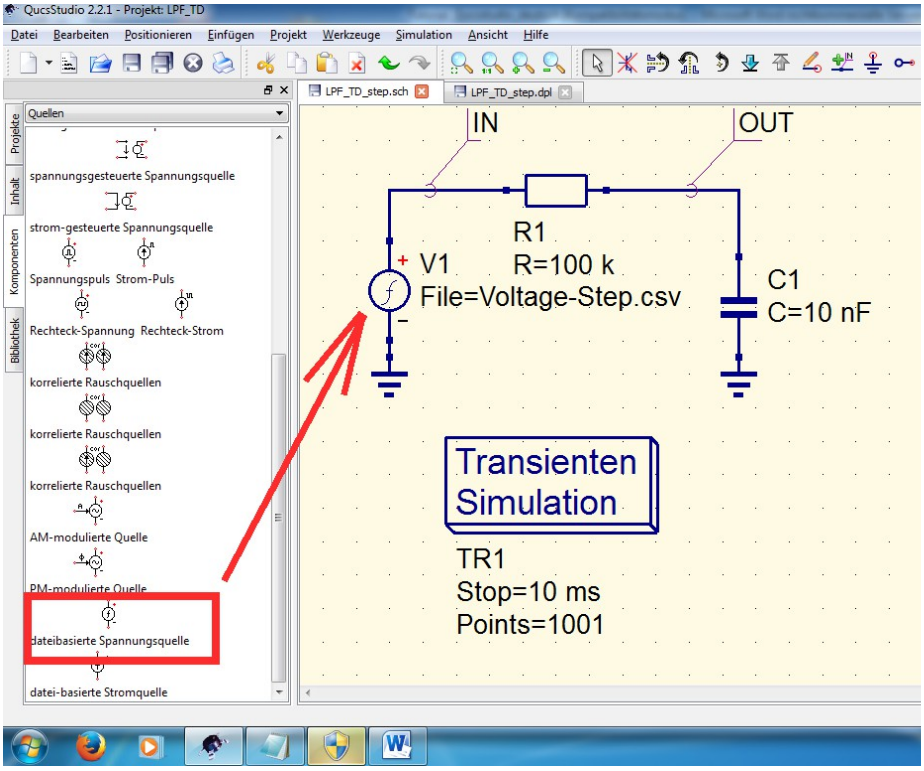
b) Der Spannungsverlauf muss in einer **selbst geschriebenen** „*.csv“-Datei“ als eine Folge von Wertpaaren angegeben werden. Die Datei selbst wird dann in das betroffene Projekt hineinkopiert. (Übrigens: „csv“ ist die Abkürzung für „comma separated values“…)

Fangen wir mit der Datei „**Voltage-Step.csv**“ an.
Begonnen wird beim Nullpunkt mit Null Volt. Bei t = 1 Millsekunde haben wir immer noch Null Volt, aber 1 Mikrosekunde später bereits 1 Volt. Das bleibt so bis zum Ende der Simulationszeit von 10 ms.

Vorsicht:
Jedes Wertpärchen (= erst die Zeit, dann die Spannung) kommt in eine eigene Zeile, die jeweils mit der Eingabe-Aufforderung (= Return-Taste) abgeschlossen werden muss.

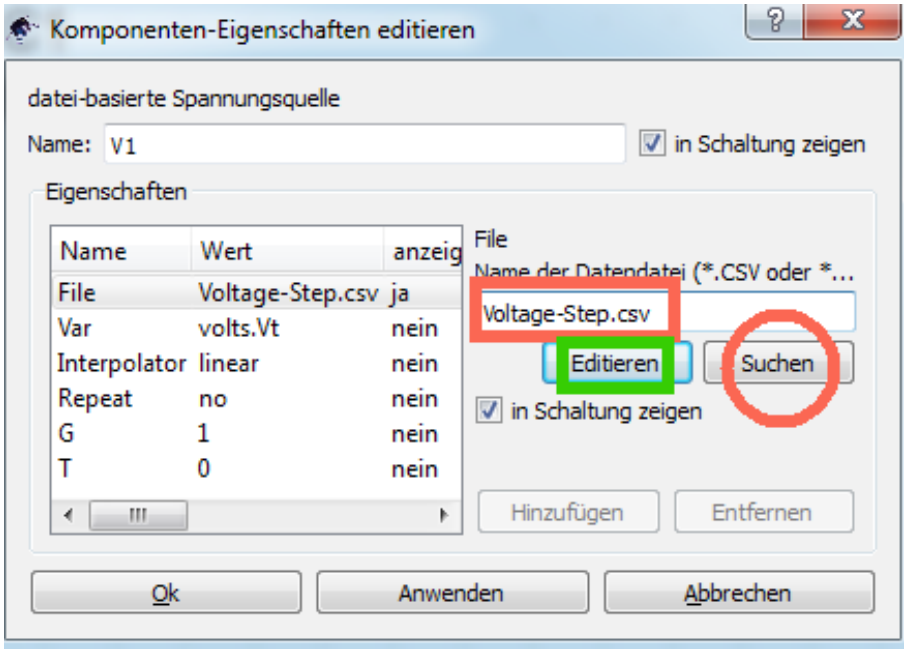
So sieht dann der Datei-Inhalt von „Voltge-Step.csv“ aus:

0	**0**
1e-3	**0**
1.001e-3	**1**
10e-3	**1**



Geschrieben wird das mit einem Texteditor und man darf nicht vergessen, vor dem Abspeichern auf „**Alle Dateien / ANSI**“ umzuschalten. Speicherort ist natürlich das Projekt, in dem die zugehörige Schaltung abgelegt ist.

Das ergibt diese Schaltung. Zum korrekten Einbinden der Wertedatei wird mit der rechten Maustaste auf das Symbol der Quelle geklickt und dann über „Eigenschaften editieren“ ihr Property-Menü geöffnet.



Zuerst stellen wir über „**Suchen**“ den Pfad zur gespeicherten „**Voltage-Step.csv**“-**Datei** her.

Ihre Bezeichnung muss dann im rot markierten Feld auftauchen.

Falls etwas an ihr zu ändern wäre, kann sie über „Editieren“ geöffnet und korrigiert werden.

Anschließend reicht „Anwenden“ und OK, damit die Simulation gestartet werden kann.

Nach der Simulation benötigen wir ein kartesisches Diagramm und füllen dessen „Eigenschaften-Menü“ folgendermaßen aus:

a) **IN.vt wird aktiviert und erscheint im Graphen-Fenster.**

b) Die Liniendicke wird auf „2“ gestellt, die Farbe ist automatisch „blau“.

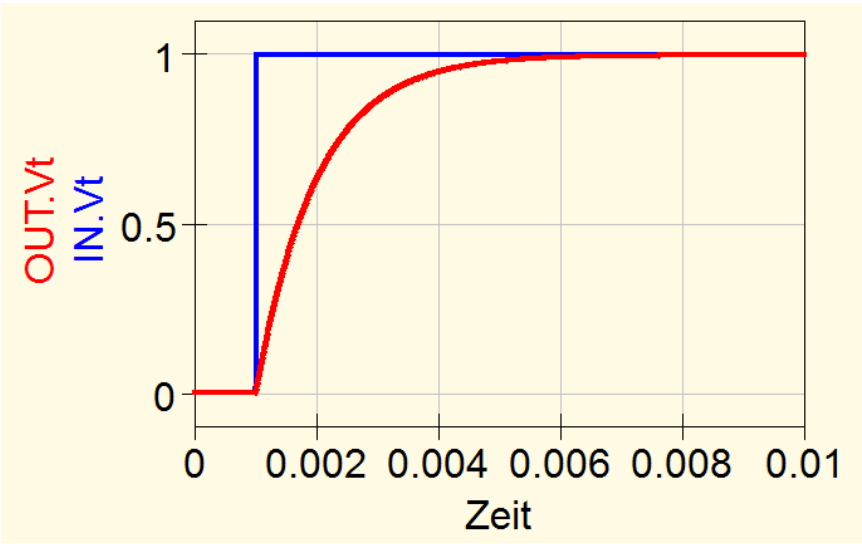
c) Dann wird „New Graph“ angeklickt und der nochmals in der zweite Zeile auftauchende Graph „IN.vt“ gelöscht.

d) **Dafür wird nun „OUT.vt“ angewählt und dadurch übernommen.**

e) Die Liniendicke für „OUT.vt“ sollte ebenfalls noch auf „2“ stehen, aber die Linie sollte eine rote Farbe aufweisen.

f) Auf der Karteikarte „Eigenschaften“ wird für die X-Achse die Bezeichnung „Zeit“ angegeben-

g) Und mit „Anwenden“ und OK sollte sich folgendes Diagramm einstellen.



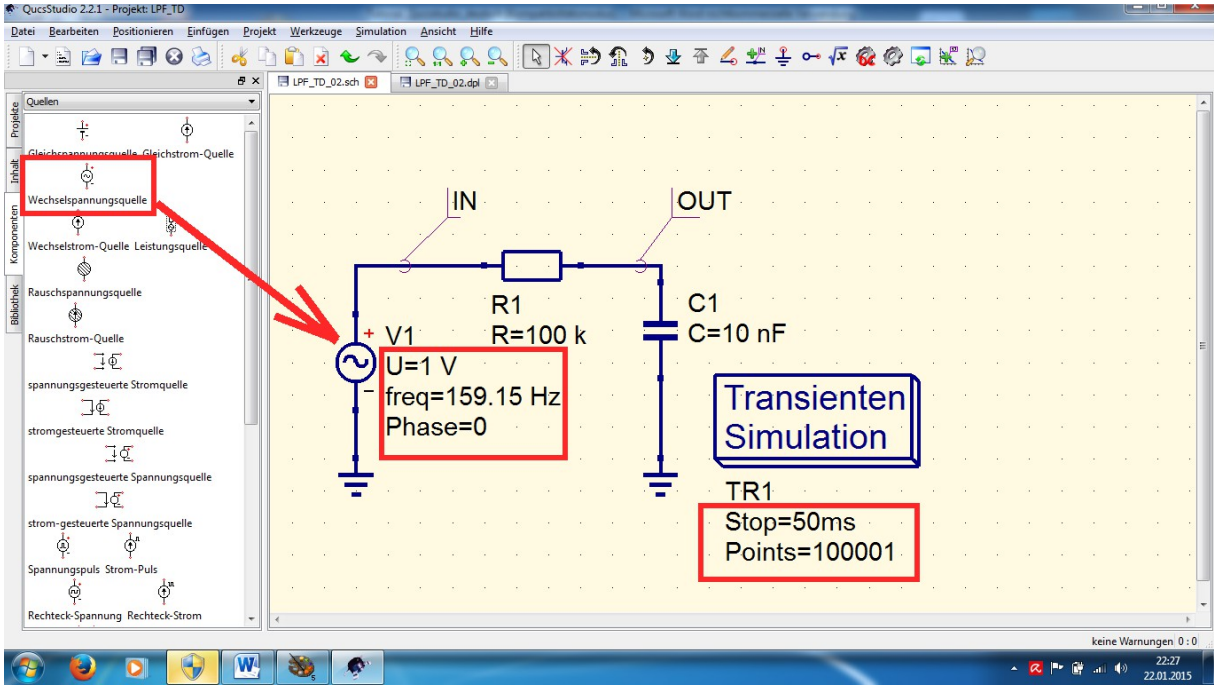
3.2.3. Periodische Signale am Eingang

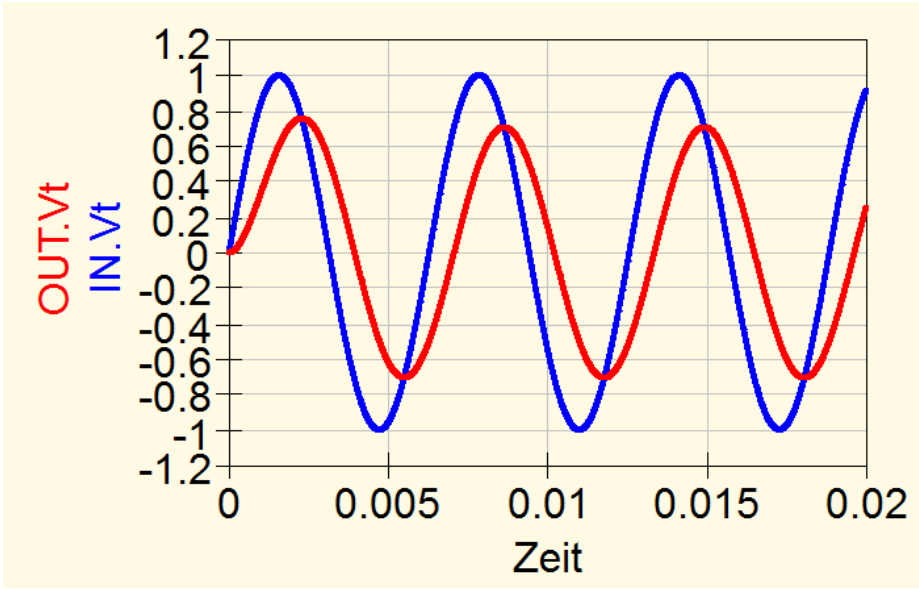
3.2.3.1. Speisung mit einer Sinusspannung

Wir nehmen exakt den Fall der Grenzfrequenz. Sie beträgt

fg = 1 / 2πRC = 1 / 2 * π * 100k * 10nF = 159,155 Hz

und wählen eine sinusförmige Eingangsspannung („Wechselspannungsquelle“) mit dieser Frequenz und einem Spitzenwert von 1V.
Damit erstellen wir folgendes Schaltbild und geben eine Simulationszeit von 0….50ms bei einem kleinsten Time Step (= Zeitschritt) von 500 Nanosekunden vor. Das ergibt 100 001 simulierte Samples.

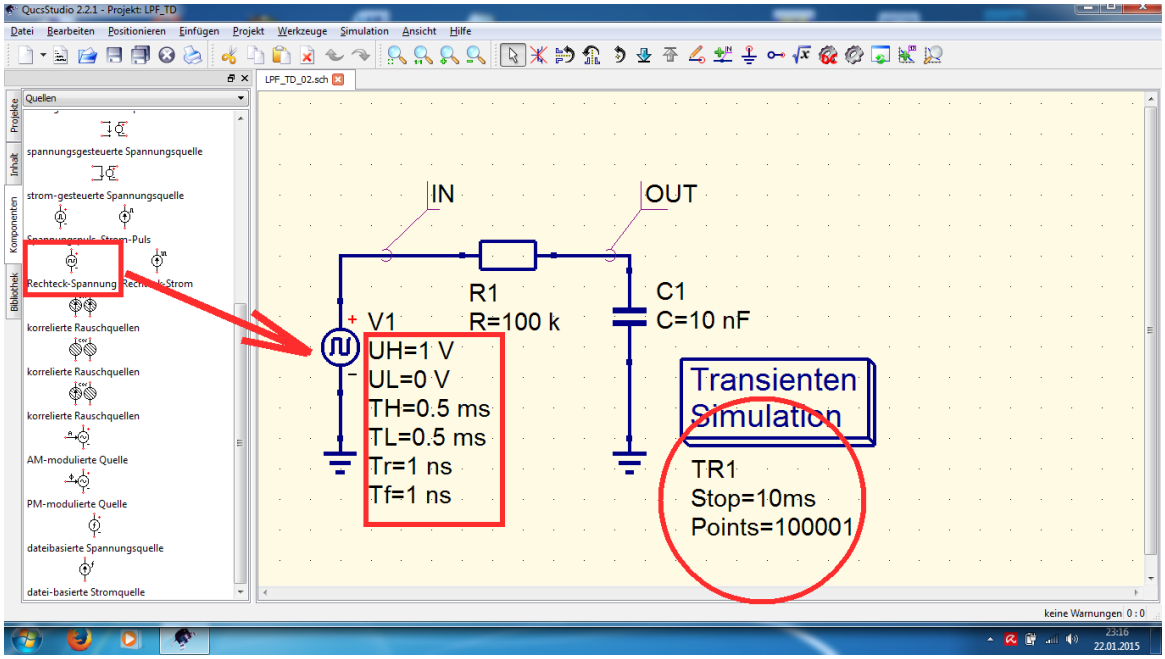




Wenn man anschließend bei der Ergebnisausgabe im kartesischen Diagramm wie in den letzten Beispielen vorgeht, dann bekommt man dieses Bild.
Allerdings erst, wenn man im Menü „Eigenschaften“ die x-Achsen-Kalibrierung auf „**manuell**“ umgeschaltet und **eine Zeit von 0…20ms sowie eine Teilung („Schritt“) von 0,005ms** eingestellt hat.
Entsprechend gilt bei der senkrechten Achse eine Schrittweite von 0,2.

3.2.3.2. Speisung mit einem Rechteck-Signal

Da arbeiten wir mit einer periodischen Rechteckspannung von **1 kHz**, um die Integrationswirkung des Tiefpasses zu zeigen. Die Simulationszeit beträgt 10 ms bei einer Schrittweite von 100 Nanosekunden.



Noch einige Worte zur Spannungsquelle:
Es wurden alle Eigenschaften des Rechtecksignals sichtbar gemacht -- ihre Bezeichnungen sind selbst erklärend.

Zum Einstellen jedes Wertes geht man so vor:

a) Durch einen doppelten linken oder mit einem rechten Mausklick auf das Symbol der Quelle ihre Property Liste öffnen.

b) Dann die gewünschte Größe durch Anklicken in der linken Menüliste markieren.

c) Im rechten Fenster den neuen gewünschten Wert eintragen.

d) „Anwenden“ drücken.

e) Das Häkchen bei „in Schaltung zeigen“ setzen.

f) Am Ende, wenn alles erledigt ist, „OK“ drücken.

Dann wird durch einen Klick auf das Zahnrädchen simuliert und wieder ein kartesisches Diagramm auf die Arbeitsfläche gezogen.

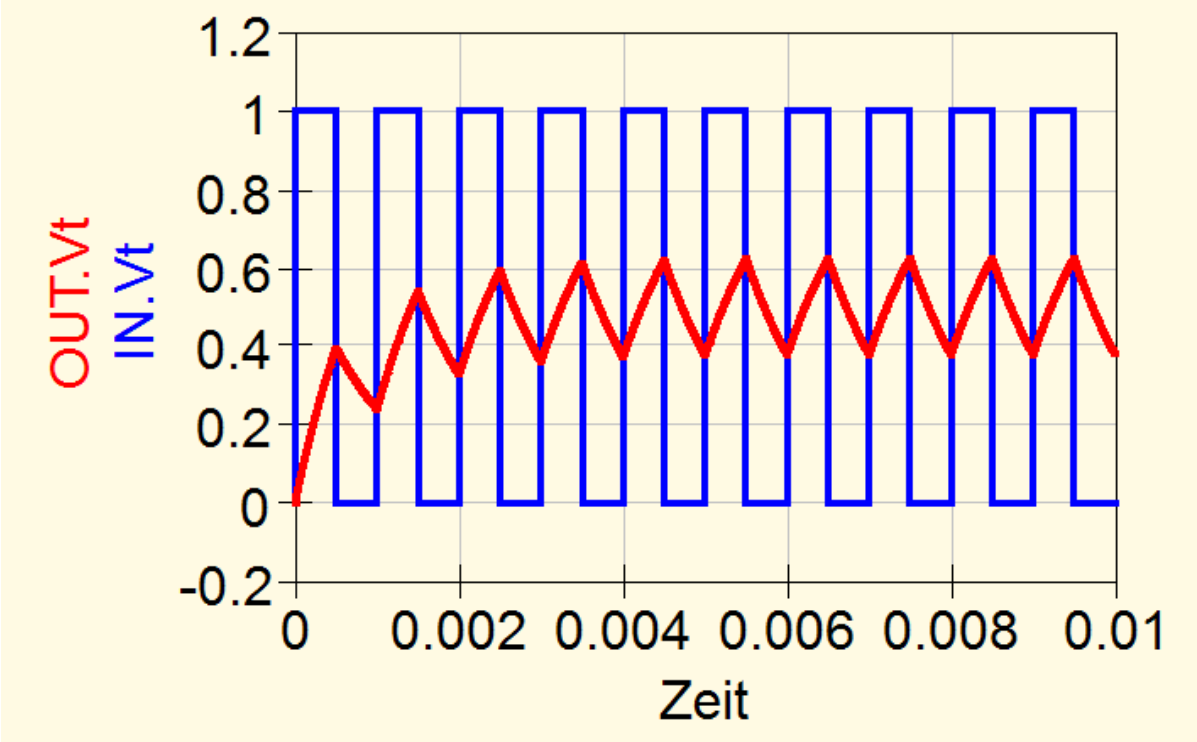
Wir sorgen erneut für eine Darstellung der beiden Spannungen in zwei unterschiedlichen Farben mit einer Liniendicke von „2“.

Auf der Karteikarte „Eigenschaften“ wird die x-Achse mit „**Zeit**“ beschriftet.

Auf der Karteikarte „Grenzen“ sorgt man dafür, dass die y-Achse von -0,2 bis +1,2 mit einer Teilung von 0,2 dargestellt wird.

Die x-Achse reicht von 0……10 ms mit einer Schrittweite von 2 ms.

So soll es aussehen:

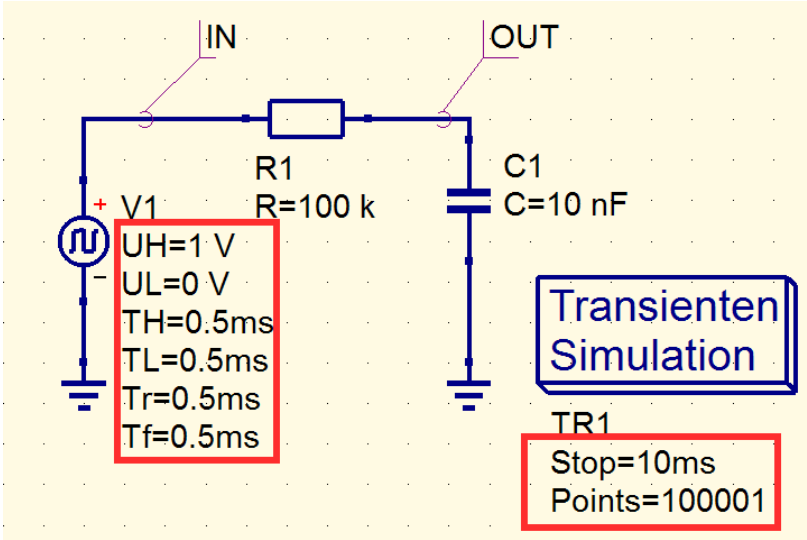


3.2.3.3. Speisung mit einem Dreieck-Signal

Das ist ganz einfach, denn wir nehmen die eben benützte Schaltung mit dem Rechtecksignal (f = 1 kHz) und verändern es folgendermaßen:

a) Mit einer langen Anstiegs- und Abfallzeit erhalten wir den gewünschten Dreiecks-Verlauf, aber

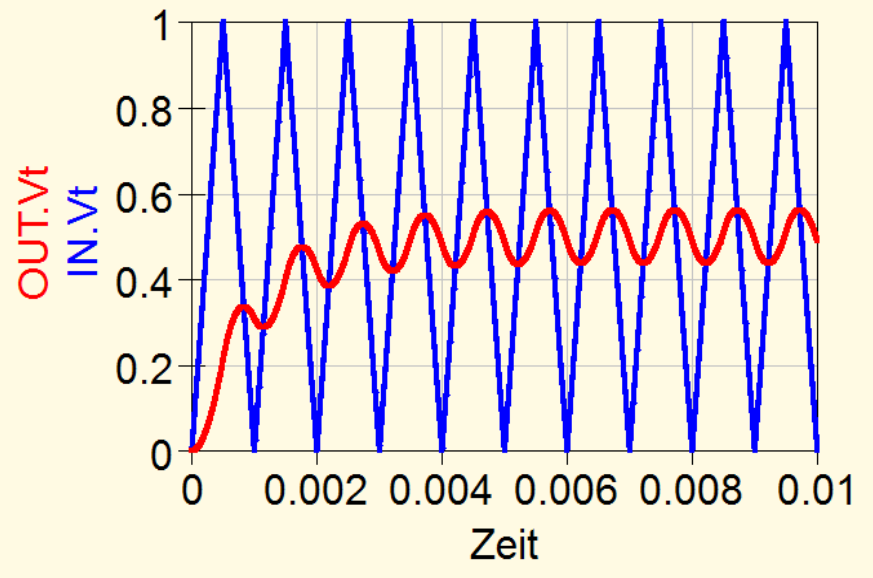
b) Puls- und Pausenlänge sind dabei extrem kurz oder sogar Null.



Wenn man nun noch weiß, **dass die Anstiegszeit immer als Teil der „High-Pegel-Zeit“ und die Abfallzeit immer als Teil der „Low-Pegel-Zeit“ gezählt wird**, ist die Sache noch einfacher:

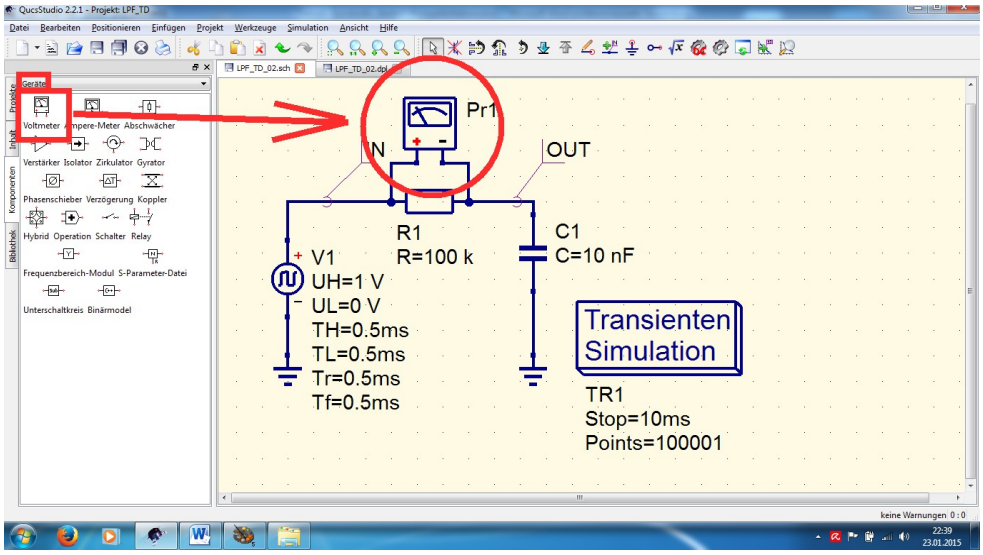
da braucht man nur alle 4 Zeitangaben gleich groß und 0,5 Millisekunden lang zu machen, um ein 1 kHz-Dreieck zu erhalten…

Simuliert wird für 10 ms mit einer Schrittweite von 100 Nanosekunden.



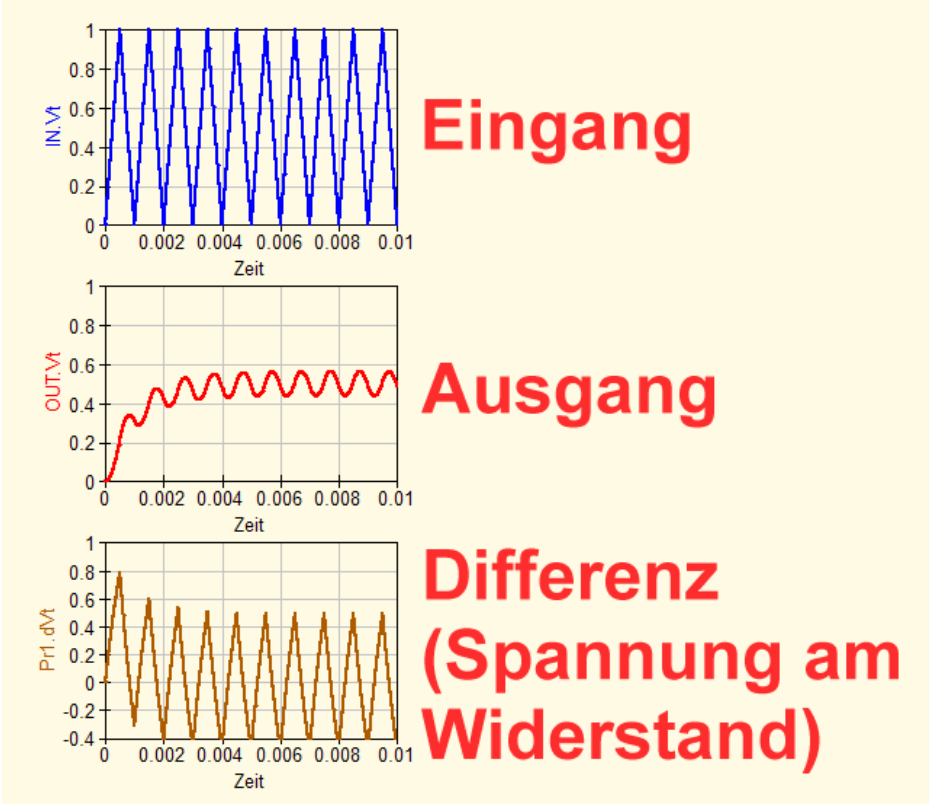
So sieht das Ergebnis aus, wenn man der waagrechten Achse die Bezeichnung „**Zeit**“ zuweist und (**über die Karteikarte „Grenzen“) die Achsenteilungen verfeinert.**

3.2.3.4. Differenz-Messungen



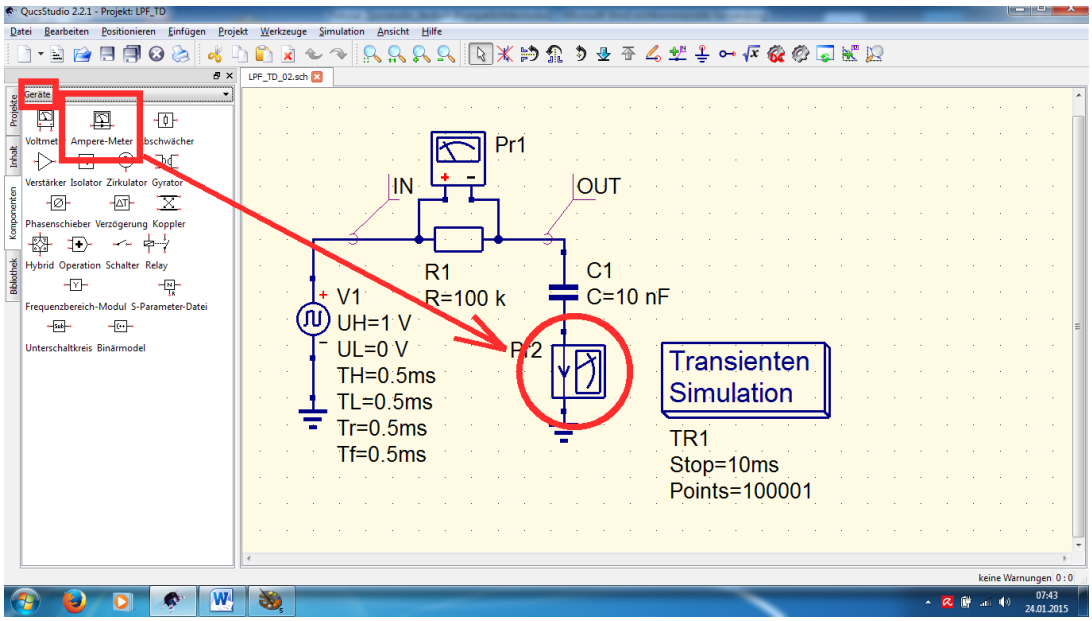
Nehmen wir an, dass uns im letzten Beispiel die Spannung über dem Widerstand interessiert. Das ist überhaupt kein Problem, denn unter „**Geräte**“ finden wir ein **Voltmeter Pr1,** das wir an den Widerstand anklemmen können.

Nimmt man das Diagramm aus dem vorigen Beispiel und kopiert es zwei Mal (…erst <CTRL +c>, dann <CTRL +v>), so kann man die drei Spannungen einzeln (mit gleichen Diagramm-Eigenschaften) untereinander anordnen.

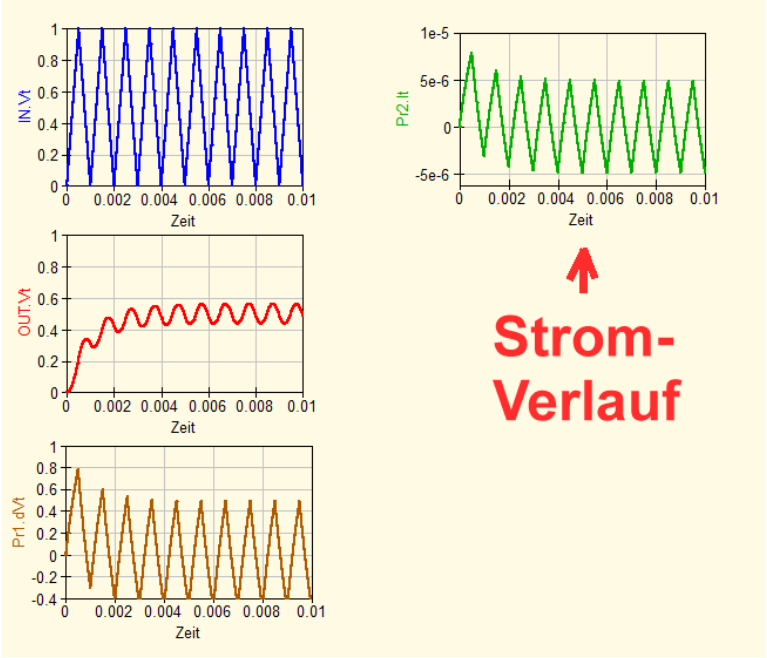


Lediglich beim Differenzsignal muss man den Anzeigebereich bis in den negativen Teil erweitern.

3.2.3.5. Strom-Messung



Auch hierfür gibt es unter „Geräte“ ein passendes Instrument („**Ampere-Meter**“), das wir in den Stromkreis einbauen und nochmals das vorige Beispiel simulieren.



Am schnellsten geht die Ergebnisausgabe, wenn wir uns eines der linken Diagramm kopieren (...am besten die Anzeige des Voltmeters). Anschließend sind folgende Maßnahmen nötig:

a) Die Eigenschaften-Liste wird durch einen rechten Mausklick auf dieses neue Diagramm geöffnet.

b) Auf der Karteikarte „Daten“ wird der alte Graph-Eintrag „Pr1.dVT“ gelöscht und dafür „Pr2.it“ aktiviert (= Button „Anwenden“). Auch die Linienbreite „2“ wird kontrolliert.

c) Dann wechselt man auf die Karte „Grenzen“ und entfernt bei der linken Achse das Häkchen für „manuell“.

Das obige Bild folgt automatisch auf den Druck von „OK“.

3.3. Die FFT (= Fast Fourier Transformation) als Brücke zwischen Zeit- und Frequenzbereich

3.3.1. Zusammenfassung der Grundlagen

Oft interessiert nicht nur der zeitliche Verlauf der einzelnen Spannungen (oder Ströme) in der Schaltung. Sobald nämlich ihre Kurvenform nicht mehr rein sinusförmig ist, gibt es außer der Grundwelle noch „**Oberwellen**“, also Signale mit doppelter, dreifacher, vierfacher Frequenz usw. Oder es handelt sich um ein „Gemisch“ oder um ein moduliertes Signal usw. und dann will man doch wissen, was da so alles drin steckt. **Die FFT berechnet ein solches Frequenzspektrum aus dem zeitlichen Verlauf des Signals.**

Allerdings ist eine ganze Reihe an Vorüberlegungen bereits für die Simulation im Zeitbereich nötig, wenn die FFT anschließend befriedigende und korrekte Werte liefern soll:

==========

Die Startfrequenz des simulierten Frequenzspektrums UND die Breite der dargestellten Linien UND der kleinste mögliche Frequenzschritt im Spektrum werden durch die **Simulationszeit in der Time Domain** bestimmt:

Maximaler Frequenzschritt = Startfrequenz = Linienbreite = 1 / Simulationszeit

Außerdem muss bei periodischen Signalen die **Simulationszeit immer ein ganzzahliges Vielfaches der Periodendauer des simulierten Signals sein** (…“es darf keine Periode angeschnitten werden“). Wird die Forderung verletzt, dann erhält man plötzlich zusätzliche „gelogene“ Linien und / oder ein starkes „Grundrauschen“ (Fachausdruck für diese Dreckeffekte: Leakage).

==========

Der „**Maximum Time Step**“ im Zeitbereich erfordert ebenfalls mehrere Überlegungen:

a) Er soll so klein sein, dass die simulierte Kurve nicht „eckig“ aussieht.

b) Die zugehörige „**minimale Samplefrequenz = 1 / Timestep“** muss **mindestens doppelt so hoch** sein wie die allerhöchste im simulierten Signal vorkommende Oberwellen-Frequenz. Erst dann ist die „Shannon-Bedingung“ erfüllt und man braucht nicht mit „Aliasing-Effekten“ zu rechnen (..die das Ergebnis fast unbrauchbar machen).

c) Für die FFT muss eine bestimmte **Mindestmenge an Samples** zur Verfügung stehen, denn sie bestimmt (**neben dem Maximum Timestep und der dadurch festgelegten Samplefrequenz) die höchste im Spektrum dargestellte Frequenz UND den Dynamikbereich der Amplitudenanzeige**.

==========

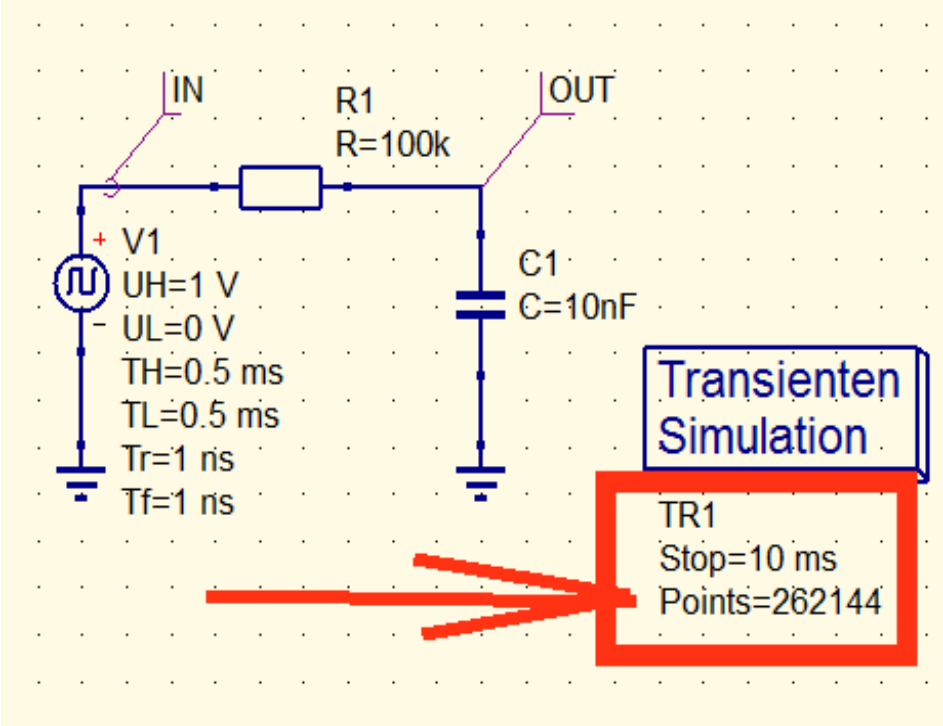
Die für eine FFT verwendete **Anzahl an Samples muss immer eine Zweierpotenz sein**. Das bedeutet, dass man immer 2 oder 4 oder 8 oder 16 oder 32 oder 64 oder 128 oder 256 oder 512 oder 1024 oder 2048 oder 4096 oder 8192 oder 16384 oder 32768 oder.....Werte für eine Berechnung einsetzen muss.

Verletzt man diese Bedingung, dann geht die „Fast Fourier Transformation“ in eine gewöhnliche Fourier Transformation über. Das bedeutet einen immensen Anstieg bei der Rechenzeit und beim Speicherplatz, denn die Rechenzeit steigt bei der FFT nur mit der Wurzel aus der Sample-Anzahl, bei der FT dagegen mit der Potenz der Sample-Anzahl....oft stürzt in einen solchen Fall sogar der Rechner ab.....

Und immer muss man im Hinterkopf behalten:
Mehr Samples für eine FFT geben bessere Amplitudenauflösung und höhere Grenzfrequenz, erhöhen aber entsprechend die Rechenzeit und die Größe der Ergebnisdatei!

==========

3.3.2. Ein praktisches Beispiel



Dazu benützen wir unseren nun schon gut bekannten Tiefpass, an den eine Pulsspannung mit f = 1000 Hz und UH = 1V bzw. UL = 0 V angelegt wird.
Die entscheidende Maßnahme ist nun die **Festlegung der „Point“-Anzahl**.

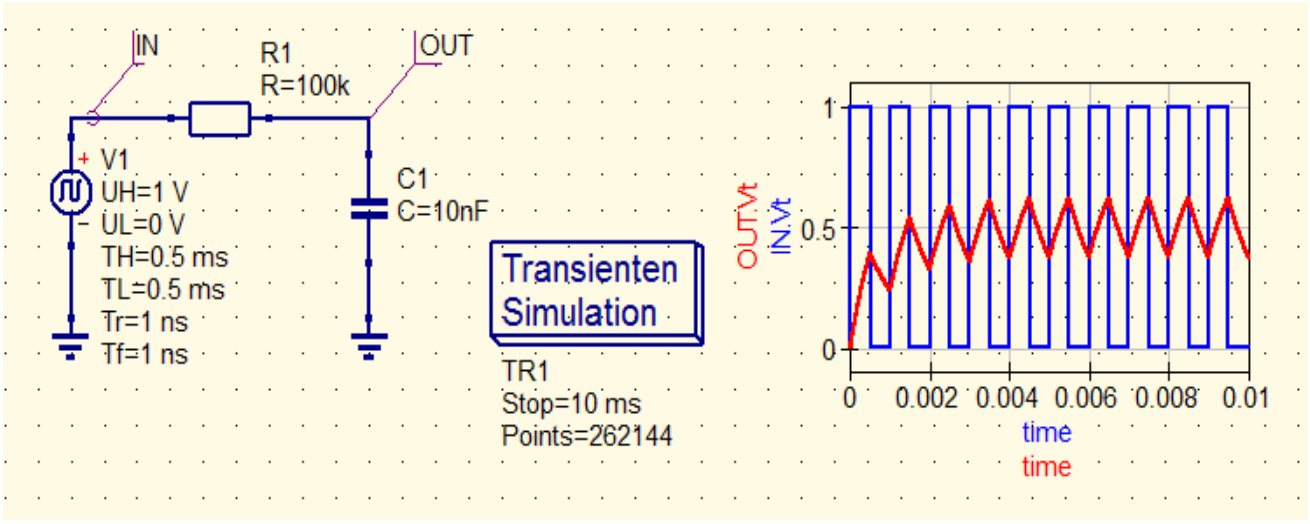
Sie muss eine **Zweierpotenz** sein und deshalb unterteilen wir die Simulationszeit von 10 Millisekunden in

262 144 Time Steps

Außerdem kennen wir nun schon gleich die Frequenzauflösung des Spektrums. Sie beträgt

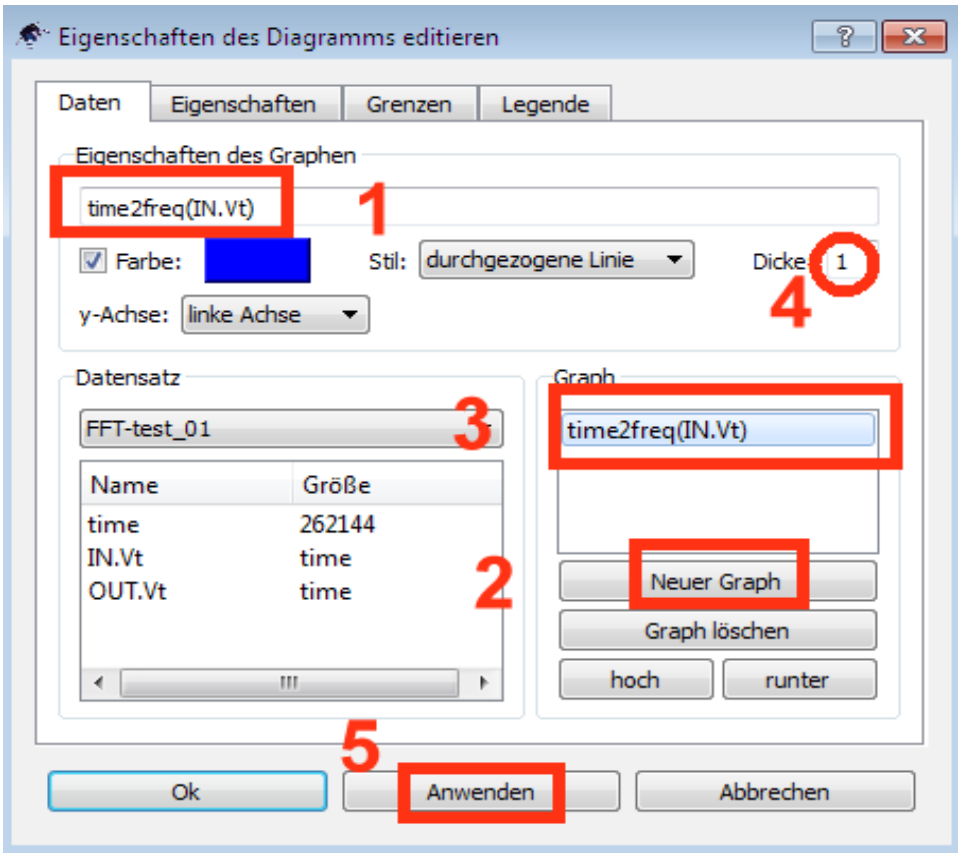
1/10ms = 100Hz

Nach der Simulation wählen wir wieder in bekannter Weise unser kartesisches Diagramm und lassen uns beide Spannungen für die Zeit von t = 0 …. 10ms darstellen. Aber jetzt machen wir uns das Leben leichter: Wir kopieren das Ergebnis in die Zwischenablage (<CRTL + c>), wechseln auf die Schaltbildseite und fügen es neben dem Schaltbild ein (<CTRL + v>). So haben wir einen besseren Überblick bei Änderungen:



Wenn wir nun nochmals auf das Zahnrädchen klicken und simulieren, schaltet qucsstudio zwar automatisch auf die Ergebnisseite. Aber die Schaltplanseite enthält jetzt auch das Diagramm mit den beiden Spannungen -- und bei Änderungen wird hier ebenfalls upgedatet.

Doch nun zurück zur Ergebnis-Bildschirm Da holen wir uns zuerst nochmals ein neues kartesisches Diagramm und setzen es im Arbeitsfenster ab. Darauf hin öffnet sich sofort sein Property Menü (= Eigenschaften-Fenster und es wartet Arbeit auf uns -- wir wollen die FFT zum Laufen bringen!



1:
In die Zeile „**Eigenschaften des Graphen**“ geben wir die FFT-Aufforderung ein:

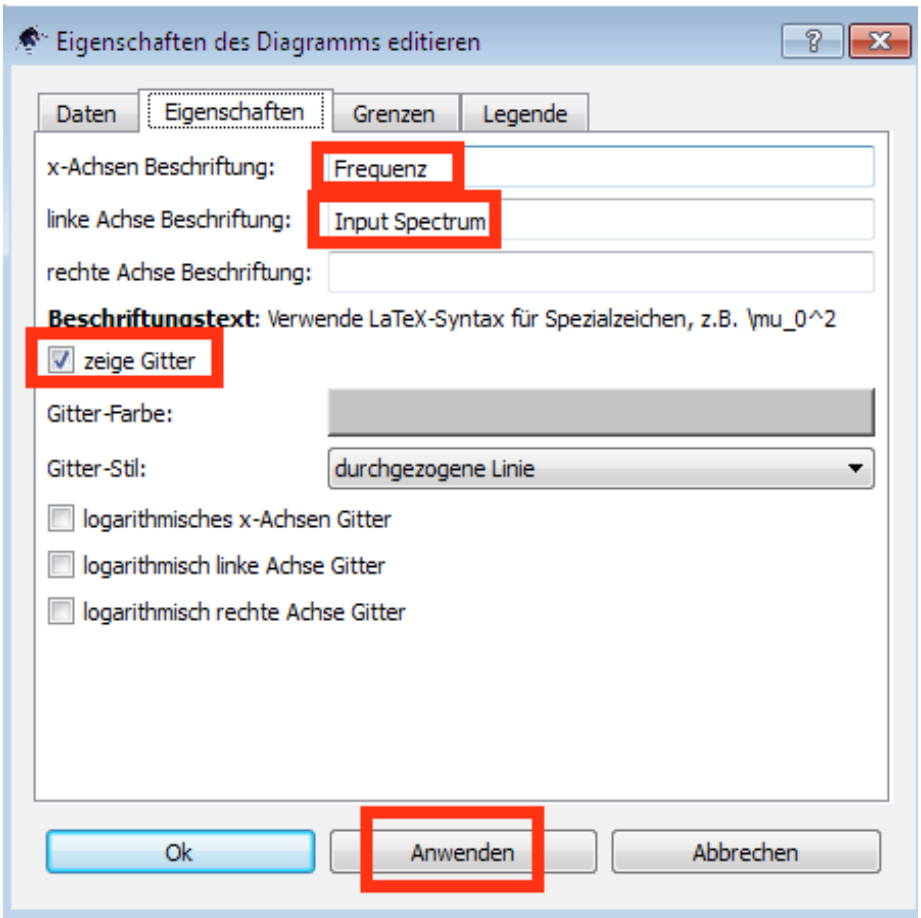
time2freq(IN.Vt)

2 und 3:
Das wird mit „**Neuer Graph**“ bestätigt und sollte dann in der Graphenliste auftauchen.

4:
Die Linienbreite stellen wir auf „**1**“ und

5:
klicken auf „**Anwenden**“.

Dann wechseln wir zur Karteikarte „Eigenschaften“.



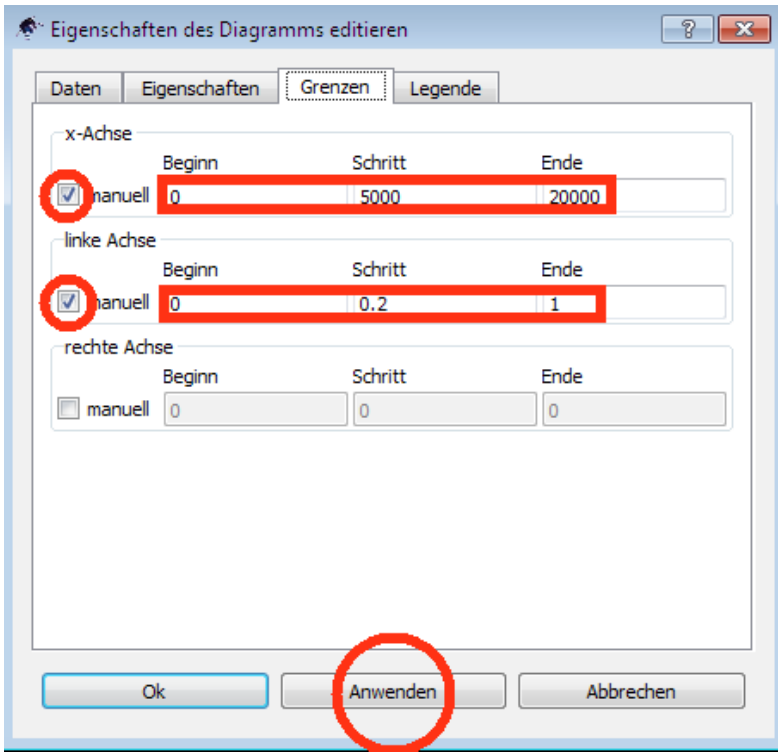
Bei der x-Achsen-Beschriftung wird „**Frequenz**“ eingetragen.

Bei der linken senkrechten Achse wollen wir „**Input Spectrum**“ sehen.

Das Häkchen bei „**zeige Gitter**“ wird angebracht und …..

das Ganze mit „**Anwenden“** abgeschlossen.

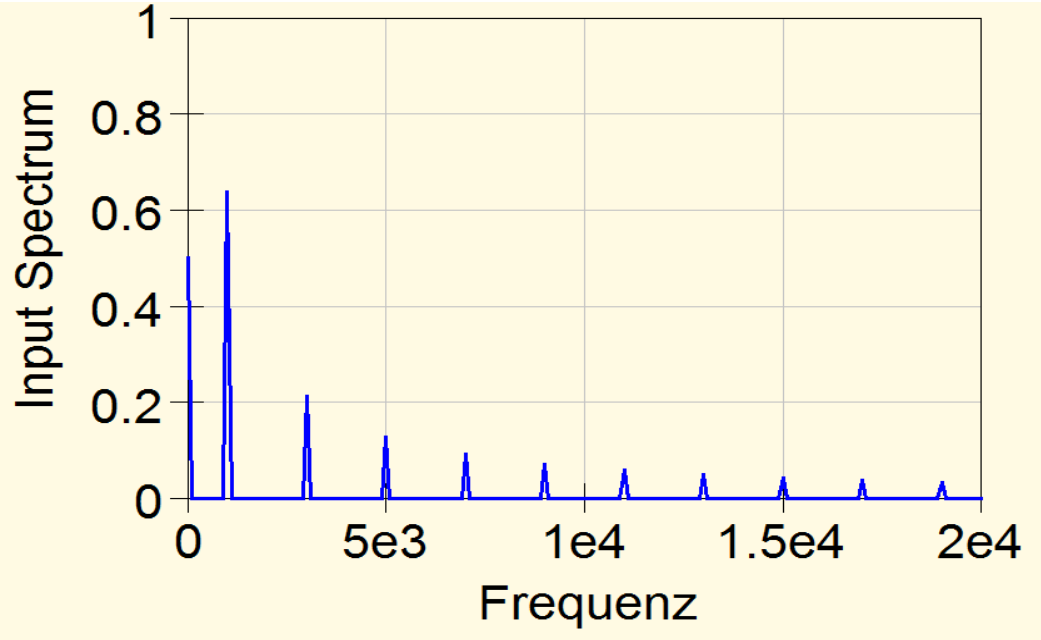
Auf der Karte „Grenzen“ geht es weiter. Beide Achsen werden auf „manuell“ umgestellt und folgende Bereiche vorgesehen:



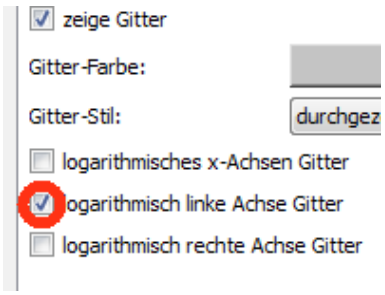
In X-Richtung:
Start bei Null, Maximalwert = 20000Hz, Schritt = 5000Hz

In Y-Richtung:
Start bei Null, Ende bei 1 Volt, Schritt = 0,2 Volt

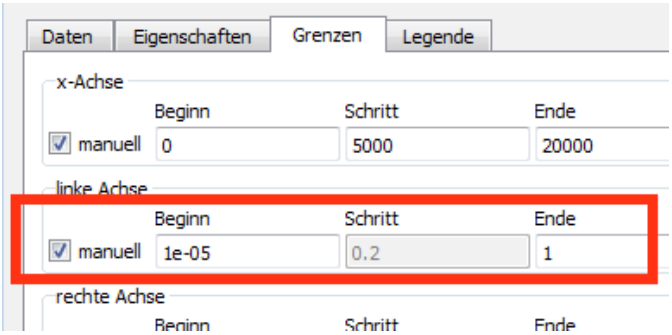
Dann reicht ein Klick auf „Anwenden“, gefolgt von OK, um das Ergebnis zu präsentieren:



Weil das so schön geklappt hat, wiederholen wir das Ganze mit einer **logarithmisch geteilten senkrechten Achse.**
Da kann man sich das Leben leicht machen: einfach das vorhin erstellte Diagramm kopieren und nochmals in die Zeichenfläche einfügen. Dann rechts auf das neue Diagramm klicken, „**Eigenschaften editieren**“ anklicken und loslegen:



Auf der Karteikarte „**Eigenschaften“** wird eine logarithmische linke Achse gewählt und das mit „Anwenden“ bestätigt

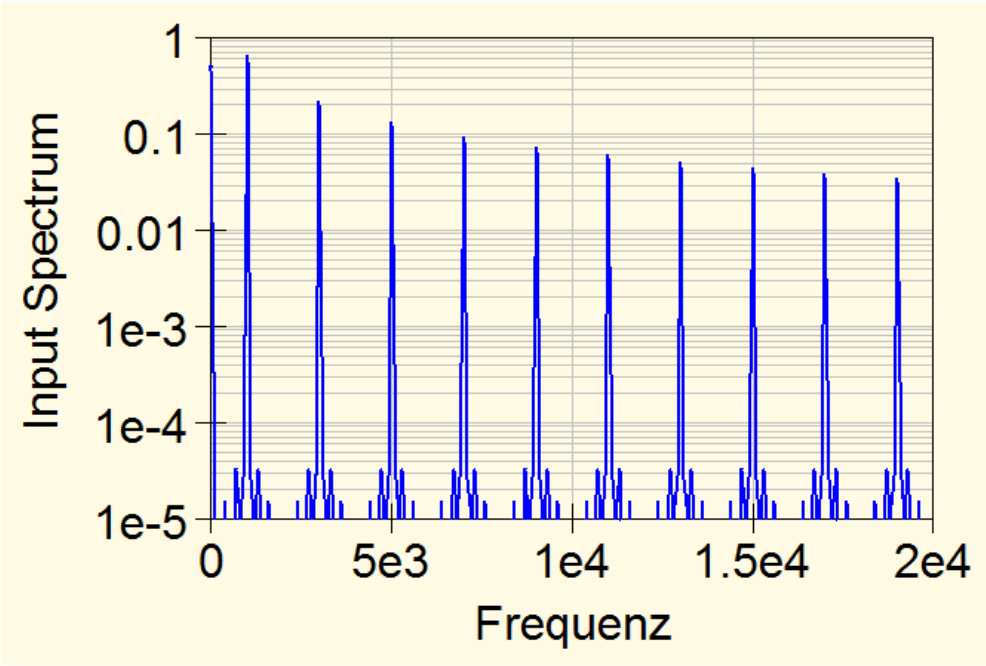


Auf der nächsten Karte „**Grenzen**“ müssen wir natürlich die Skalenteilung vorgeben:

Beginn = 1e-5

Ende = 1

Und mit „Anwenden“, gefolgt von OK, sind wir schon am Ziel.



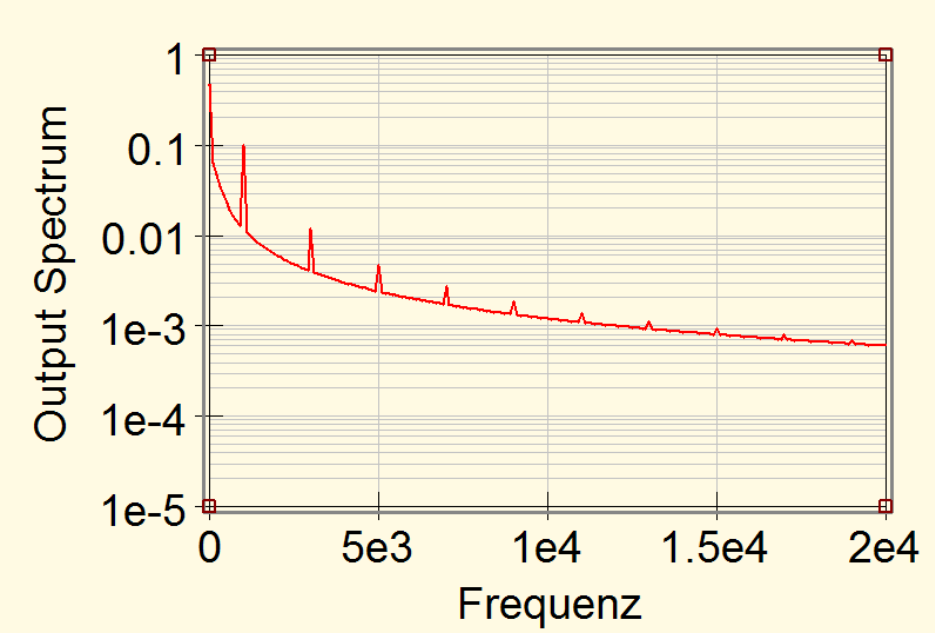
Na also….

Übrigens: wer eine **dickere Linie** wünscht, der rufe nochmals (nach einem Rechtsklick auf das Diagramm) „**Eigenschaften editieren**“ und „**Daten**“ auf. Sobald man in der Graphenliste die Formel **„time2freq(IN.Vt)“** anklickt, werden plötzlich die vorher gesperrten Felder freigegeben. Da kann man nun die gewünschte Liniendicke einstellen. Mit „**Anwenden“** wird das übernommen und mit OK schließt sich das Menü wieder.

Die nun folgende Aufgabe

„Stellen Sie zusätzlich das Ausgangsspektrum mit linearer Frequenzachse, aber logarithmisch geteilter senkrechter linker Achse dar“

dürfte nun keine Probleme mehr bereiten und sollte folgendes Diagramm liefern:



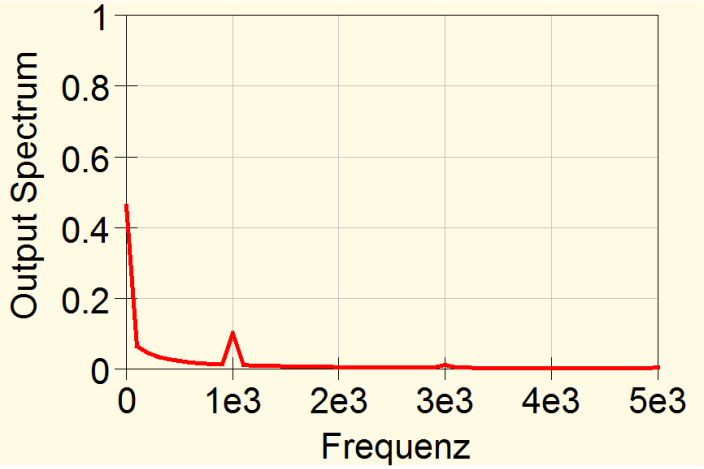
Lösung:
Man kopiert das passende Diagramm des Eingangsspektrums, editiert die Eigenschaften und gibt die neue Anweisung

time2freq(OUT.Vt)

ein. Ein Klick auf „Neuer Graph“ sorgt für die Übernahme und dadurch stehen plötzlich 2 Einträge in der Graphenliste. Da wird einfach der erste (mit der blauen Linienfarbe) gelöscht und die rote Linienfarbe samt „Dicke = 1“ überprüft. Dann darf man nicht vergessen, unter „Eigenschaften“ die neue Achsenbezeichnung „Output Spectrum“ vorzusehen.

Nach „Anwenden“ und OK sollte die obige Darstellung zu sehen sein.

Zusatzaufgabe:



Erstellen Sie dieses Diagramm für das **Ausgangsspektrum mit linear geteilter senkrechter Achse**.
Stellen Sie nur den **Bereich bis 5 kHz** dar und verwenden Sie eine Liniendicke „2“.

Dies ist die Lösung.

Wer sich an den doch **recht breiten Spektrallinien** im Diagramm stört, der bedenke:

Ihre Breite wird durch die Simulationszeit festgelegt und bei tsim = 10 ms bekommen wir eben nur eine Auflösung von b = 1 / tsim = 100 Hz.

Wer also 10 Hz Auflösung möchte, muss

t = 1 / 10 Hz = 100 ms

lang simulieren….

3.3.3. Der Kampf mit den Schaltvorgängen

3.3.3.1. Eine Windowing-Funktion muss her

Wer sich das simulierte Ausgangsspektrum mit logarithmischer senkrechter Achse im vorigen Beispiel genau ansieht, dem sollte etwas auffallen:

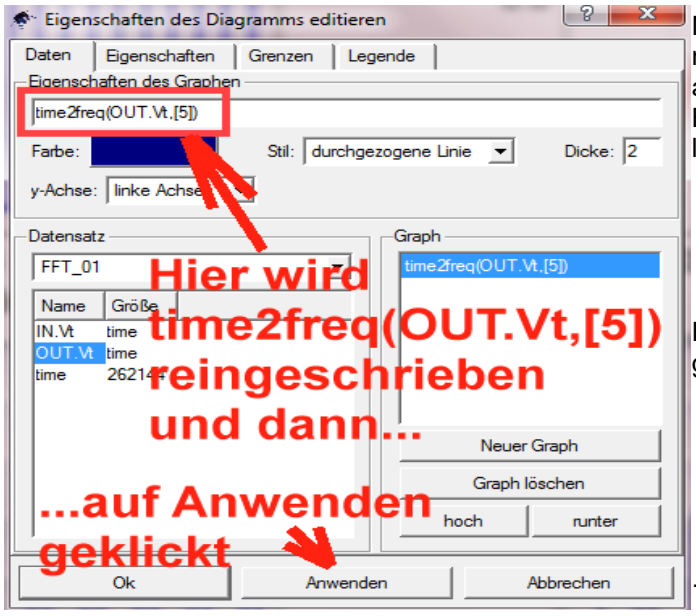
Wir finden den Raum zwischen den Spektrallinien gegenüber dem Eingangsspektrum sehr stark „mit Energie aufgefüllt“ und dieser Effekt klingt erst zu höheren Frequenzen hin ab.

Da muss man nochmals auf die Simulation in der Time Domain zurückkommen, um die Ursache zu finden: Nach dem Start der Simulation finden wir am Ausgang neben der Sägezahnspannung des Dauerbetriebs den typischen „**Einschaltvorgang durch die Kondensatoraufladung“. Er besitzt ein lückenloses Energiespektrum in der Frequency Domain und das addiert sich zum Linienspektrum, welches durch das periodische Eingangssignal erzeugt wurde.** Abhilfe bringt hier eine „**Window-Funktion**“. Sie wird mit dem Verlauf des Ausgangssignals multipliziert und startet meist mit dem Wert „**Null**“. Ebenso endet sie meist mit dem Wert Null, denn damit schwächen wir alle Bereiche (= Start und Ende) ab, bei denen solche Schaltvorgänge mitspielen und abrupte Übergänge vorkommen. Damit arbeiten wir bei unserer FFT tatsächlich mit dem „eingeschwungenen Zustand“ (= steady state) und untersuchen nur den Dauerbetrieb. Qucsstudio stellt fünf der bekanntesten Window-Funktionen zur Verfügung und ihre Anwendung ist leicht.

Der Aufruf der FFT lautet nun für unser Ausgangssignal:

time2freq(OUT.Vt,[w]) oder time2freq(OUT.Vt,w)

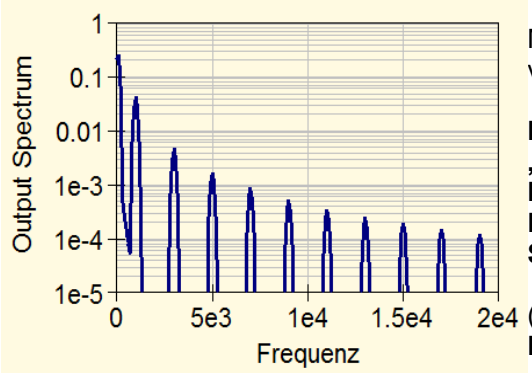
(Ein Tipp: Bitte das Komma **vor die eckige Klammer setzen und nicht vergessen**. Sonst läuft nichts. Und ein weiterer Tipp des Programmautors nach dem Probelesen: **die eckige Klammer KANN weggelassen werden, aber das Komma MUSS BLEIBEN**)



Nach meiner persönlichen Erfahrung fährt man mit w = 3 (**Hanning)** oder w = 5 (**Blackmann**) oft am besten, deshalb wenden wir sie in unserem Beispiel auf das Ausgangsspektrum mit logarithmischer senkrechter Achse an.

Hier ist als Beispiel zu sehen, wie das mit „w = 5“ geht….

….und so sieht das Ergebnis aus:

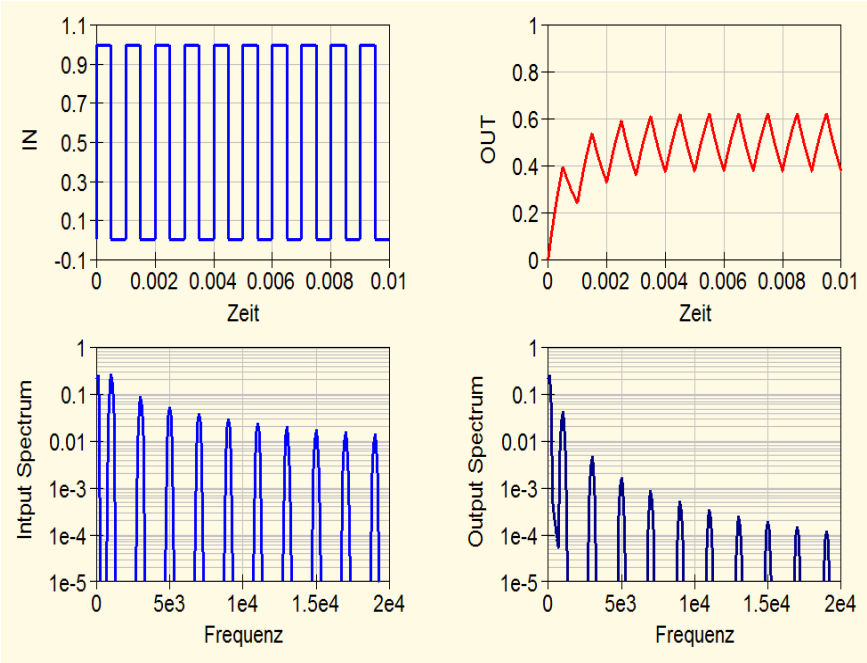


Man erkennt, dass der Einfluss des Startvorgangs nun fast verschwunden ist, aber:

Durch die Window-Funktion wurde die Zahl der „wirksamen Samples“ für die FFT reduziert. Und das ergibt leider eine schlechtere Frequenzauflösung mit größerer Linienbreite, denn dadurch wurde die „wirksame Simulationszeit“ kürzer.

(Das hatten wir ja vorhin schon:
Frequenzauflösung = = 1 / Simulationszeit)…….

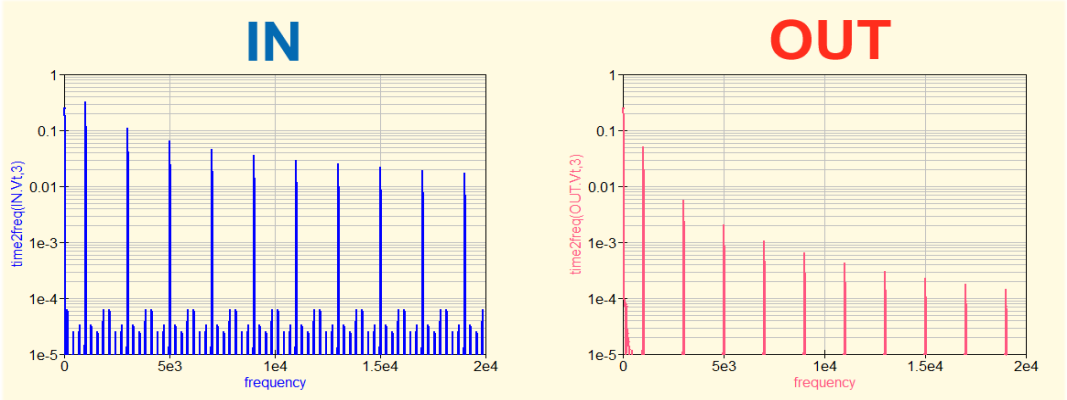
Jetzt kann man sich noch eine Freude machen und vier Diagramme auf dem Schirm anordnen, nämlich IN und OUT sowohl in der Time Domain wie auch in der Frequency Domain:



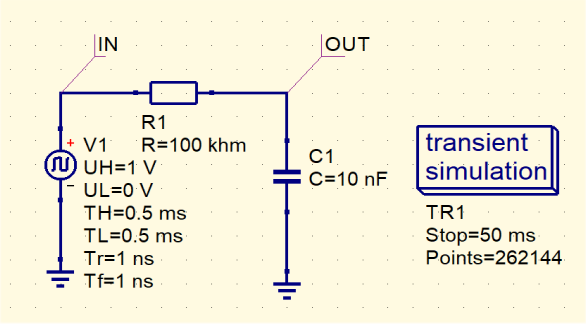
Daraus kann man sehr gut das Tiefpass-Verhalten der Schaltung entnehmen und analysieren.

Hinweis auf eine weitere Tücke beim Ausgangsspektrum mit Window-Bearbeitung:
Es werden nicht nur die Spektrallinien breiter – durch die wegen der Window-Funktion „weggeworfenen Samples“ wird auch der korrekte Absolutwert der Spannungsamplituden vermindert (….aber die relativen Unterschiede zwischen den Linien bleiben erhalten). Zur Ermittlung absoluter korrekter Amplitudenwerte muss also die Window-Funktion ausgeschaltet werden!

Aufgabe:
Durch die Window-Funktion hat sich ja die Anzahl der zur Verfügung stehenden Samples vermindert. Das entspricht einer kürzeren Simulationszeit und ergibt deshalb eine schlechtere Frequenz-Auflösung.
Gleichen Sie das durch eine **Erhöhung der Sample-Zeit auf 50 ms** aus und verwenden Sie weiterhin ein Hanning-Fenster (Index = 3). Kontrollieren Sie die Verbesserung durch einen Vergleich mit dem obigen vorigen Beispiel



Zusammenfassende Aufgabe:



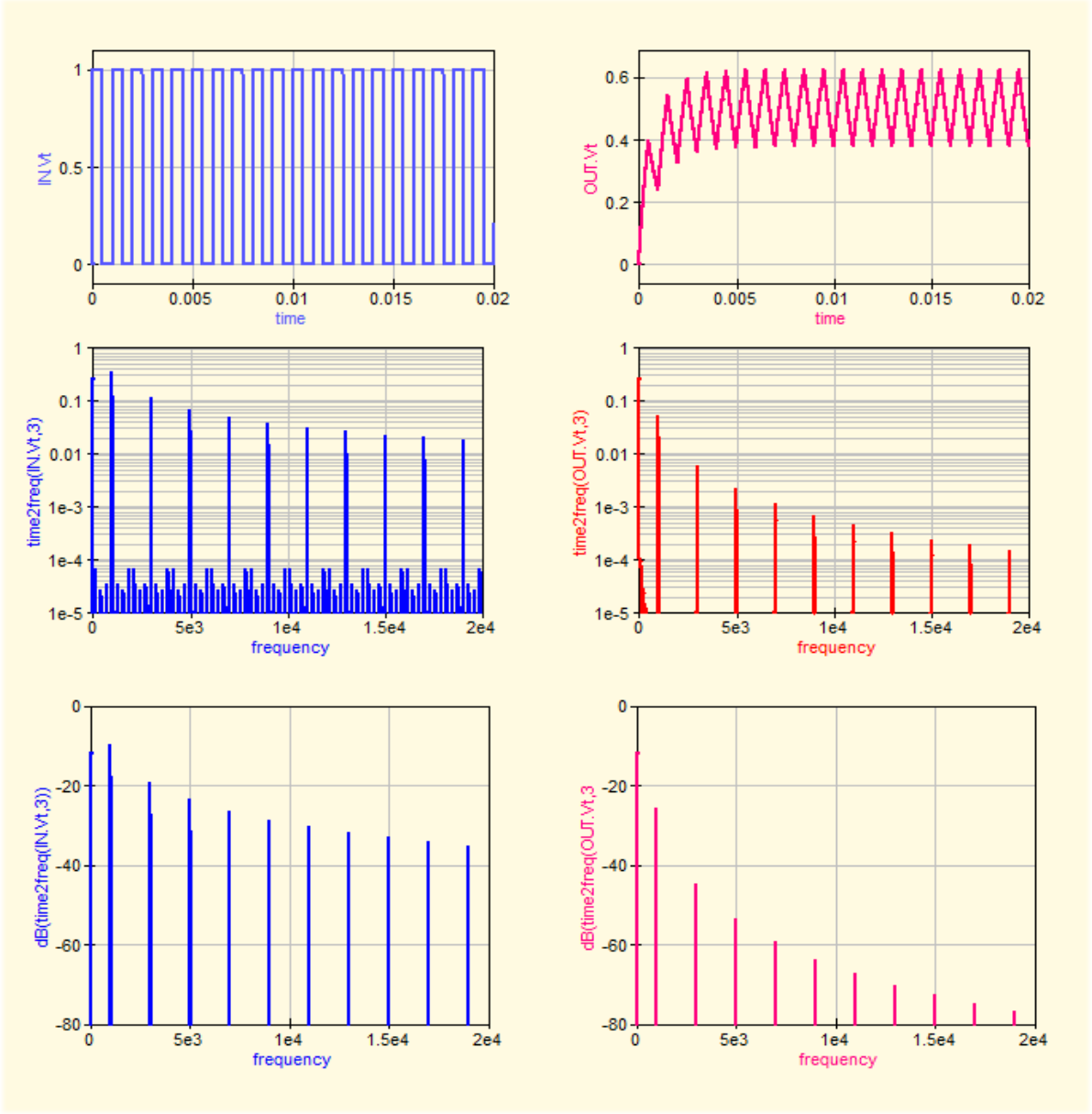
Verwenden Sie die Schaltung unseres bisher verwendeten RC-Tiefpasses und stellen Sie eine Simulationszeit von 50 ms ein.

Simulieren Sie die Eingangs- und die Ausgangsspannung in der Time Domain

Simulieren Sie für jede Spannung das Frequenzspektrum bis f = 20 kHz (mit logarithmischer senkrechter Achse).

Simulieren Sie zusätzlich diese Spektren mit „dB“-Anzeige bis f = 20 kHz (Achtung: senkrechte Achse auf „linear“ einstellen).

Anweisung zur Spektrumsberechnung mit dB-Anzeige: dB(time2freq(IN.v)) bzw. dB(time2freq(OUT.v))



3.3.3.2. Umrechnung von Spannungswerten in „dBm an 50 Ohm“

Die Darstellung in dB bietet den Vorteil, dass große Spannungs**unterschiede** in einem einzigen Diagramm sehr übersichtlich dargestellt werden können. Allerdings fehlt eine „Kalibrierung“ oder „Eichung“, um direkt die echten Amplituden zu bestimmen-
Hier hat man sich mit der Einführung eines „**Bezugspegels in dBm**“ geholfen – die Definition lautet:

Pegel in dBm = eine Leistung von 1 Milliwatt am Systemwiderstand (meist 50Ω)

Für **Z = 50 Ω ergibt das einen Effektivwert der Spannung** von

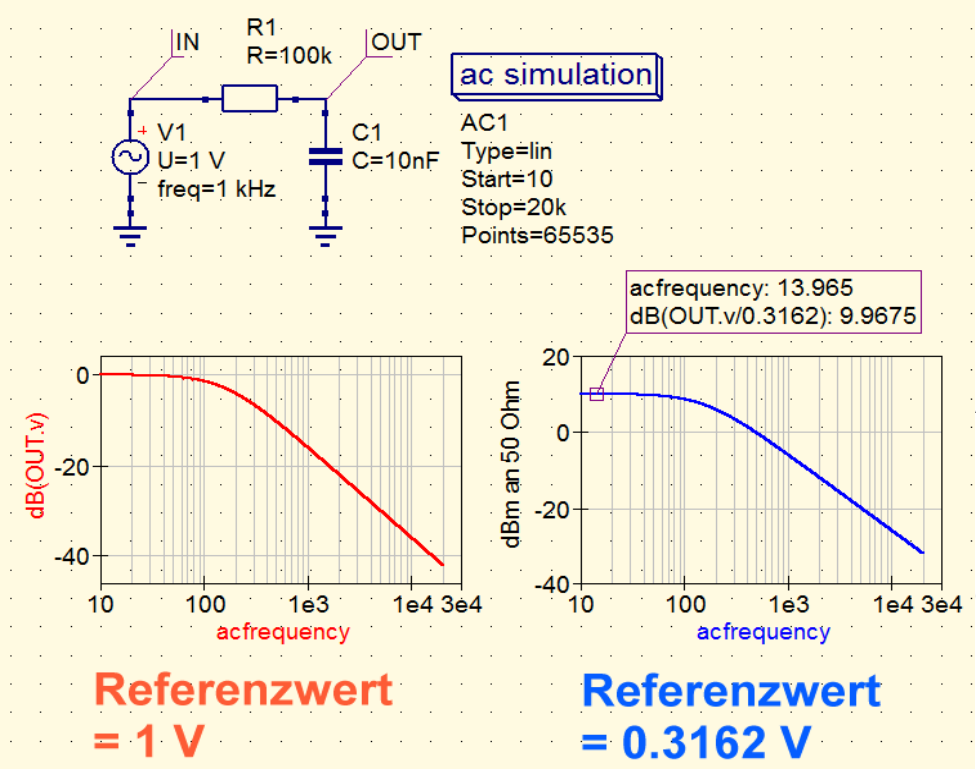
U = sqrt(Leistung * Widerstand) = sqrt[1exp(-3) * 50Ω] = 0,2236 V

Allerdings müssen wir bei allen **Simulationsprogrammen** hier **sehr aufpassen, denn dort wird bei den Spannungen stets mit dem Spitzenwert (= peak value) gearbeitet. Zur Leistungsberechnung gehört aber der Effektivwert....**

Deshalb benützen wir auch bei qucsstudio in diesem Fall einen Spitzenwert von

0,2236V * sqrt(2) = 0,3162V

als Referenz.



Das linke Bild zeigt unser erstes Tiefpass-Simulations-Beispiel, bei dem wir in Kapitel 3.1.1. mit dem AC-Sweep begonnen haben.

Der linke Ergebnis-Graph ist die **dB-Ausgabe der Ausgangsspannung „OUT.v“** bei dem mit einer Referenzspannung von

Uref = 1 Volt

gearbeitet wird.

Die Anweisung für das rechte untere Diagramm lautet dagegen

dB(out.v/0.3162)

Zusätzlich wurde dort noch die Bezeichnung der senkrechten Achse in

„dBm an 50 Ohm“

geändert.

Wer den Taschenrechner nimmt und den Spitzenwert des Ausgangssignals bei tiefen Frequenzen mit ca. 1 V durch 0,3162 V (= Referenzpegel) dividiert, dann „log x“ drückt und das Ergebnis mit 20 multipliziert, der erhält „+10 dBm“.

Die Formel lautete ja:

Spannungspegel in dBm = 20 * log(U / Uref) = 20 * log(OUT.v / 0,3162)

Und genau das zeigt das rechte Diagramm.

Im linken Diagramm wird dagegen vom Programm die Referenz „U = 1 V“ verwendet. Dort finden wir bei tiefen Frequenzen natürlich einen Pegel von Null dB, der zur Eingangsspannung von 1 V passt. Die korrekte Einheit an der senkrechten Achse dieses Diagramms ist
„dBV“ (Sprich: „dB Volt“).

4. Zweites Projekt: Signalformen und Oberwellen = eine kleine FFT-Übung

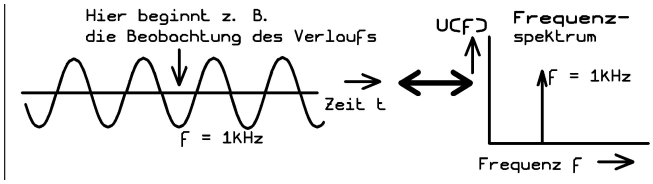
4.1. Grundlagen

Selbst viele Laien wissen sofort die Antwort auf die Frage: „Wie sieht die Spannung aus, die in jedem Haus aus der Steckdose kommt?“ Sie lautet immer: „Na klar, ein Sinus mit 230 Volt Effektivwert und einer

Frequenz von 50 Hz“. Aber bei der nächsten Frage:

„Welche verschiedenen Frequenzen stecken eigentlich in diesem Steckdosensignal?“

kriegt man selbst von Fachleuten oft die falsche Antwort. Das liegt daran, dass man erstens genau die „Mess- und Beobachtungsbedingungen“ beachten muss, denn damit ändert sich direkt die Lösung. Zweitens muss man sozusagen die große Lupe nehmen und das Signal mit der Lupe absuchen (...wie einen Hund nach Flöhen..) und auf jede Kleinigkeit oder Unregelmäßigkeit achten, denn die sind enorm wichtig.....Formulieren wir es mal ganz korrekt:

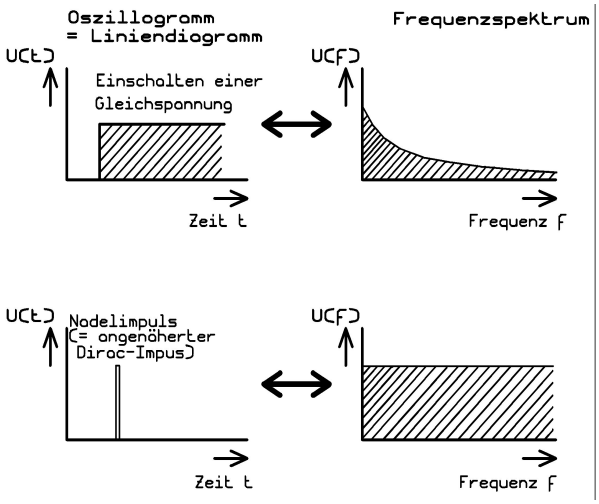


Nur eine rein sinusförmige Spannung (hier: f = 1 kHz), die vor sehr langer Zeit eingeschaltet wurde und die in absehbarer Zeit nicht aufhört (...also weiterläuft bis zu den Flügen von Raumschiff Enterprise im Jahr 2150....) besteht nur aus einer einzigen Frequenz.

Da diese Formulierung so gar nicht zu dem passt, was man kennt und weiß und täglich sieht, folgt hier die Lösung des Rätsels: Man muss nämlich sorgfältig zwischen „**Schaltvorgängen**“ und „**Verzerrungen der Kurvenform bei periodischen Signalen**“ unterscheiden.

Schaltvorgänge

Jede **einmalige Änderung** (= Einschalten, Ausschalten, Aufdrehen, Kleinermachen, kurz Unterbrechen.....) erzeugt ein **einmaliges „Störsignal“ oder „Störspektrum“**, das vom Entstehungsort wegwandert. Das kennt jeder von uns als Knacken im Radio, wenn er plötzlich das Licht im Raum einschaltet....man hat das bloß noch nie so gesehen oder beachtet! Entscheidend in diesem Störsignal ist die Tatsache, dass ***der komplette Frequenzbereich einschließlich der sehr hohen Frequenzen während dieser kurzen Zeit lückenlos mit Signalen belegt ist***. Die tiefen Frequenzanteile breiten sich dann natürlich über Kabelverbindungen aus, die hohen Anteile werden dagegen schon von den als Antennen wirkenden Verbindungsdrähten der Schaltung als elektromagnetische Wellen abgestrahlt. Und das ist der erwähnte Knacks im Radio, der dann selbst in einem batteriebetriebenen Kofferradio zu hören ist! Glücklicherweise sinken die Amplituden der hohen Frequenzanteile bei den üblichen Schaltvorgängen sehr schnell ab, sonst könnten wir überhaupt nicht drahtlos kommunizieren. Die Geschwindigkeit der Änderung beim „Schaltvorgang bestimmt wesentlich diese Amplitudenverteilung und wenn sich etwas sehr langsam ändert, können wir diese Effekte fast gar nicht nachweisen. Aber es gibt ein Signal, in dem alle Frequenzen enthalten sind und vor dem sollte man sich auch in abgeschwächter Form hüten: es ist der „**Dirac-Impuls**“ mit unendlich kurzer Pulsdauer und unendlich großer Amplitude, denn er kann geradezu sagenhaft stören.



Hier zwei Beispiele:
Links die Oszillogramme („Liniendiagramme“) vom Einschalten einer Gleichspannung und vom Nadelimpuls, der den Dirac-Impuls annähern soll.

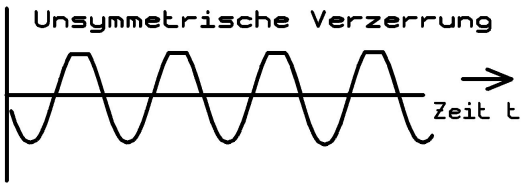
Rechts die zugehörigen Frequenzspektren, wobei für den Nadelimpuls eine sehr kleine Pulsdauer angenommen wurde. (Genauere Details folgen im Sampling-Kapitel).
Man erkennt gut, dass es kein Entkommen vor diesen Störfrequenzen gibt, denn es existieren bei beiden Signalen **keine Lücken im Spektrum**!

4.2. Verzerrungen bei periodischen Signalen

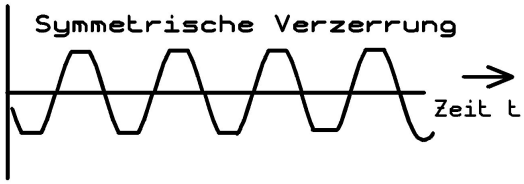
Da merken wir uns gleich den Schlüsselsatz:

Auch die allerkleinste, aber sich regelmäßig wiederholende Abweichung von der idealen Sinusform führt zum zusätzlichen Auftreten neuer Signale im Frequenzspektrum!

Diese neuen Frequenzen tragen den Fachausdruck „Harmonische“ oder „Oberwellen“ und sind stets **ganzzahlige Vielfache der Grundfrequenz** (Im Klartext heißt das: man findet nun plötzlich auch die doppelte, dreifache, vierfache, fünffache, sechsfache....Frequenz im Spektrum!).
Dabei gilt noch eine weitere wichtige Spielregel:



Ist nur eine Halbwelle von der Verzerrung betroffen ODER die positive und die negative Halbwelle sind unterschiedlich verformt, dann spricht man von ***unsymmetrischer*** Verzerrung. Man findet dann ***alle Oberwellen*** im Spektrum.



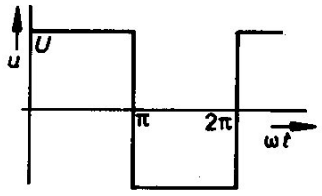
Ist das ursprünglich ideale Sinussignal aber ***symmetrisch*** verzerrt (= positive und negative Halbwelle sind exakt in derselben Weise verändert), dann gibt es ***keine geraden***, sondern nur ***ungerade Oberwellen*** (= außer der Grundfrequenz beobachtet man zusätzlich nur die dreifache, die fünffache, die siebenfache, die neunfache.....Grundfrequenz).

Abschließender Hinweis:
Sobald man am idealen Sinus irgendwas im ***Rhythmus eines anderen Signals verändert*** (= Amplitude, Phase oder Frequenz), gelangt man zu den entsprechenden **Modulationsarten.**

Auch hier müssen folglich neue Signale entstehen. Aber damit sind wir bei einem anderen Thema…

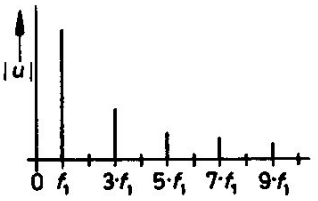
Auf dem nächsten Blatt folgt noch eine kleine Zusammenstellung einiger Signalformen und der zugehörigen Spektren:

Rechteckspannung

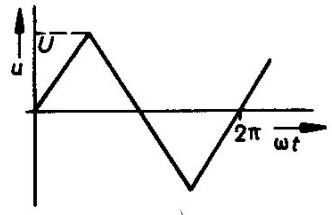


$$f(t) = \frac{4 \cdot U}{\pi}\left(\sin \omega \cdot t + \ldots + \frac{1}{3}\sin 3\,\omega t + \frac{1}{5}\sin 5\,\omega t + \ldots\right)$$

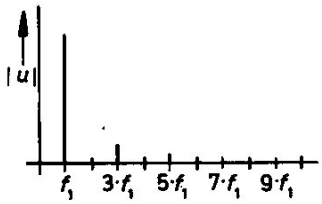
Spektrum: Absolutwerte der auftretenden Frequenzen



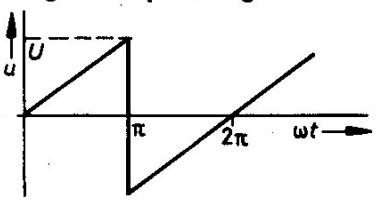
Dreieckspannung



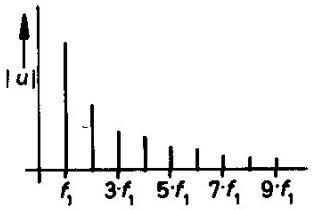
$$f(t) = \frac{8\,U}{\pi^2}\left(\sin \omega\, t - \ldots + \frac{1}{3^2}\sin 3\,\omega t + \frac{1}{5^2}\sin 5\,\omega t - \ldots\right)$$



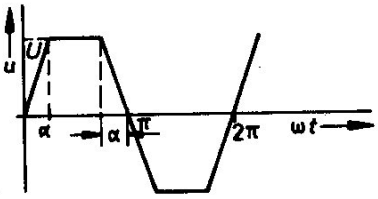
Sägezahnspannung



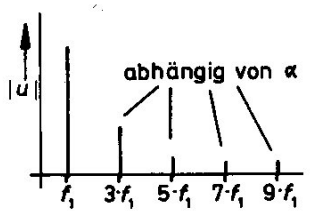
$$f(t) = \frac{2\,U}{\pi}\left(\sin \omega t - \frac{1}{2}\sin 2\,\omega t + \ldots + \frac{1}{3}\sin 3\,\omega t - \ldots + \right)$$



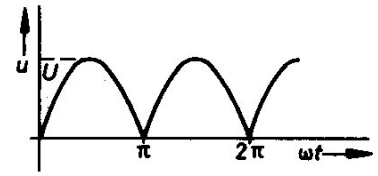
Trapezspannung



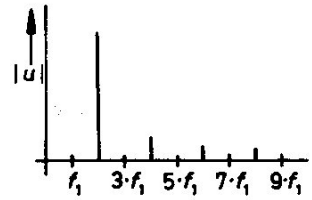
$$f(t) = \frac{4 \cdot U}{\pi \cdot \alpha}\left(\sin \alpha \cdot \underset{\text{tot}}{\sin \omega t} + \ldots + \frac{\sin 3\,\alpha}{3^2} \cdot \sin 3\,\omega t + \ldots + \frac{\sin 5\,\alpha}{5^2}\sin 5\,\omega t + \ldots\right)$$



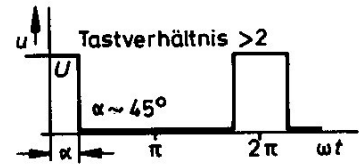
Doppelweg-Gleichrichtung



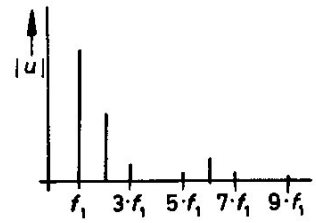
$$f(t) = \frac{4\,U}{\pi}\left(\frac{1}{2} - \frac{1}{1 \cdot 3}\cos 2\,\omega t - \ldots - \frac{1}{3 \cdot 5}\cos 4\,\omega t - \ldots - \frac{1}{5 \cdot 7}\cos 6\,\omega t - \ldots\right)$$



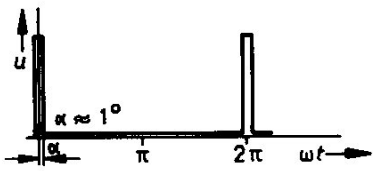
Rechteckspannung
Tastverhältnis > 2

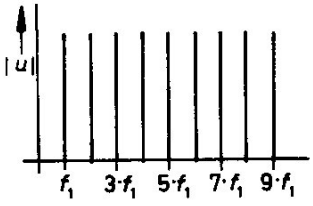


$$f(t) = \frac{2\,U}{\pi}\left(\frac{\alpha}{2} + \frac{\sin \alpha}{1}\cos \omega t + \ldots + \frac{\sin 2\,\alpha}{2}\cos 2\,\omega t + \ldots + \frac{\sin 3\,\alpha}{3}\cos 3\,\omega t + \ldots\right)$$



Nadelimpulse





4.3. Simulation des Spektrums eines einmaligen Pulses

Wir wollen mal prüfen, ob die Behauptung aus dem vorigen Kapitel stimmt. Dazu verwenden wir ein Puls-Signal mit folgenden Daten:

Umin = 0V / Umax = 1000 V / Anstiegs- und Abfallzeit je 1 Mikrosekunde / Pulsdauer = 1 Millisekunde.

Zusätzlich bauen wir eine „**Delay Time“ nach dem Start von 2 Millisekunden ein** und simulieren bis 50 Millisekunden im Zeitbereich die Spannung am Lastwiderstand. Für eine gute Auflösung bei der FFT sehen wir 262144 Samples vor. Den Signalverlauf bilden wir mit einer „**PWL“-Quelle** nach, der folgende Wertepaare zur Verfügung gestellt werden:

bei t = Null	**U = 0**
bei t = 2 Millisekunden	**U = 0**
bei t = 2,001 Millisekunden	**U = 1000 V**
bei t = 3,001 Millisekunden	**U = 1000 V**
bei t = 3,002 Millisekunden	**U = 0**
bei t = 50 Millisekunden	**U = 0**

Diese Möglichkeit, sich einen bestimmten Spannungsverlauf durch eine Datei mit vielen Wertepaaren (= Zeitpunkt + Amplitude bei einem bestimmten Zeitpunkt) zu erzeugen, gibt es in allen Simulationsprogrammen und das ist eine überaus praktische Sache. Dabei werden bei der Simulation die einzelnen „Samples“ durch gerade Linien miteinander verbunden und der gewünschte Verlauf angenähert.

Allerdings muss man beim qucsstudio auf einige Feinheiten achten:

c) Der Aufruf dieser Quelle erfolgt über „Komponenten / Quellen“. Aber die **PWL-Quelle sucht man vergeblich**, denn hier heißt sie „**dateibasierte Spannungsquelle“.**

d) Der Spannungsverlauf muss in einer **selbst geschriebenen „*.csv“-Datei“** als eine Folge von Wertpaaren angegeben werden. **Die Datei selbst wird dann in das betroffene Projekt hineinkopiert.** (Übrigens: „csv“ ist die Abkürzung für „comma separated values“…)

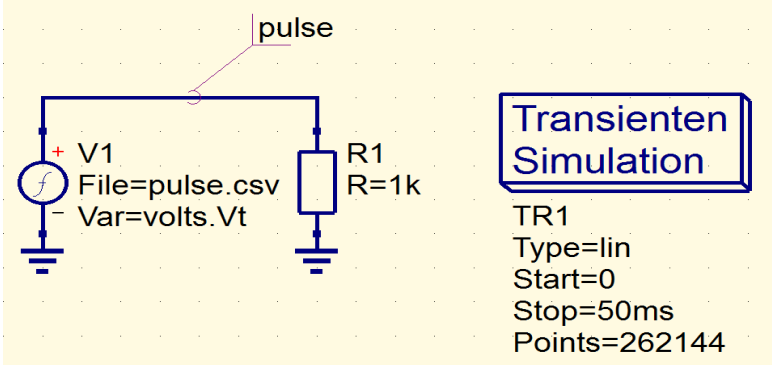
Fangen wir mit der Datei „**pulse.csv**“ an.
Begonnen wird beim Nullpunkt mit Null Volt. Bei t = 2 Millisekunde haben wir immer noch Null Volt, aber eine Mikrosekunde später sind es bereits 1000 Volt. Erst ab 3,002 Mikrosekunden sind wir wieder bei Null Volt und das bleibt so bis zum Ende der Simulationszeit von 50 ms.

Vorsicht:
Jedes Wertpärchen (= erst die Zeit, dann die Spannung) kommt in eine eigene Zeile, die jeweils mit der Eingabe-Aufforderung (= Return-Taste) abgeschlossen werden muss.
So sieht dann der Datei-Inhalt von „**pulse.csv**“ aus:

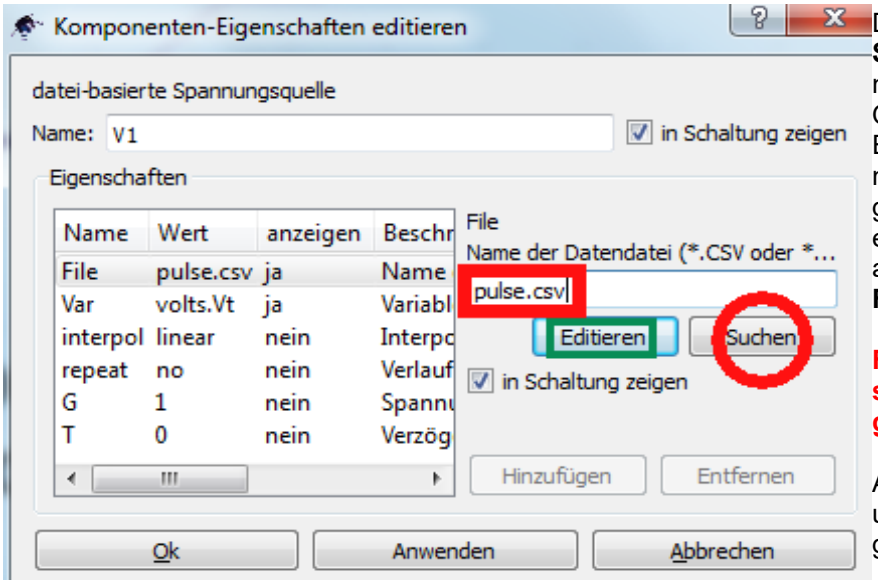
```
0          0
2e-3       0
2.001e-3   1e3
3.001e-3   1e3
3.002e-3   0
50e-3      0
```

Geschrieben wird das mit einem Texteditor und man darf nicht vergessen, vor dem Abspeichern als „**pulse.csv**“ auf „**Alle Dateien / ANSI**“ umzuschalten. **Speicherort ist natürlich das Projekt, in dem die zugehörige Schaltung abgelegt ist.**

Zum korrekten Einbinden der Wertedatei wird mit der rechten Maustaste auf das Symbol der **Quelle** geklickt und dann über „Eigenschaften editieren“ ihr Property-Menü geöffnet.



Das soll diese Schaltung ergeben.



Die „**dateibasierte Spannungsquelle**“ gibt es natürlich unter „Komponenten / Quellen“. Nachdem man ihre Eigenschaften editiert hat, muss man über „**Suchen**“ den Pfad zur gespeicherten „**pulse.csv**“-Datei einstellen. Ihre Bezeichnung muss anschließend im **rot markierten Feld** auftauchen.

Falls etwas zu ändern ist, kann sie jederzeit über „Editieren“ geöffnet und korrigiert werden.

Anschließend reicht „Anwenden“ und OK, damit die Simulation gestartet werden kann.

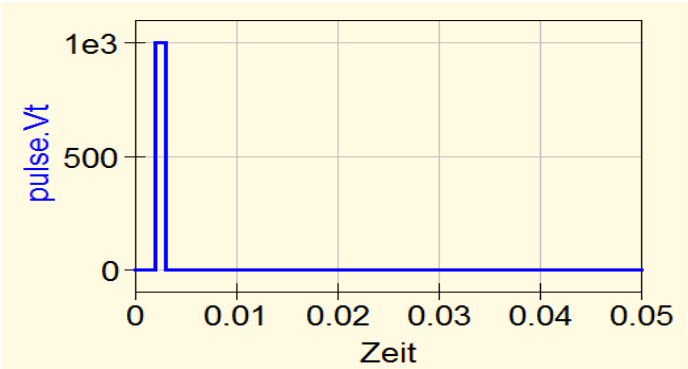
Nach der Simulation benötigen wir ein kartesisches Diagramm und füllen dessen „Eigenschaften-Menü“ folgendermaßen aus:

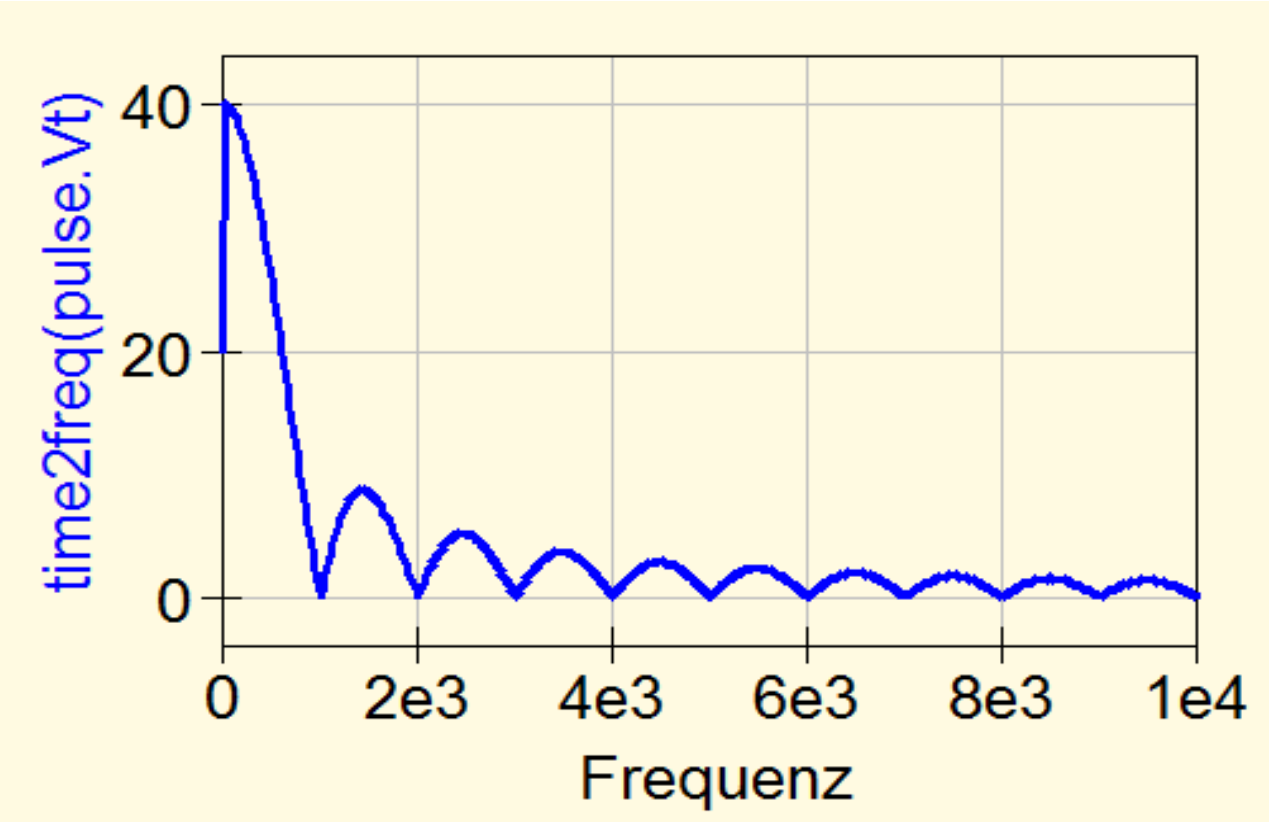
h) **pulse.vt wird aktiviert und erscheint im Graphen-Fenster.**

i) Die Liniendicke wird auf „2“ gestellt, die Farbe ist automatisch erst mal „blau“.

j) Auf der Karteikarte „Eigenschaften“ wird für die X-Achse die Bezeichnung „Zeit“ angegeben.

k) Und mit „Anwenden“ und OK sollte sich folgendes Diagramm einstellen.





Um dieses Bild zu erhalten, holen wir erst ein kartesisches Diagramm, rufen seine Eigenschaften auf und geben oben im Fenster "**time2freq(pulse.Vt)**" ein.

Mit "**neuer Graph**" wird diese Eingabe übernommen und wir sorgen gleich für eine **Linienbreite "2"**.

Auf der Karteikarte "Eigenschaften" wird die Bezeichnung "**Frequenz" für die waagrechte Achse vorgesehen.**

Unter "Grenzen" sorgen wir für eine **Frequenzanzeige von 0....10 kHz**. Dann wird "Anwenden" und OK angeklickt.

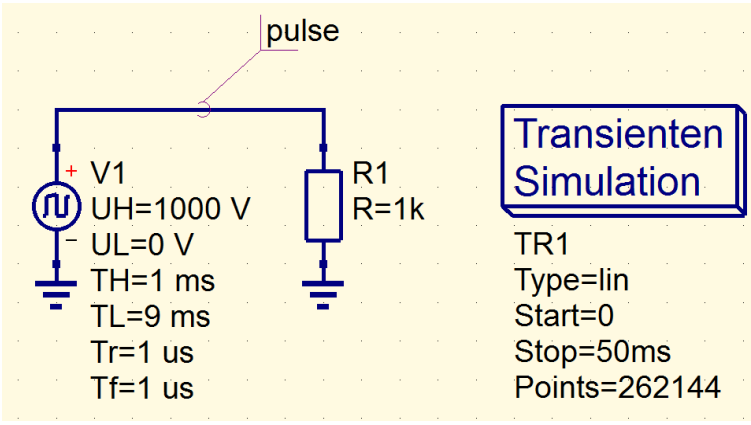
Sehr schön ist nun der "**sinx / x**" - Verlauf der Energieverteilung zu sehen. Es handelt sich ja um ein lückenloses Spektrum und damit wird im Prinzip die "**Leistungsdichte pro Hertz Bandbreite**" dargestellt.

Übrigens:
Die Theorie behauptet, dass bei **allen Vielfachen** von

f = 1 / Pulslänge

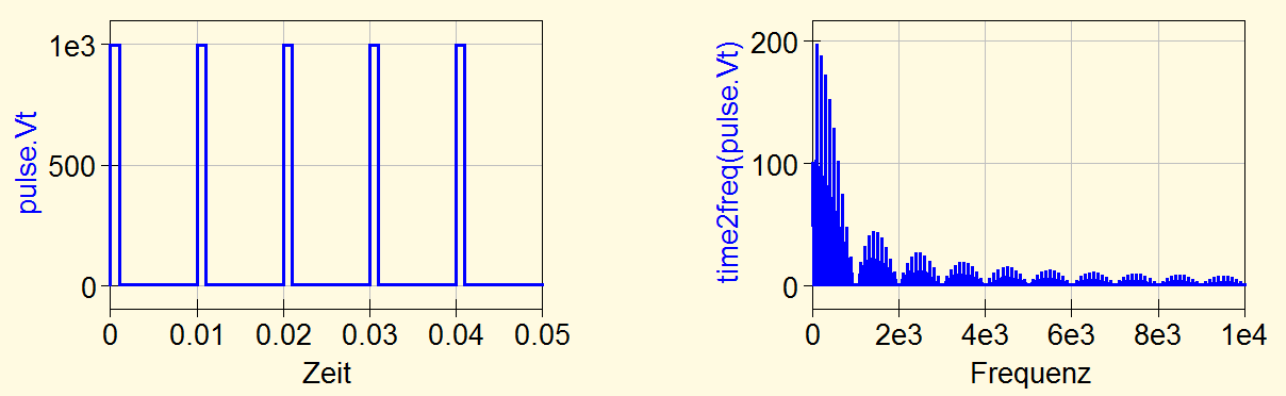
Nullstellen im Frequenzspektrum auftreten müssen. Bitte prüfen Sie das nach.

4.4. Simulation des Frequenzspektrums bei einem periodischen Pulssignal

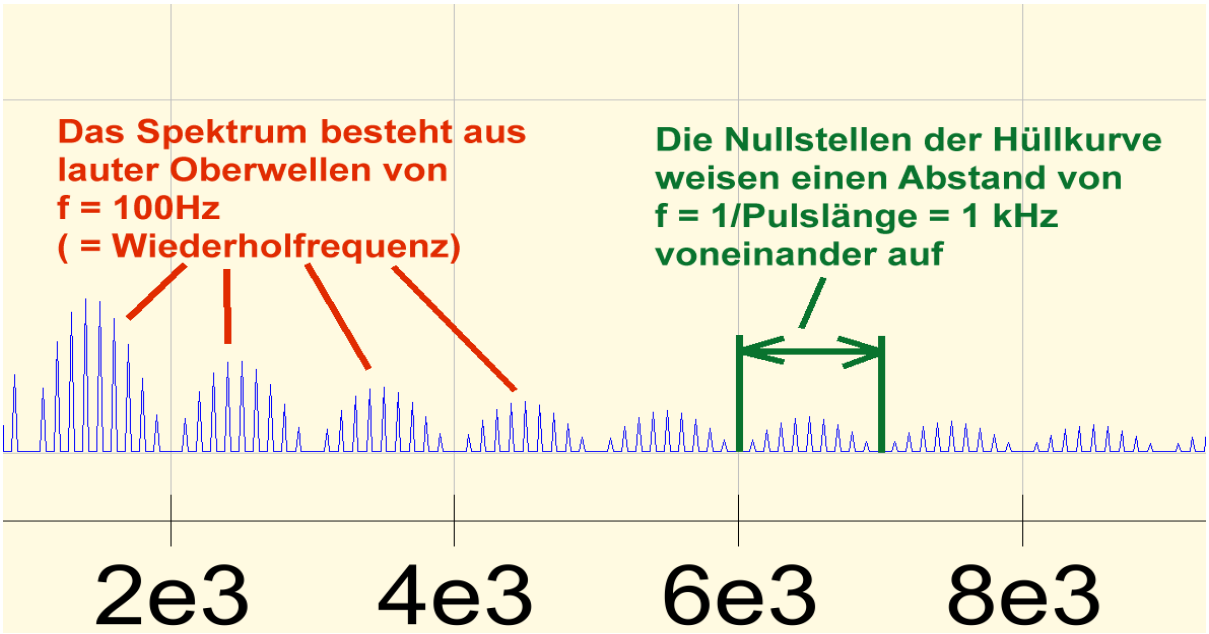


Wir benützen denselben Spannungsverlauf wie vorher, wiederholen ihn aber **100x in der Sekunde.** Deshalb müssen wir die **Dateibasierte Spannungsquelle**“ durch eine **Rechteck-Spannungsquelle** ersetzen.
Die Simulationszeit beträgt weiterhin 50ms, damit wir auf eine Frequenzauflösung von 1 / 50ms = 20Hz kommen.

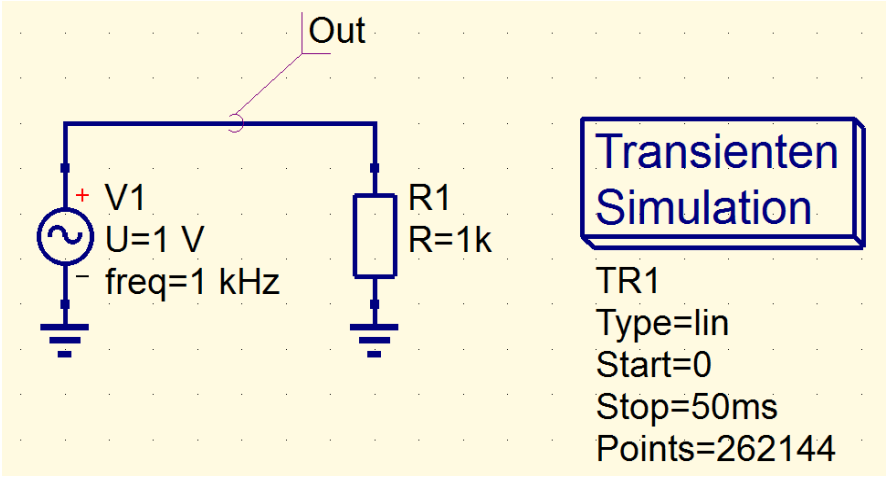
Damit erhalten wir folgenden Spannungverlauf und das zugehörige Frequenzspektrum:



Die Hüllkurve entspricht dem Verlauf im vorigen Beispiel (= nur ein einziger Puls), aber durch die Wiederholung mit f = 100 Hz verändert sich das lückenlose Spektrum des Einzelpulses zu einem Linienspektrum nur mit Harmonischen dieser Grundfrequenz f = 100 Hz. Ein Zoom des Diagramms zeigt das sehr schön.



4.5. Der ideale Sinus

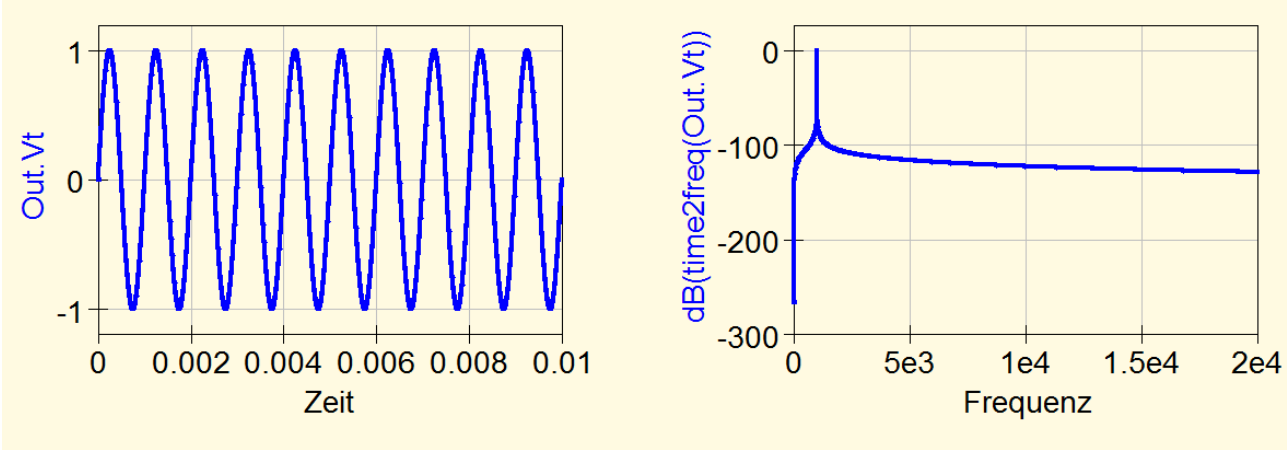


Sehen wir uns mal an, ob sich die Behauptung aus dem Grundlagenteil:

„Nur ein Sinussignal, bei dem weder Anfang noch Ende zu erkennen sind, enthält nur eine einzige Spektrallinie!“

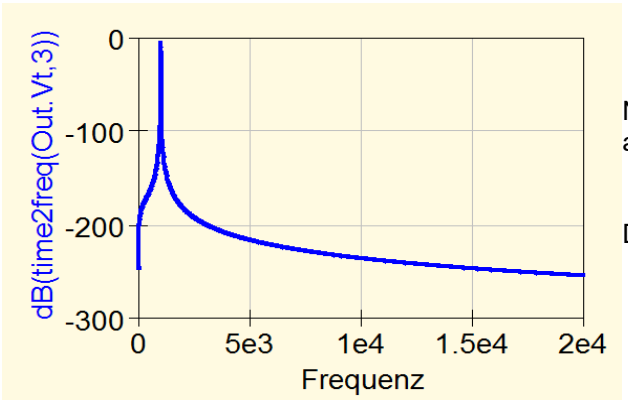
beweisen lässt.
Simulieren wir deshalb eine 1kHz-Sinusspannung (Spitzenwert = 1V) für 50ms mit 262144 Samples.

Das kommt dabei heraus:



Wer enttäuscht ist, dass es da ab -100 dB im Spektrum plötzlich nicht mehr richtig nach unten weiter geht, der bedenke:
Wir kämpfen mal wieder mit dem Problem des „**Einschalt- und Ausschaltvorganges**“ und müssen deshalb ein „**Window“ (Hanning = Index 3**) über den zeitlichen Verlauf im linken Bild legen.
Dann erhalten wir das folgende Spektrum mit der Formel:

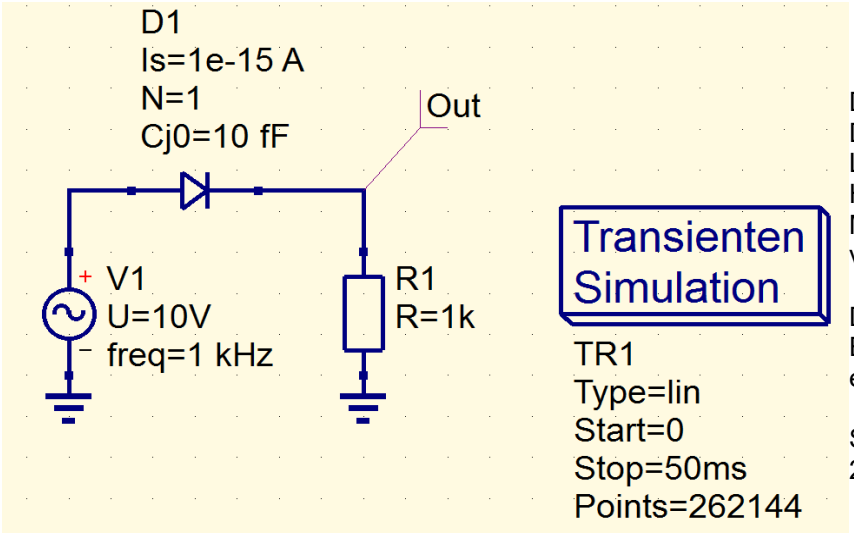
dB(time2freq(Out.Vt,3))



Na, das sieht gleich ganz anders und recht vernünftig aus.

Das ist nun wirklich ein nahezu perfekter Sinus…

4.6. Der unsymmetrisch verzerrte Sinus

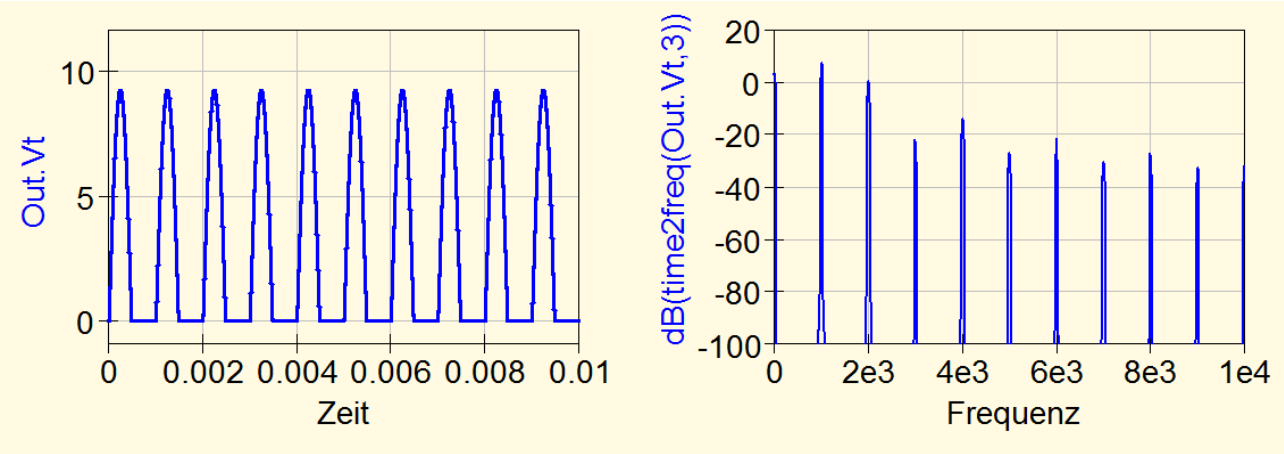


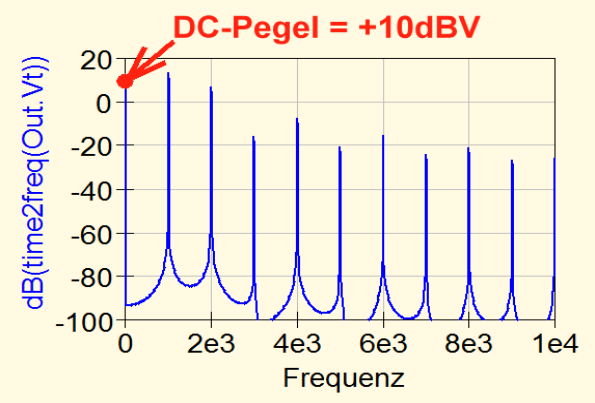
Da schneiden wir einfach durch eine Diode zwischen Quelle und Lastwiderstand die negative Halbwelle ab. Das ist die extremste Möglichkeit, eine Halbwelle zu verzerren..

Der Spitzenwert der sinusförmigen Eingangsspannung beträgt 10 V erhöht, die Frequenz f = 1 kHz.

Simuliert wird bis t = 50 ms mit 262144 Punkten.

Im linken Bild mit tmax = 10 ms sehen wir sehr schön den zeitlichen Verlauf am Ausgang. Der Wert der positiven Halbwelle ist nun – wie es sich gehört! – genau um die Schwellspannung der leitenden Diode mit 0,7 Volt kleiner als beim Eingangssignal.
Das rechte Bild ist das Ergebnis der FFT mit einem **Hanning-Fenster (Index = 3)** und reicht bis zu f = 10 kHz.





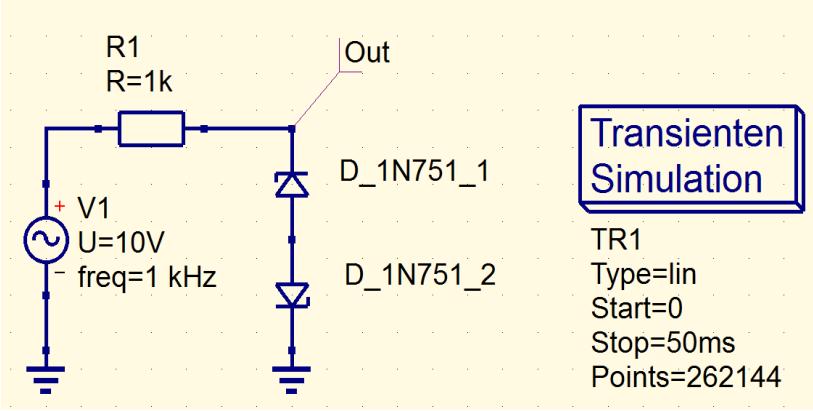
Lässt man das Hanning-Window (Index = 3) bei der FFT weg, dann erhält man dieses Ergebnis.
Im Vergleich zum obigen rechten Bild haben wir nun natürlich das „Auffüllen des Spektrums mit Energie durch den Einschalt- und Ausschaltvorgang“.
Aber: Durch den **Wegfall der „Fensterung“** mit ihrer Abschwächung der Werte im Start- und Schlussbereich wird jetzt plötzlich der **DC-Anteil korrekt dargestellt**.
+10 dBV entsprechen nämlich etwa 3 Volt und die Theorie behauptet, dass der DC-Anteil in einem solchen „gleichgerichteten Signal“ den Wert

Udc = Umax / π

hat.

Und Umax = 9,3 Volt geteilt durch π gibt etwa 3Volt. Bitte vergleichen!

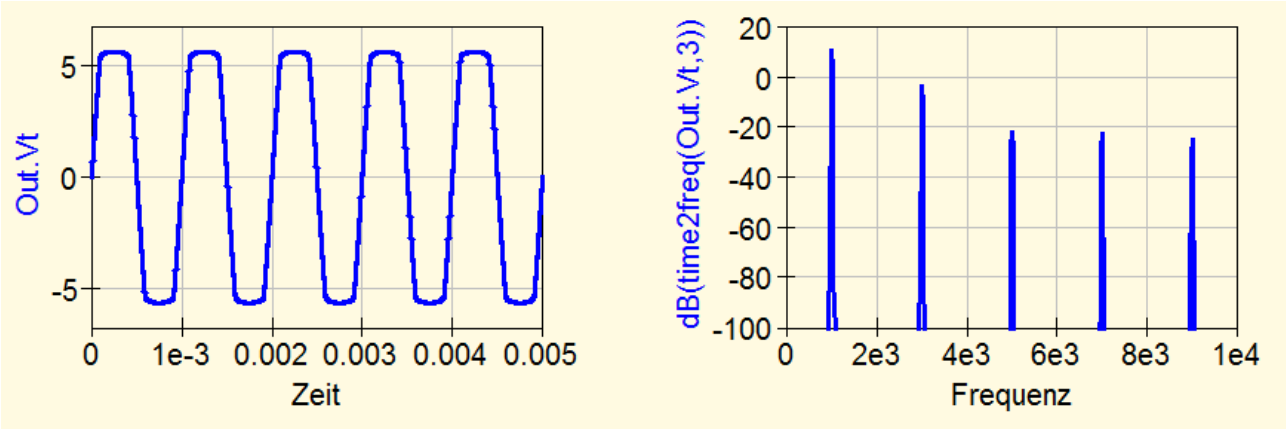
4.7. Der symmetrisch verzerrte Sinus

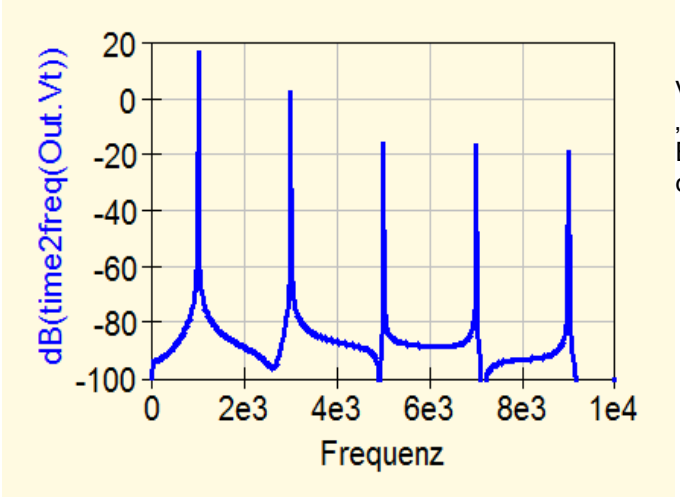


Eine sehr einfache Übung: wir schalten zwei gleiche Z-Dioden gegensinnig in Reihe und verwenden denselben Diodentyp „1N751“. Sie weisen eine Durchbruchsspannung von 5,1 V auf und wir finden sie in der untersten Karteikarte „Bibliotheken“ unter „Z-Diodes“ (= linker Rand der Zeichenfläche).

Wieder wird ein Sinus mit f = 1kHz / Upeak = 10V angelegt und bis 50ms mit 262144 Punkten simuliert.

Auch beim Spektrum gilt dieselbe Kalibrierung wie im vorigen Beispiel und man erkennt sehr schön, dass diesmal (wegen der Symmetrie des Signals) nur **ungeradzahlige Harmonische** auftreten.



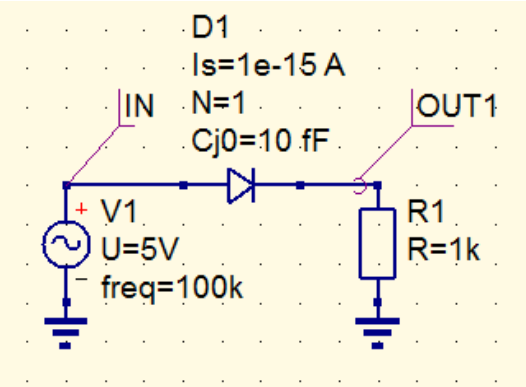


Verzichtet man bei der FFT wieder auf die „Hanning-Fensterung“, so hat man zwar erneut den Einschalt- und Ausschaltvorgang am Hals. Aber dafür stimmt die Amplitudenkalibrierung.

5. Drittes Projekt: Diodenschaltungen

5.1. Einpuls-Gleichrichter mit idealer Diode

Das ist eine einfache Sache, denn d nehmen wir



eine **Wechselspannungsquelle** (Spitzenwert = 5V, Frequenz = 100 kHz

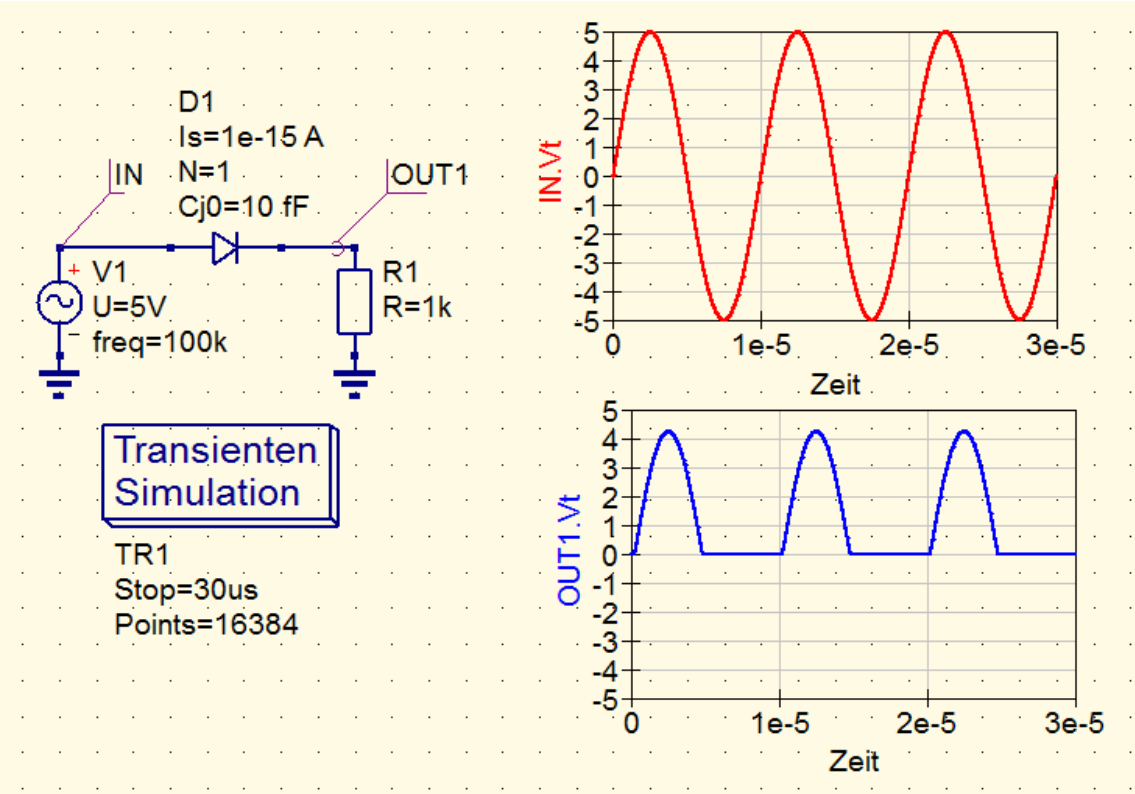
eine **ideale Diode** (aus „Komponenten / nicht-lineare Komponenten)

und einen **Lastwiderstand mit R = 1kΩ**.

Eingang und Ausgang werden mit **Labels** (= „IN“ und „OUT1“) versehen.

(Allgemeine Erinnerung:
wollen wir irgendwelche Eigenschaften -- sei es bei einem Bauteil, sei es bei einer Quelle oder bei einem Diagramm -- ändern, dann brauchen wir einen rechter Mausklick auf das Objekt, um an das Menü „Eigenschaften editieren“ heran zu kommen).

Simuliert wird für **30 Mikrosekunden** mit einer **Auflösung von 16384 Punkten** (….bitte möglichst gleich eine Zweierpotenz bei der Punktezahl verwenden, damit wir -- wenn wir auf die Idee kommen sollten -- gleich die Vorbedingungen für eine korrekte FFT erfüllen).
Nach der Simulation brauchen wir zwei kartesische Diagramme und stellen die beiden Signale in der Schaltung untereinander dar:



5.2. Verwendung von Dioden aus dem Bauteil-Vorrat

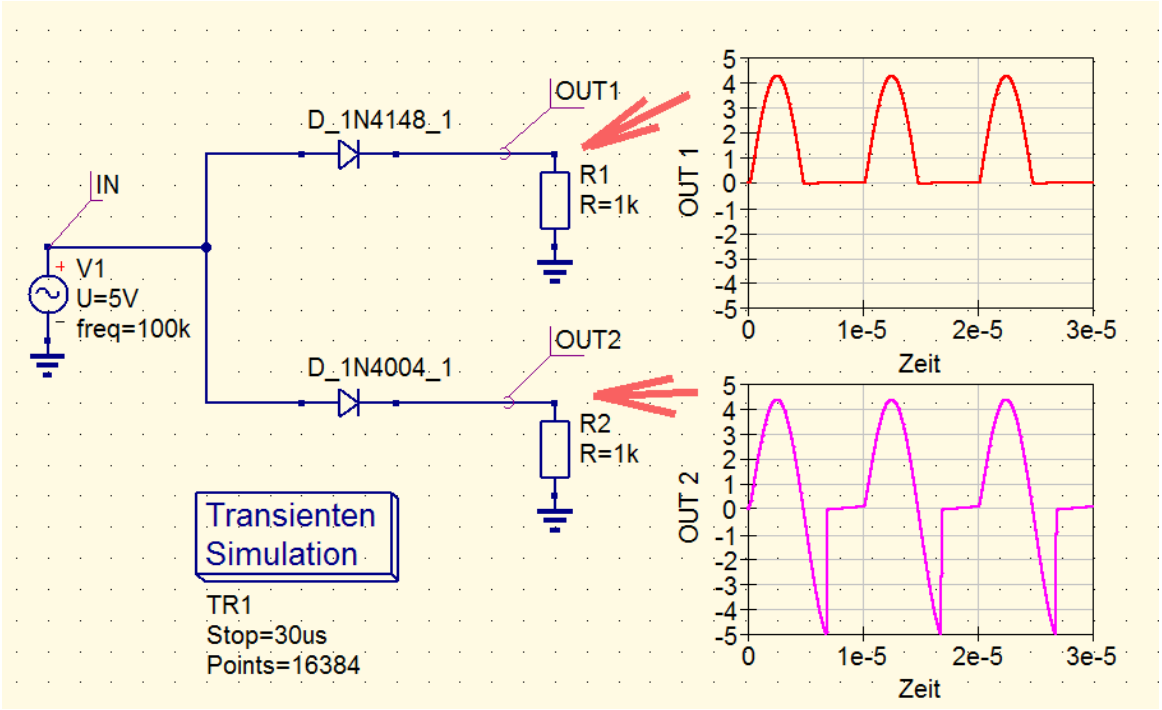
Wir greifen auf zwei sehr bekannte Bauteile zurück:

a) auf die bekannte schnelle Universal-Kleinleistungsdiode 1N4148 und

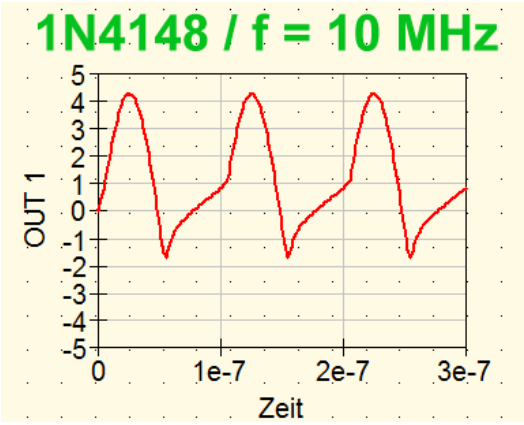
b) auf den bekannten Netzgleichrichter 1N4004

Beide finden wir im am linken Bildrand angeordneten Menü „Bibliothek“, wobei das Symbol zum „Ziehen des Bauteils in die Schaltung (= drag’n drop) ganz unten links zu finden ist.

Die Simulationseinstellungen lassen wir unverändert. Nach der Simulation erstellen wir zwei kartesische Diagramme für die beiden Ausgänge. Aber die beiden Ergebnisse werden nacheinander in die Zwischenablage kopiert und neben dem Schaltplan abgesetzt. Das gibt folgendes Bild:



Man erkennt bei der 1N4004 sehr gut den Effekt der „Sperrverzögerung“, der einen Einsatz dieser Hochlastdiode bei höheren Frequenzen unmöglich macht.



Bei der 1N4148 deutet sich dieser Effekt ganz behutsam an und erst bei f = 10 MHz kommt man langsam an die Grenze.

Radikale Abhilfe bringt da nur die Verwendung einer superschnellen und kapazitätsarmen Schottky-Diode.

Wir machen uns deshalb im nächsten Kapitel auf die Suche nach dem SPICE-File der bekannten BAT17 und sehen uns an, wie so etwas für den Einsatz in „qucsstudio“ zurechtgeschnitzt wird.

5.3. Eine wichtige Sache: Verwendung von SPICE Modellen aus dem Internet

Das ist leider wahr:
Qucsstudio kann nicht direkt mit SPICE-Modellen aus einer Bibliothek oder aus dem Internet arbeiten. Sie müssen zuerst in ein eigenes qucs-Format konvertiert werden!

Das ist manchmal nicht so einfach und kann schnell Probleme verursachen. Deshalb folgt nun der sichere Weg und er wird an der Schottky-Diode BAT17 demonstriert.

Schritt 1:
Bitte „**BAT17 spice model**“ in den Google eingeben und zur **NXP homepage** in der Ergebnisliste gehen. Dort findet man eine Datei „**BAT17.zip**“. Die wird ausgepackt und das Ergebnis ist eine neue Datei mit dem Namen „**BAT17.prm**“.

Schritt 2:
Die Datei wird mit einem Texteditor geöffnet und sofort im „**.qucs“-Projektordner** (Pfad: „Benutzer / admin / qucs“) als **„BAT17.cir“** gespeichert.
Vorsicht:
Bitte zuvor den Editor auf „ANSI“ und „Alle Dateien“ einstellen! Sonst wird „*.txt“ angehängt und qucsstudio kann die Datei nicht lesen.

So sollte der Datei-Inhalt aussehen:

```
*
******************************************
*
*BAT17
*
*NXP Semiconductors
*
*Schottky barrier diode
*
*
*IFSM =       @ tp = s
*VF   = 600mV @ IF = 10mA
**
*Package pinning does not match Spice model pinning.
*Package: SOT23
*
*Package Pin 1: Anode
*Package Pin 2: not connected
*Package Pin 3: Cathode
*
*
*Simulator:
*
******************************************
*#
.SUBCKT BAT17 1 2
*
* The resistor R1 does not reflect
* a physical device.  Instead it
* improves modeling in the reverse
* mode of operation.
*
R1 1 2 6E+07
D1 1 2 BAT17
*
```

```
.MODEL BAT17 D
+ IS = 1.419E-09
+ N = 1.022
+ BV = 6
+ IBV = 2.45E-06
+ RS = 5.112
+ CJO = 7.867E-13
+ VJ = 0.1043
+ M = 0.1439
+ FC = 0.5
+ TT = 0
+ EG = 0.69
+ XTI = 2
.ENDS
```

Bitte gleich alle Kommentarzeilen (….sie beginnen mit einem Sternchen...) herauslöschen und die Datei in die folgende Form bringen.
Bitte anschließend genau prüfen, ob sich keine Fehler eingeschlichen haben!

Erste lauffähige Modell-Version:

==========

Zweite lauffähige Modell-Version:
In der Google – Ergebnisliste find sich auch eine Siemens – Infineon library mit einem BAT17 Modell.

```
**********************************************************************
*SRC=BAT17s;BAT17s;Diodes;Schottky;Siemens
.MODEL BAT17s D(IS=4n RS=3.3 N=1.03 XTI=2 EG=.65
+ CJO=415f M=.156 VJ=.115 FC=.5 BV=12 IBV=5U TT=25p)
* Chip parameters; for higher frequencies package must be added
*******
```

Bitte alle Zeilen löschen, die mit einem Sternchen beginnen (= Kommentarzeilen). Dann die Datei in folgende Form bringen und als **„BAT17_Infineon.cir“ im „.qucs“ - Projektordner speichern.**

```
.MODEL BAT17s D
+ IS=4n RS=3.3 N=1.03 XTI=2 EG=.65 CJO=415f M=.156 VJ=.115 FC=.5 BV=12 IBV=5U TT=25p
```

Aber beim „Modell-Teil“ genau aufpassen:

a) Die erste Zeile mit „.MODEL BAT17s D“ muss wieder mit „Enter“ = end of line beendet werden-
b) Die zweite Zeile mit den Modellparametern muss mit „+ „ beginnen und alle Parameter enthalten. Auch sie wird mit „Enter“ = end of lines beendet.
==========

Dritte lauffähige Modell-Version:
Dazu verwenden wir nochmals dasselbe Siemens-Infineon – File. Aber es muss anschließend so aussehen:

```
.MODEL BAT17s D IS=4n RS=3.3 N=1.03 XTI=2 EG=.65 CJO=415f M=.156 VJ=.115 FC=.5 BV=12 IBV=5U TT=25p
```

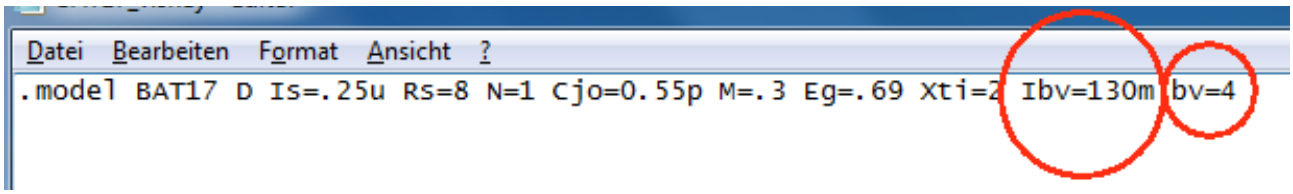
Nun wird ALLES in eine einzige Zeile gepackt und diese Zeile mit „Enter“ = end of file abgeschlossen. Anschließen diese Datei als „BAT17_infineon_02.cir“ im „.qucs“ - Projektordner speichern.

Vierte lauffähige Modell-Version:
Im Internet findet sich auch folgende Vishay - BAT17 -Datei:

.model BAT17 D(Is=.25u Rs=8 N=1 Cjo=0.55p M=.3 Eg=.69 Xti=2 Iave=130m Vpk=4 mfg=Vishay type=Schottky)

Die müssen wir natürlich für eine Verwendung in qucsstudio umstricken. Dabei gibt es zwei Möglichkeiten:

Erste Lösung:



Wir schreiben ALLES in eine einzige Zeile, die wir mit „Enter“ = end of line abschließen und entfernen auch die Klammer.
Außerdem löschen wir die beiden letzten Einträge (= „mfg=Vishay“ und „type=schottky). Zusätzlich **müssen die beiden letzten Modellparameter umbenannt werden:**

„Iave=130m“ muss jetzt „Ibv=130m“ heißen

„Vpk=4“ muss jetzt „bv=4“ heißen (….“bv“ bedeutet: breakdown voltage...)
...
Das Ergebnis wird als „**BAT17_Vishay_01.cir**“ gespeichert.

Zweite Lösung:

.model BAT17 D
+ Is=.25u Rs=8 N=1 Cjo=0.55p M=.3 Eg=.69 Xti=2 Ibv=130m bv=4

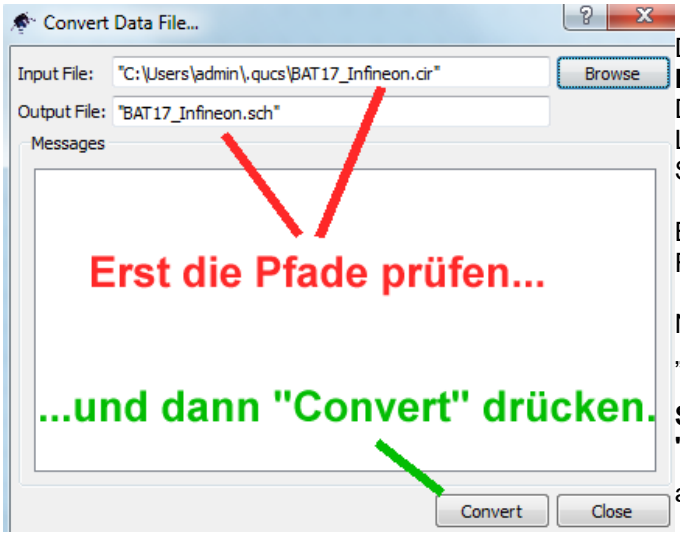
Änderungen:
a) Die Einträge für „type“ und „manufacturer“ werden **gelöscht**

b) „Iave=130m“ wird in „**Ibv=130m**“, „Vpk=4“ dagegen in „**bv=4**“ umbenannt

c) **„.model BAT17 D“ kommt in eine eigene Zeile, die mit „+ „ beginnt und die mit „Enter“ = end of line abgeschlossen wird.**

Gespeichert wird das Ergebnis als „**BAT17_Vishay_02.cir**“
==========

Schritt 4:
Jetzt kommt der Test mit der Datei „**BAT17_Infineon.cir**“.



Das Menü „**Project“** wird geöffnet und „**Import Data**“ gewählt.
Dann wählen wir „**BAT17_Infineon.cir“** aus der Liste aus.
So sollte der Bildschirm jetzt aussehen.

Bitte nun die Pfade für das Input- und das Output File prüfen und hinterher „Convert“ drücken.

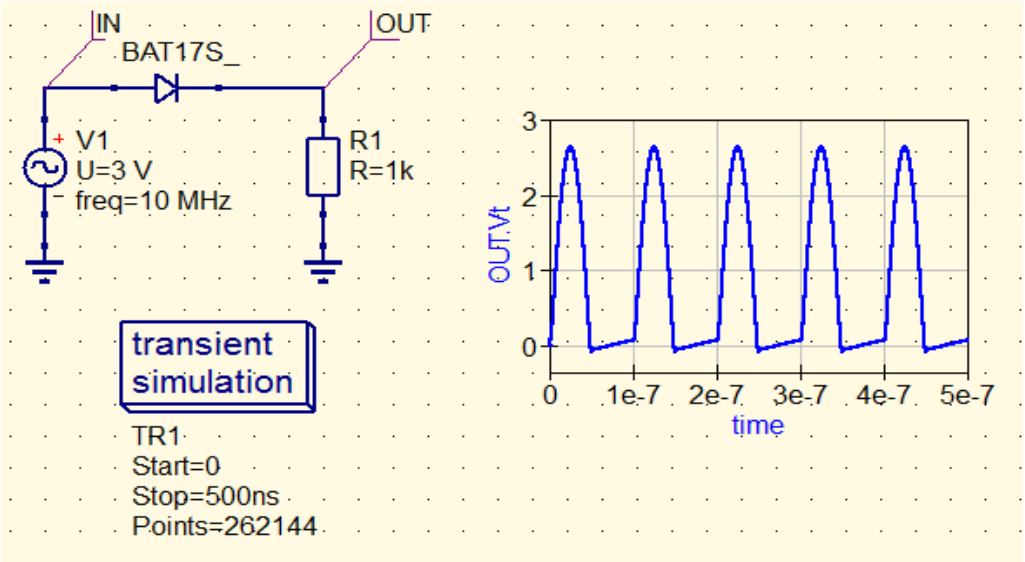
Nach der Konversion gibt es ein neues Schaltbild „**BAT17_Infineon.sch“** und eine Erfolgsmeldung

Successfully converted file 'BAT17_Infineon.sch'.

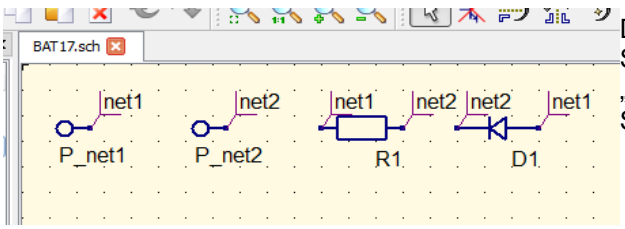
auf dem Schirm.

Jetzt dieses neue Schaltbild „**BAT17_Infineon.sch**“ öffnen, die folgende Schaltung zeichnen und anschließend simulieren**.**

Achtung:
Der Spitzenwert der Eingangsspannung muss auf 3V reduziert werden, da die Diode nur eine Sperrspannung von -4V aushält!

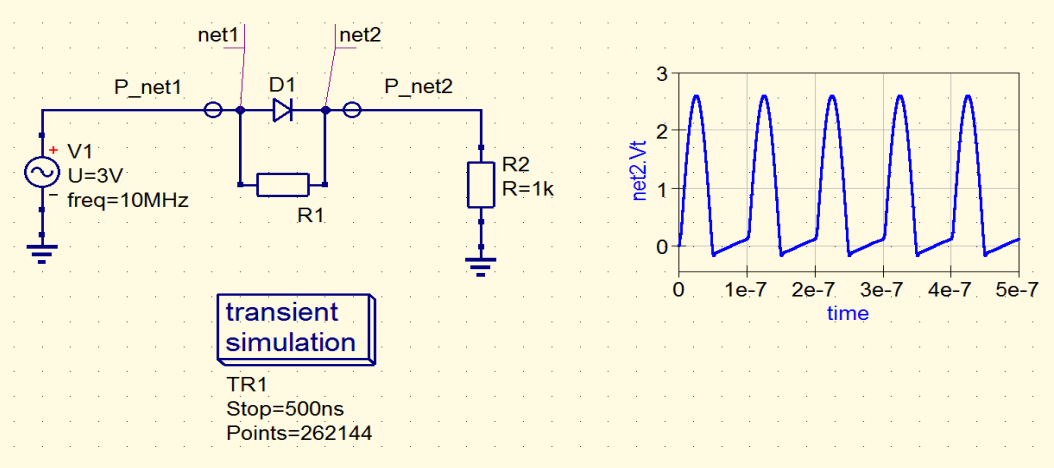


Spätestens jetzt weiß man, warum man bei hohen Frequenzen nur noch Schottky – Dioden einsetzen kann.
Zum Abschluss wollen wir noch das korrigierte **NXP-File „BAT17.cir“** einsetzen. Das ist interessant, weil hier ein zusätzlicher Widerstand für eine höhere Simulationsgenauigkeit sorgen soll.



Das liefert uns der Konverter als Ergebnis. Dieses Schaltbild trägt wieder automatisch den Namen „**BAT17.sch**“ und das müssen wir erneut zur kompletten Simulations-Schaltung ergänzen.

So sieht dann die komplette Simulation aus:



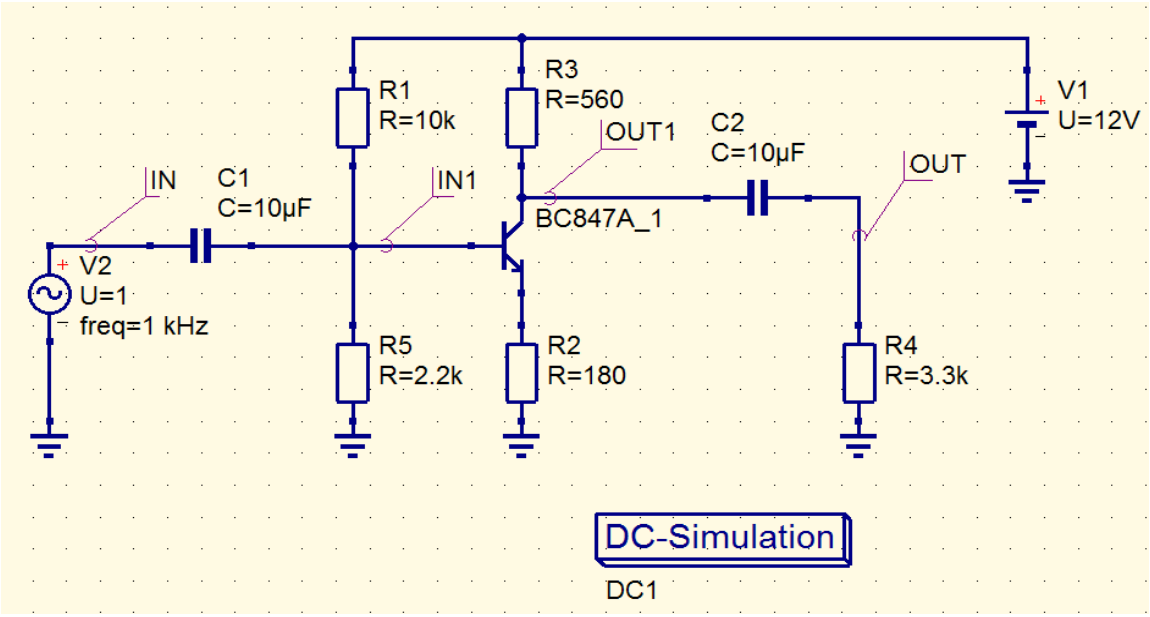
Das Ergebnis entspricht dem vorigen Beispiel.

Aufgabe:
Testen Sie mit dieser Schaltung alle restlichen erzeugten Modelle auf korrekte Funktion.

6 Untersuchung einer Verstärkerstufe mit Transistor

.

6.1. Arbeitspunktbestimmung über eine DC-Analyse

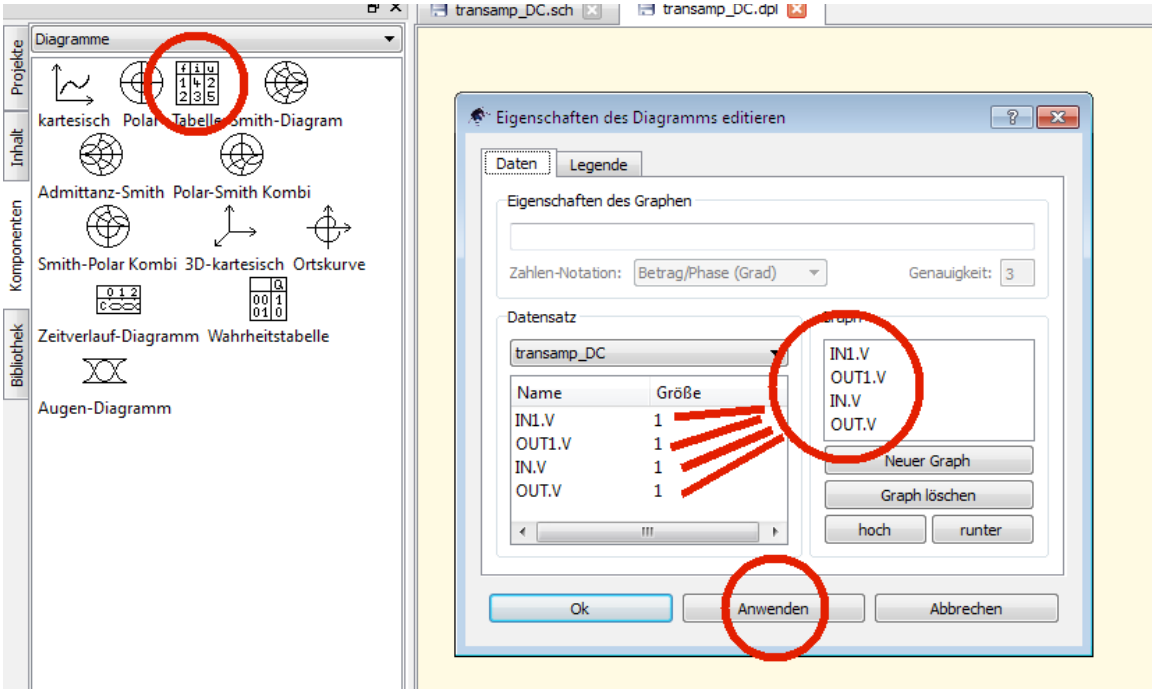


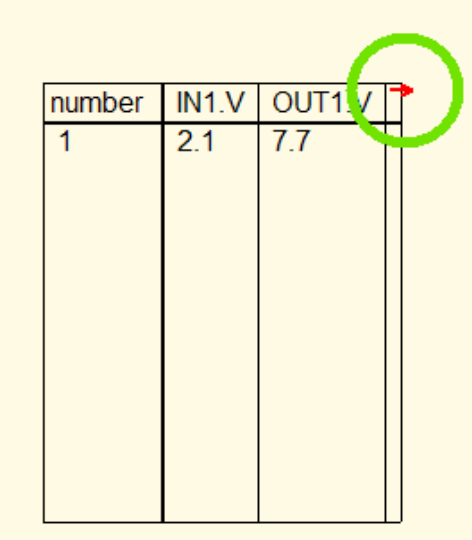
Wir zeichnen diese Schaltung und fügen gleich die „DC Simulation“ (aus „Komponenten / Simulationen) dazu. Bitte die 4 Labels nicht vergessen… Das Speichern der Schaltung sollte man nicht vergessen und gleich vornehmen.

Nach erfolgreicher Simulation ziehen wir uns eine **Tabelle** auf die Arbeitsfläche, holen die vier Variablen

IN.Vt
IN1.Vt
OUT1.Vt
OUT.Vt

durch Mausklicks in die Graphenliste und wählen „**Anwenden“**.





Das Ergebnis ist etwas unbefriedigend, denn die Tabelle zeigt nur die Hälfte der Ergebnisse.

Aber der Trick ist natürlich der kleine rote Pfeil in der rechten oberen Ecke.

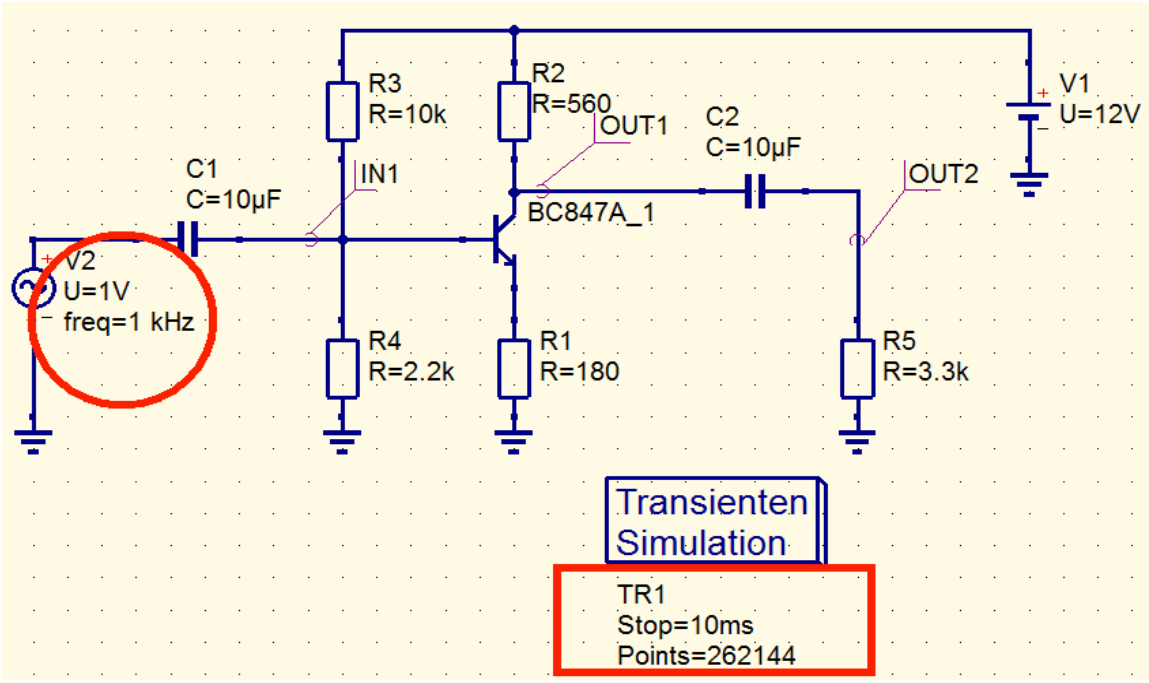
Also ziehen wir zuerst mit der Maus (bei gedrückter linker Maustaste) einen Rahmen um die Tabelle und markieren sie auf diese Weise. Der Erfolg ist ein dicker grauer Tabellenrahmen.

Jetzt fährt man auf den roten Pfeil, drückt die linke Maustaste und hält sie fest. Rollt man nun die Maus nach rechts, dann folgt automatisch die Rahmengröße der Mausbewegung.

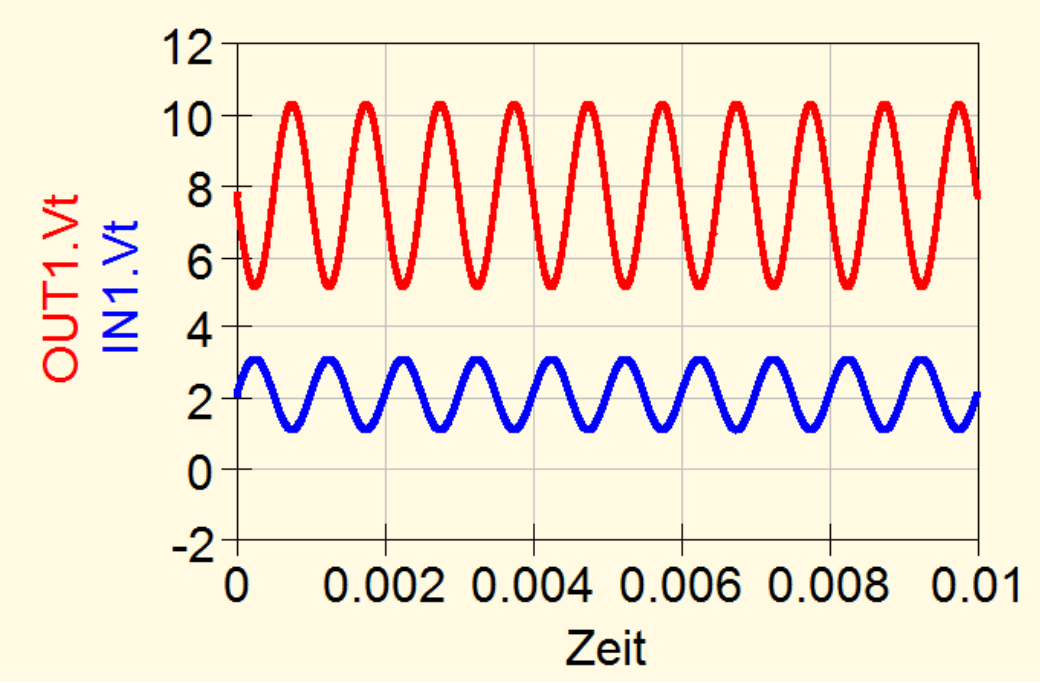
So sieht das schon besser aus.

number	IN1.V	OUT1.V	IN.V	OUT.V
1	2.1	7.7	0	0

6.2. Transienten-Simulation: Ansteuerung mit einem Sinus-Signal



Dazu steuern wir die Schaltung mit einer sinusförmigen Spannung (Spitzenwert = 1V / Frequenz = 1 kHz) an, simulieren für t = 10ms und wählen 262 144 Punkte (Erinnerung: es sollte gleich eine Zweierpotenz zur erfolgreichen FFT-Simulation sein).

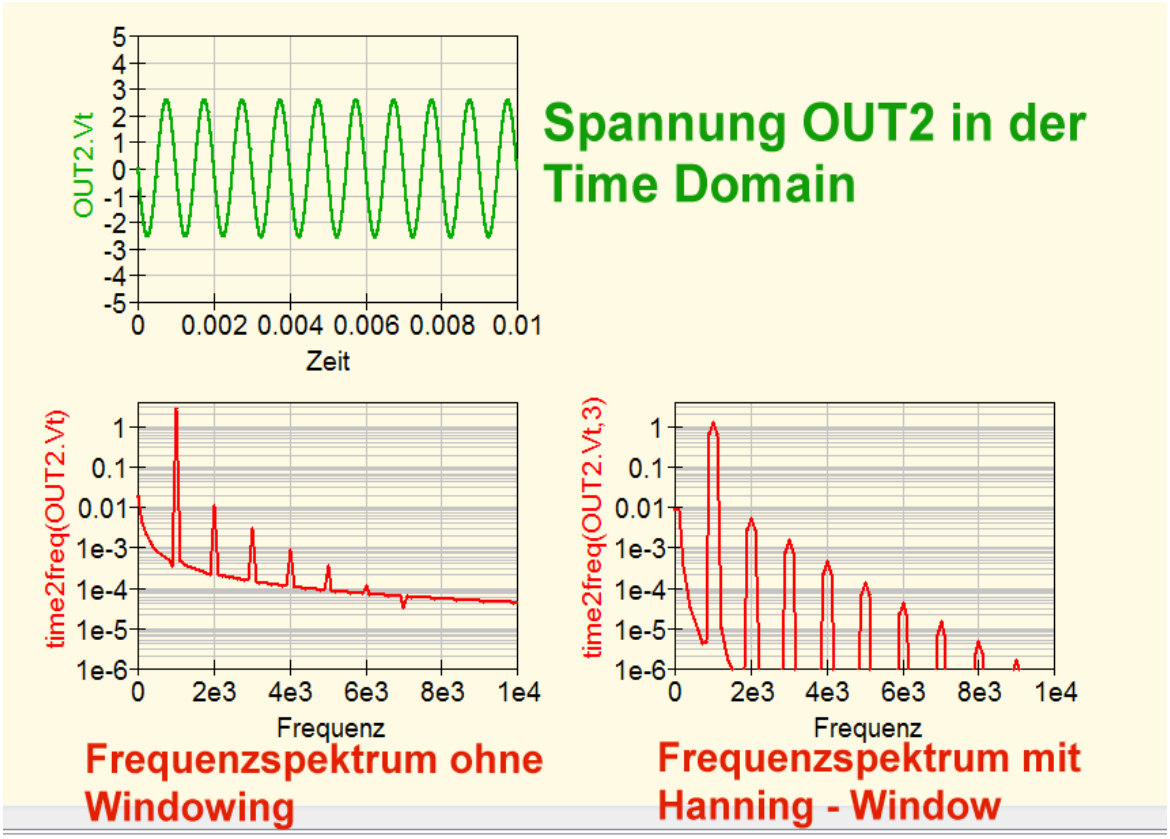


Die Auswertung der beiden Signale „IN1“ und „OUT1“ bietet wenige Überraschungen:

Sie sind (wie von der Theorie her verlangt) gegenphasig und auch der Arbeitspunkt ist für eine große Aussteuerung richtig gewählt.

Nun nehmen wir uns die Ausgangsspannung am Lastwiderstand genauer vor. Zuerst kommt der zeitliche Verlauf in der Time Domain. Dann folgen 2 Frequenzspektren dieses Signals, wobei das rechte einer **„Hanning-Fensterung“** unterzogen wurde. Die zugehörige Anweisung für den Graphen lautet:

time2freq(OUT2.Vt,3)

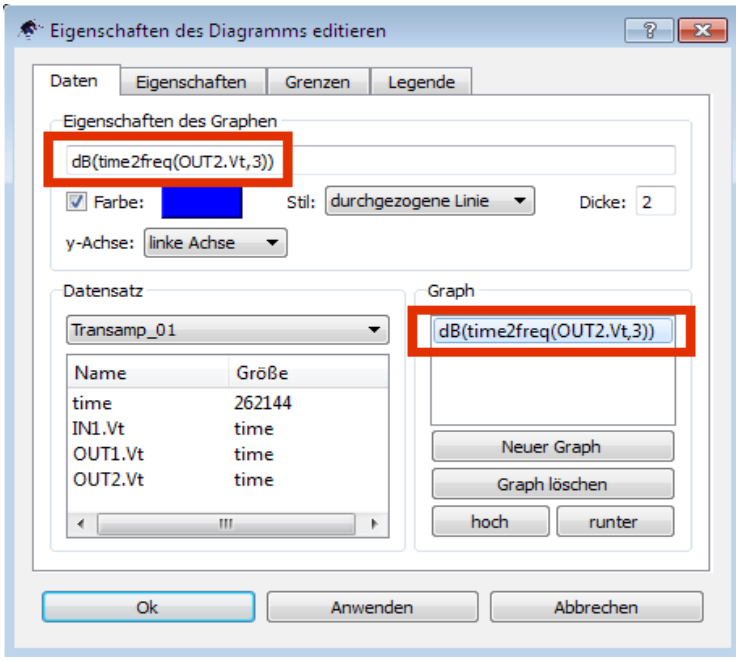


Man sieht im linken unteren Diagramm noch deutlich die Folgen des Einschalt- und Startvorganges der Schaltung (…sie führen zu einem mit steigender Frequenz abnehmenden „Grundrauschen“). Das wird natürlich im rechten unteren Bild durch die Fensterung stark reduziert, aber durch die jetzt verminderte wirksame Sample-Zahl wird sofort die Frequenzauflösung schlechter. Dadurch werden alle Linien breiter.

Übrigens:
Der an der senkrechten Achse dargestellte Amplitudenbereich reicht von 1e-6 bis 4.
Das sind über 120 dB…..
Also dürfen wir uns selbst noch eine besondere Freude machen und programmieren ein **Diagramm mit Ergebnisausgabe in dB**

6.3. Spektraldarstellung von OUT2 in dB

Das hatten wir ja schon genau im FFT-Kapitel durchgenommen – es sollte also keine Mühe machen. Deshalb nur eine kurze Wiederholung und Zusammenfassung.

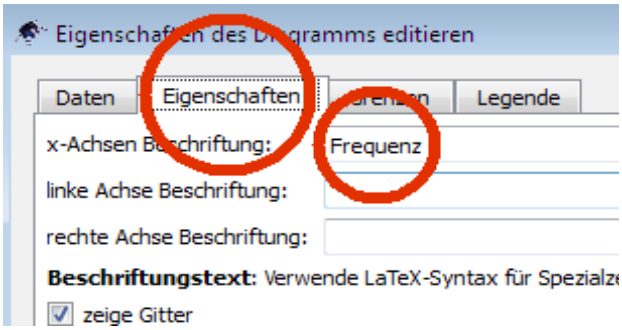


Schritt 1:
Wir holen uns ein neues kartesisches Diagramm.

Schritt 2:
Die Spektraldarstellung in dB erzielen wir durch folgende Eingabe bei der Eigenschaft des Graphen:

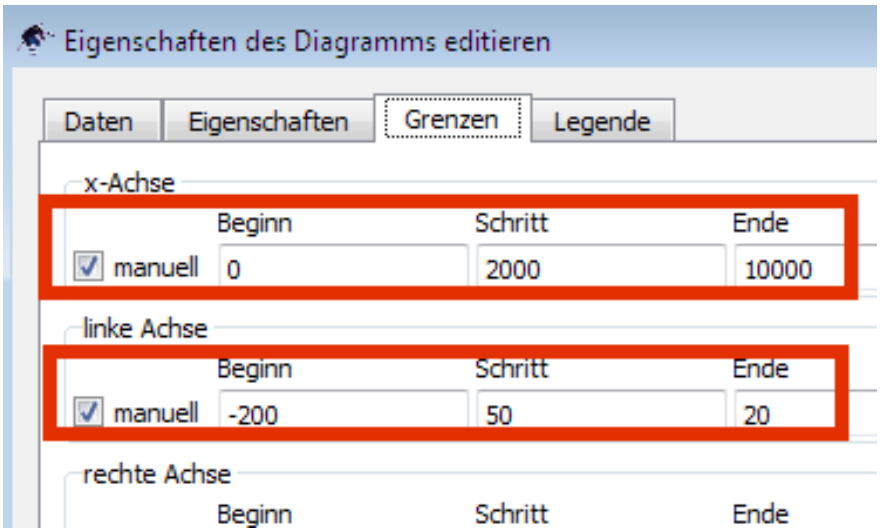
dB(time2freq(OUT2.Vt,3))

Ein Klick auf „Neuer Graph“ sorgt dann für die Übernahme in die Graphenliste.



Schritt 3:
Unter „Eigenschaften“ sehen wir „**Frequenz“** als Bezeichnung für die waagrechte Achse vor.

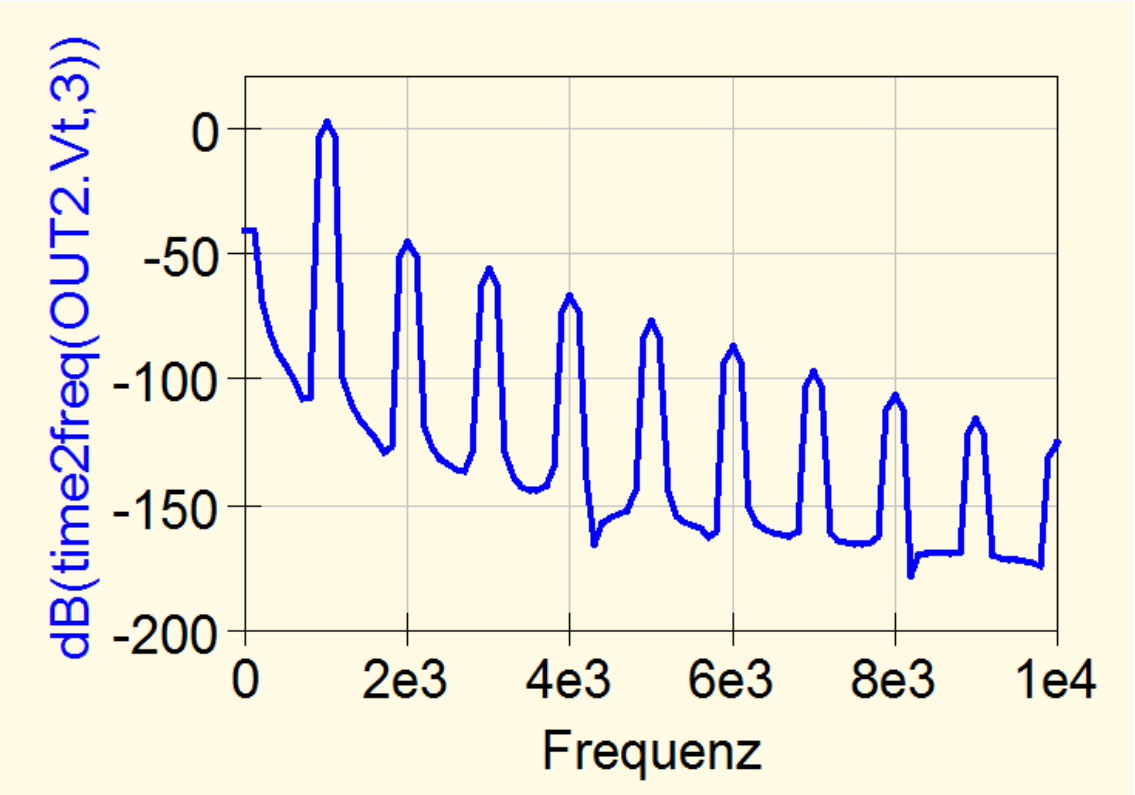
Schritt 4:
Die Eingabe der Achsenteilungen ist natürlich etwas aufwendiger.



Die x-Achse ist linear geteilt und reicht von 0 Hz bis 10000 Hz mit einem „Tick“ von 2 kHz.

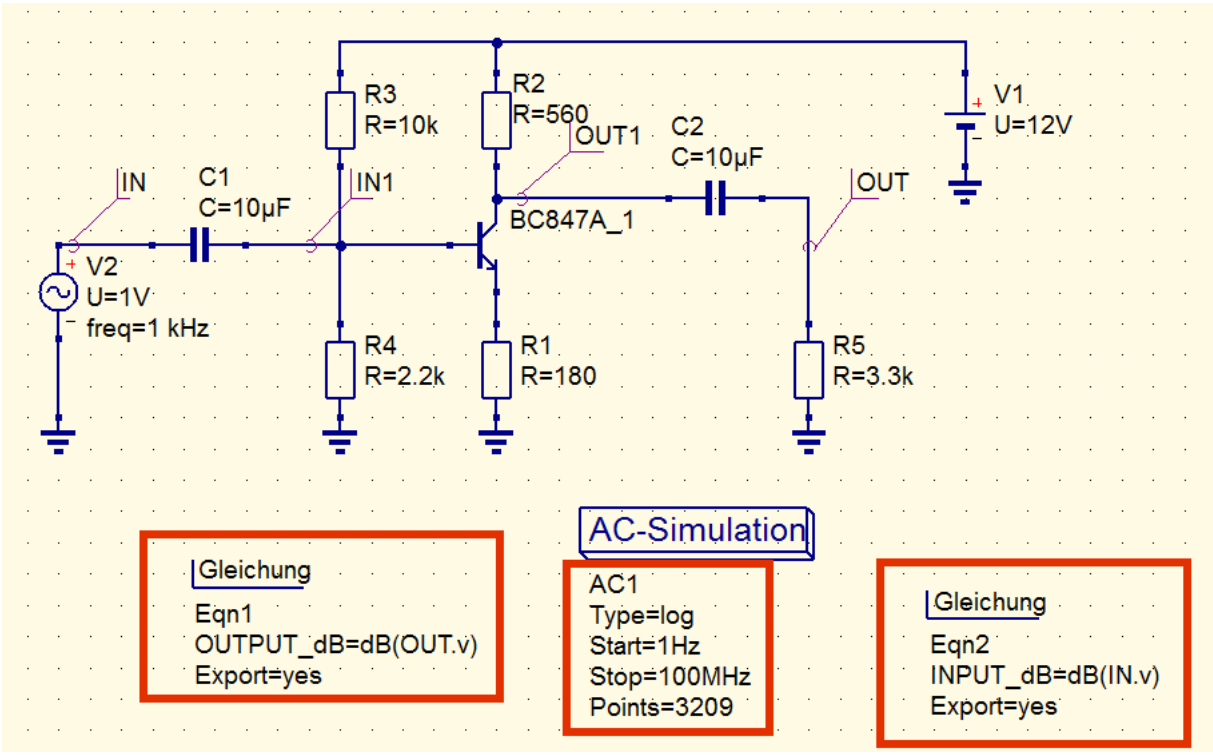
Bei der y-Achse sehen wir einen Bereich von -200 dB bis +20 dB vor und arbeiten mit einer Teilung von 50 dB.

Das sollte der Erfolg sein:

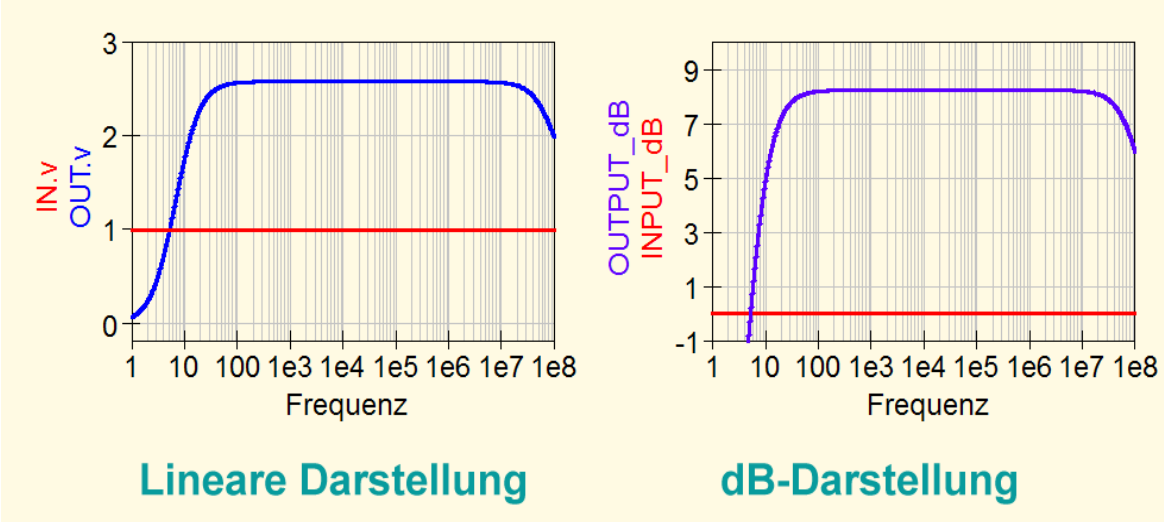


6.4. AC-Sweep zur Bestimmung der Frequenzgänge

Das ist in der Zwischenzeit ja schon eine leichtere Übung. Wir nehmen dazu einen logarithmischen AC-Sweep von 1 Hz bis 100 MHz und einer Auflösung von 401 Punkten pro Dekade.
(Die Amplitude der Eingangsspannung beträgt wieder 1 V und damit entspricht der Verlauf von „OUT“ gleichzeitig der Übertragungsfunktion).



Außer dem Sweep programmieren wir (per Gleichungen) noch zusätzlich die **Ergebnis-Ausgabe in dB** (Siehe Kapitel 3.1.2).
Das gibt nach der Simulation folgende Sammlung an Diagrammen:



6.5. Zusammenfassung: Übungs - Beispiel mit dem Transistor BFR92A

6.5.1. Beschaffung und Import des SPICE-Modells aus dem Internet

```
* Filename:  BFR92A_SPICE.PRM
* BFR92A SPICE MODEL
* PHILIPS SEMICONDUCTORS
* Date : September 1995
*
* PACKAGE : SOT23 DIE MODEL : BFR90A
* 1: COLLECTOR; 2: BASE; 3: EMITTER;
.SUBCKT BFR92A 1 2 3
Q1 6 5 7 7 BFR90A
* SOT23 parasitic model
               Lb  4 5 .4n
               Le  7 8 .83n
               L1  2 4 .35n
               L2  1 6 .17n
               L3  3 8 .35n
              Ccb  4 6 71f
              Cbe  4 8 2f
              Cce 6 8 71f
*
* PHILIPS SEMICONDUCTORS   Version:   1.0
* Filename:   BFR90A.PRM   Date: Feb 1992
*
.MODEL  BFR90A   NPN
+              IS = 4.11877E-016
+              BF = 1.02639E+002
+              NF = 9.97275E-001
+             VAF = 6.26719E+001
+             IKF = 3.20054E+000
+             ISE = 4.01062E-015
+              NE = 1.57708E+000
+              BR = 1.81086E+001
+              NR = 9.96202E-001
+             VAR = 3.36915E+000
+             IKR = 1.28155E+000
+             ISC = 2.79905E-016
+              NC = 1.07543E+000
+              RB = 1.00000E+001
+             IRB = 1.00000E-006
+             RBM = 1.00000E+001
+              RE = 1.16450E+000
+              RC = 2.32000E+000
+              EG = 1.11000E+000
+             XTI = 3.00000E+000
+             CJE = 8.90512E-013
+             VJE = 6.00000E-001
+             MJE = 2.58570E-001
+              TF = 1.54973E-011
+             XTF = 3.91402E+001
+             VTF = 2.15279E+000
+             ITF = 2.13776E-001
+             CJC = 5.46563E-013
+             VJC = 3.80824E-001
+             MJC = 2.02935E-001
.ENDS
```

Schritt 1:
Das geht natürlich wieder mit dem Start einer Suchmaschine und der Eingabe „bfr92 spice model“ los. Da landet man in der NXP-Homepage und kann sich eine gezippte Sammlung aller denkbaren Modelle für diesen Transistor herunterladen. Uns interessiert darin natürlich „BFR92A-Spice.prm“ und die sieht so aus.

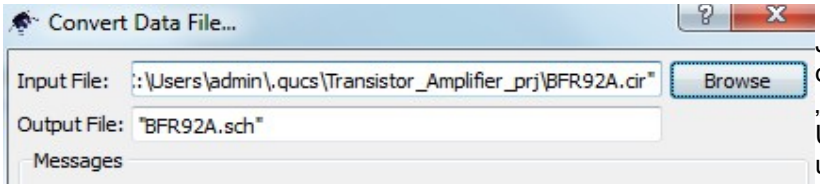
Dann legt man sich -- am besten im Verzeichnis „qucs“, in dem unsere Projekte verwaltet werden -- einen zusätzlichen **Ordner „Modelle**“ an (…dieses Projektverzeichnis findet sich auf der Festplatte c:\ unter „Benutzer“).

Gespeichert wird unser File darin als

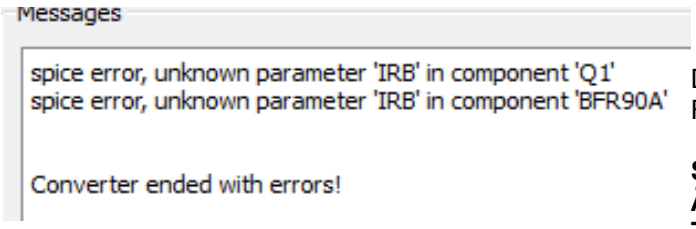
BFR92A.cir

wobei man beim Editor auf die Einstellungen „ANSI“ und „Alle Dateien“ achten muss……sonst gibt es Probleme…..

Step 2:



Jetzt starten wir qucsstudio und öffnen das Menü „**Projects**“, gefolgt von „**Import Data“**.
Über „Browse“ suchen wir nach unserem File „**BFR92A.cir**“ und starten die Konversion.



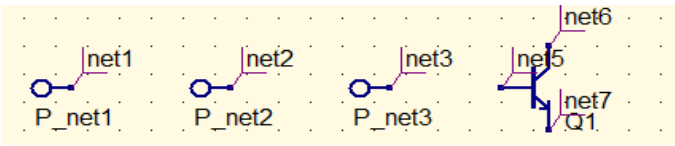
Der Konverter antwortet leider mit dieser Fehlermeldung.

Sorry:
Ältere Konverterversionen haben einfach die Tatsache ignoriert, dass IRB ein recht unwichtiges Detail des Modells ist, aber der Fortschritt.....

Deshalb öffnen wir nochmals „**BFR92A.cir**“ mit dem Texteditor, löschen die Zeile mit IRB einfach heraus und starten erneut die Konversion. Jetzt zeigt sich der Erfolg mit der Meldung

Successfully converted file „BFR92A.sch“.

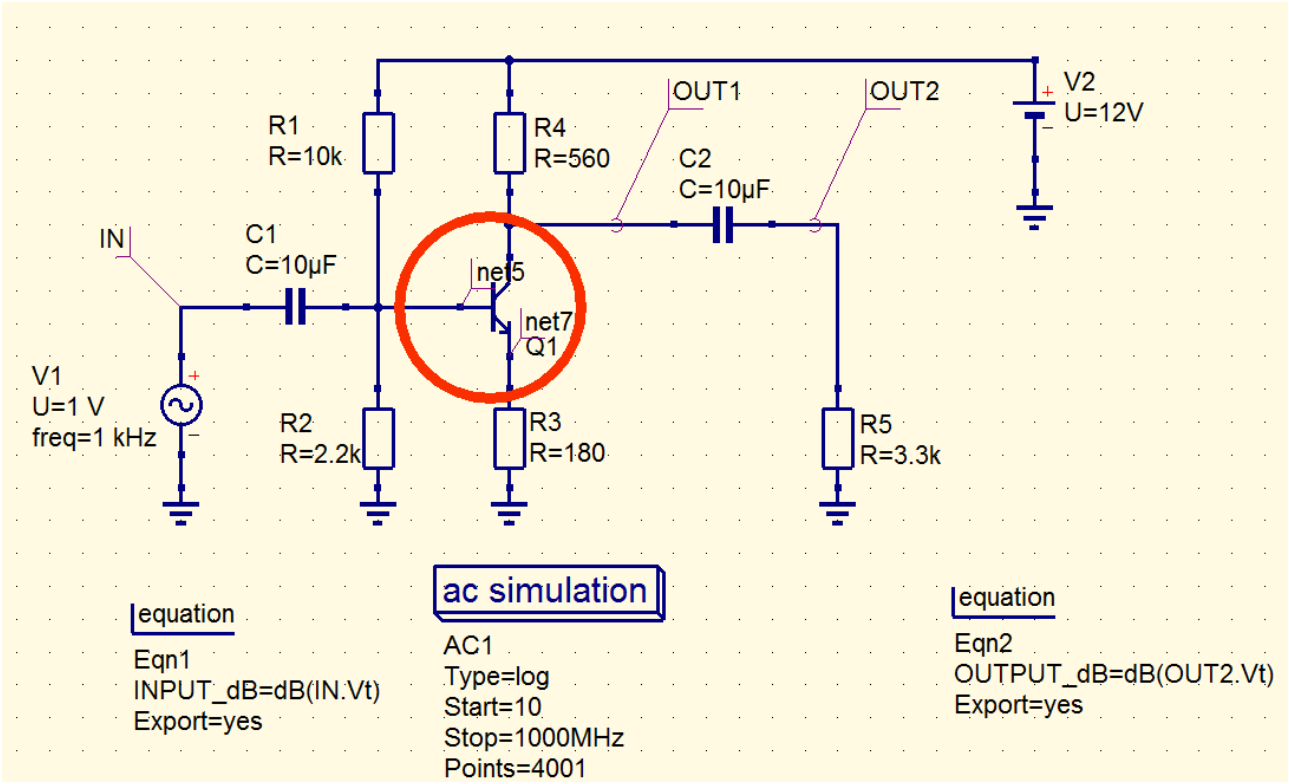
Step 3:

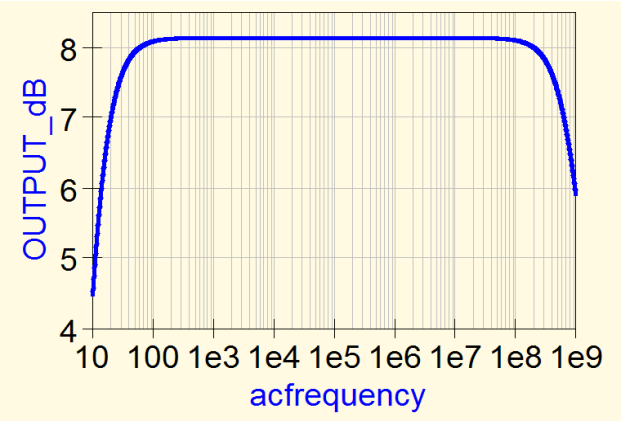


Also öffnen wir nun die vom Konverter erzeugte Datei „**BFR92A.sch**“. So sieht sie aus.

Die drei zusätzlich vom Konverter eingefügten Ports löschen wir heraus, denn es interessiert uns nur das Transistorsymbol.

Wir können nun das letzte Beispiel öffnen, das Transistorsymbol aus „BFR92A.sch“ hineinkopieren, den bisherigen Transistor BC874 herauslöschen und dafür das eben geholte Symbol an seiner Stelle einfügen.





Nach der Simulation mit den obigen Einstellungen bekommen wir dieses Bild – und das kennen wir vom letzten Beispiel her.

Aber jetzt kommt eine zusammenfassende Übungsaufgabe:
Führen Sie alle Simulationen des letzten Kapitels mal selbst aus. Das waren

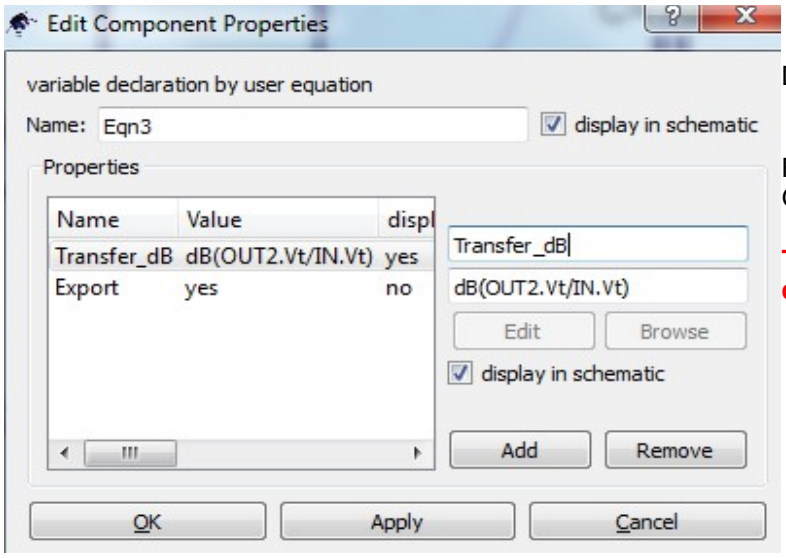
DC Simulation

Transienten – Simulation mit einem 1 kHz Sinus am Eingang. Schauen Sie sich alle Signale in der Schaltung an.

Führen Sie eine FFT des Ausgangssignals durch. Zeigen Sie die Unterschiede mit und ohne Hanning-Windowing (index = 3). Verwenden Sie sowohl eine lineare wie auch eine dB-Darstellung.

6.5.2. Ein Schlusstest

Aufgabe:
Berechnen Sie die Transfer-Funktion = (OUT2 / IN) in dB für den Bereich von 10 Hz bis 1000 MHz.



Das geht so:

Fügen Sie in Ihr Verstärkerschaltbild eine Gleichung ein. Sie lautet:

Transfer_dB= dB((OUT2.v)/(IN.v))

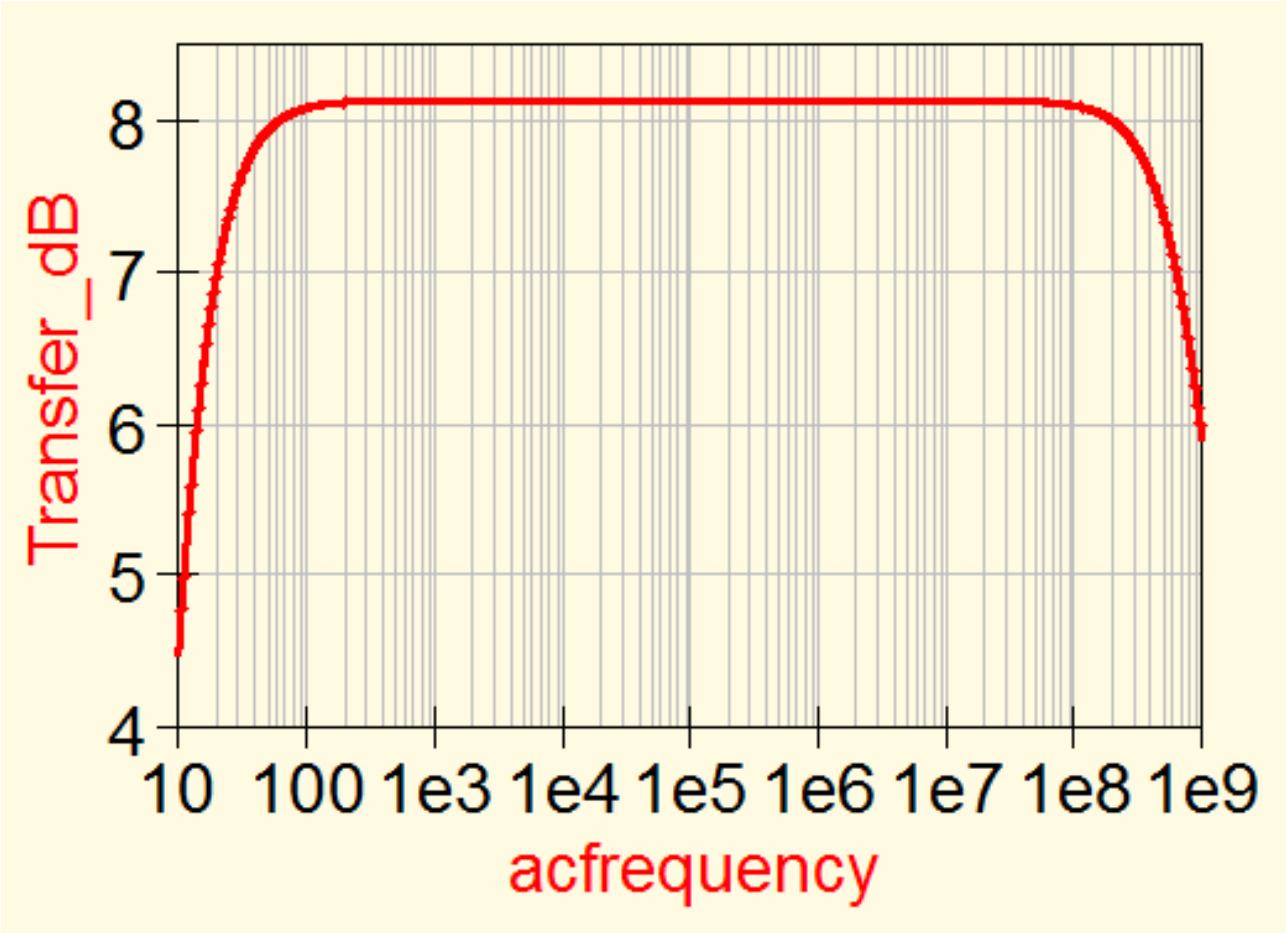
equation

Eqn3
Transfer_dB=dB(OUT2.Vt/IN.Vt)

Das muss nun neben Ihrem Schaltbild auftauchen:

Führen Sie einen logarithmischen AC-Sweep von 10 Hz bis 1000 MHz mit 4001 Punkten pro Dekade durch und wählen Sie ein kartesisches Diagramm für das Ergebnis.

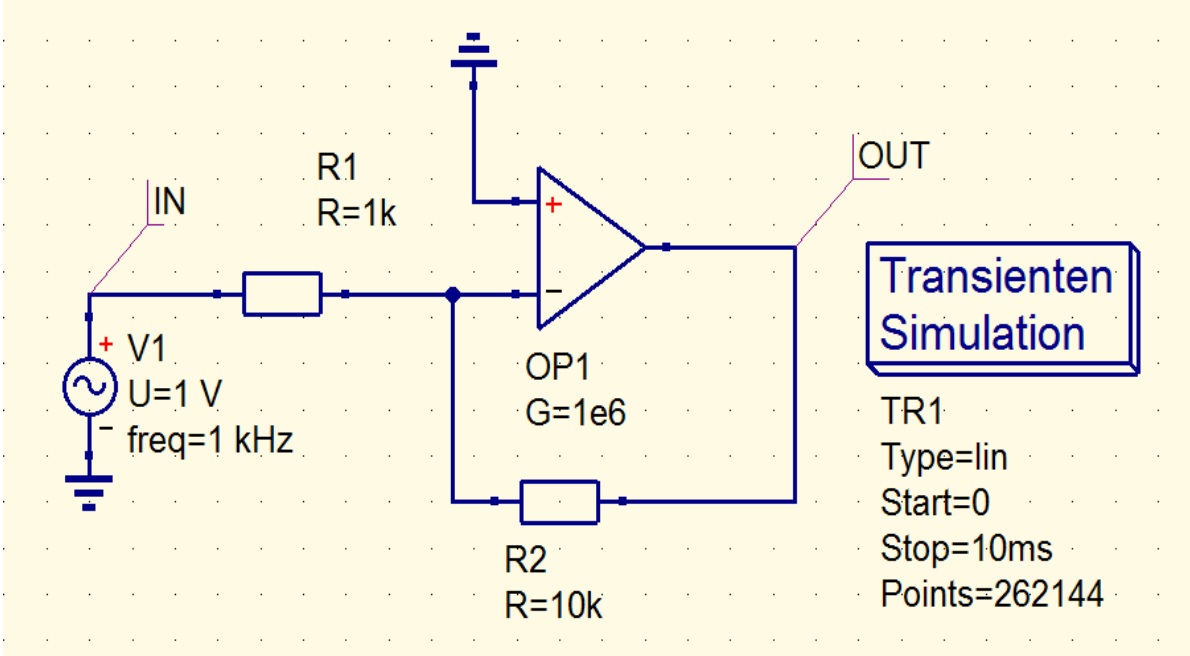
Wenn Sie die X-Achse auf „**logarithmisch**“ einstellen, sollten Sie dieses Ergebnis erhalten:



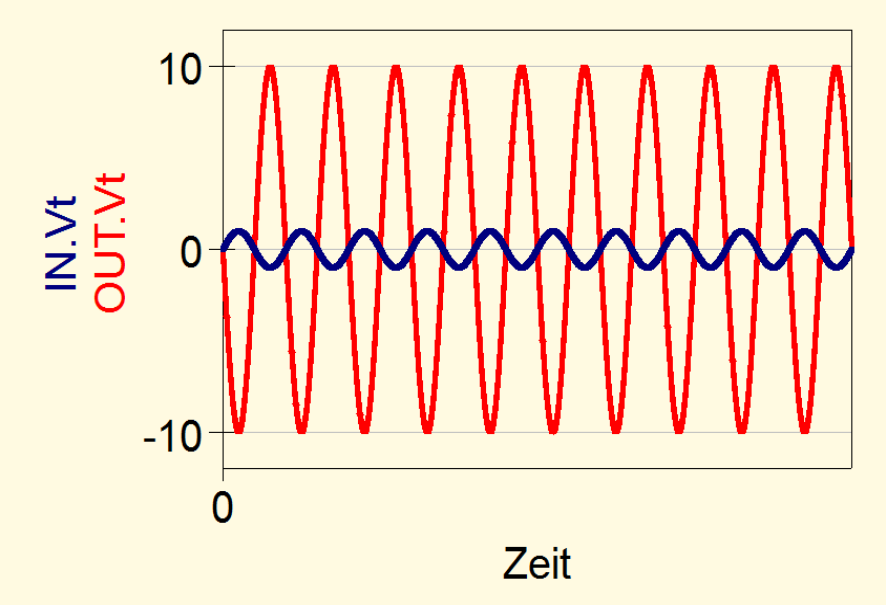
7. OPV-Schaltungen

7.1. Die umkehrende Schaltung

Kein besonderes Problem -- der ideale OPV findet sich als“ Opamp“ unter „Komponenten / Systemkomponenten.

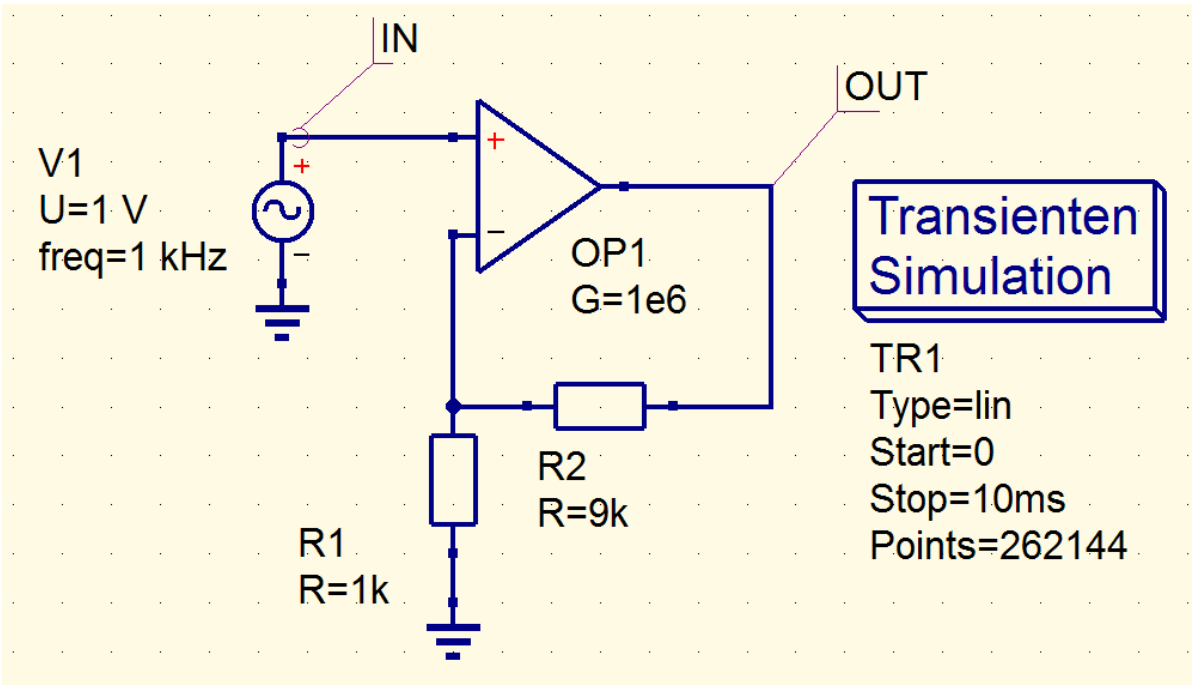


Und mit R1 = 1k bzw. R2 = 10k erhalten wir zehnfache Verstärkung einschließlich Phasenumkehr.

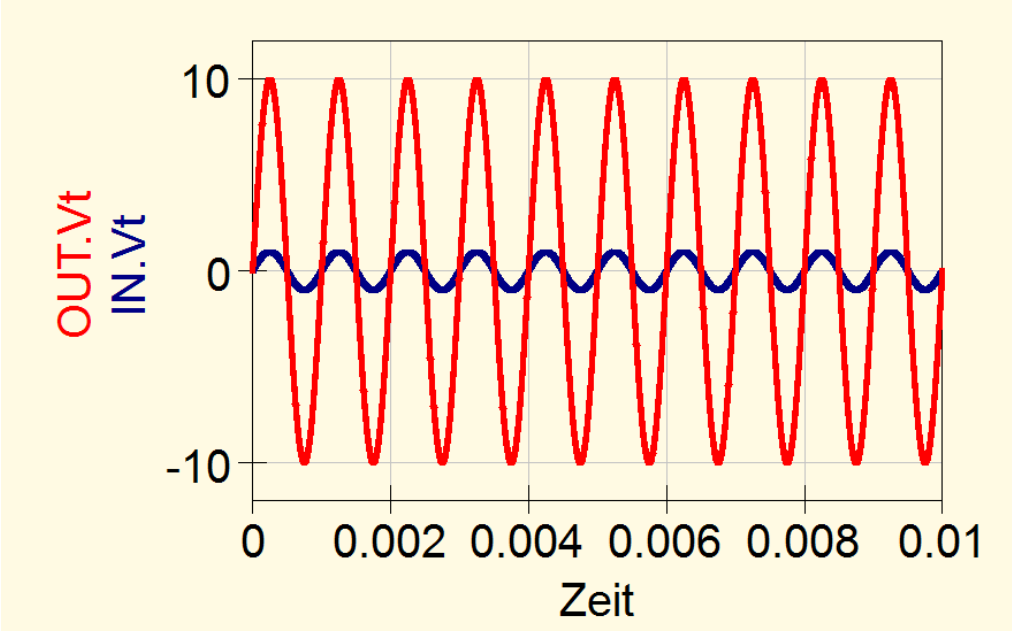


7.2. Die nicht umkehrende Schaltung

Um zehnfache Verstärkung zu erhalten, muss der Spannungsteiler (bestehend aus R1 und R2) die Ausgangsspannung auf 10% herunterteilen. Deshalb sind nun die Werte R1 = 1k und R2 = 9k nötig.



Funktioniert!



7.3. Der OPV als Analog-Addierer

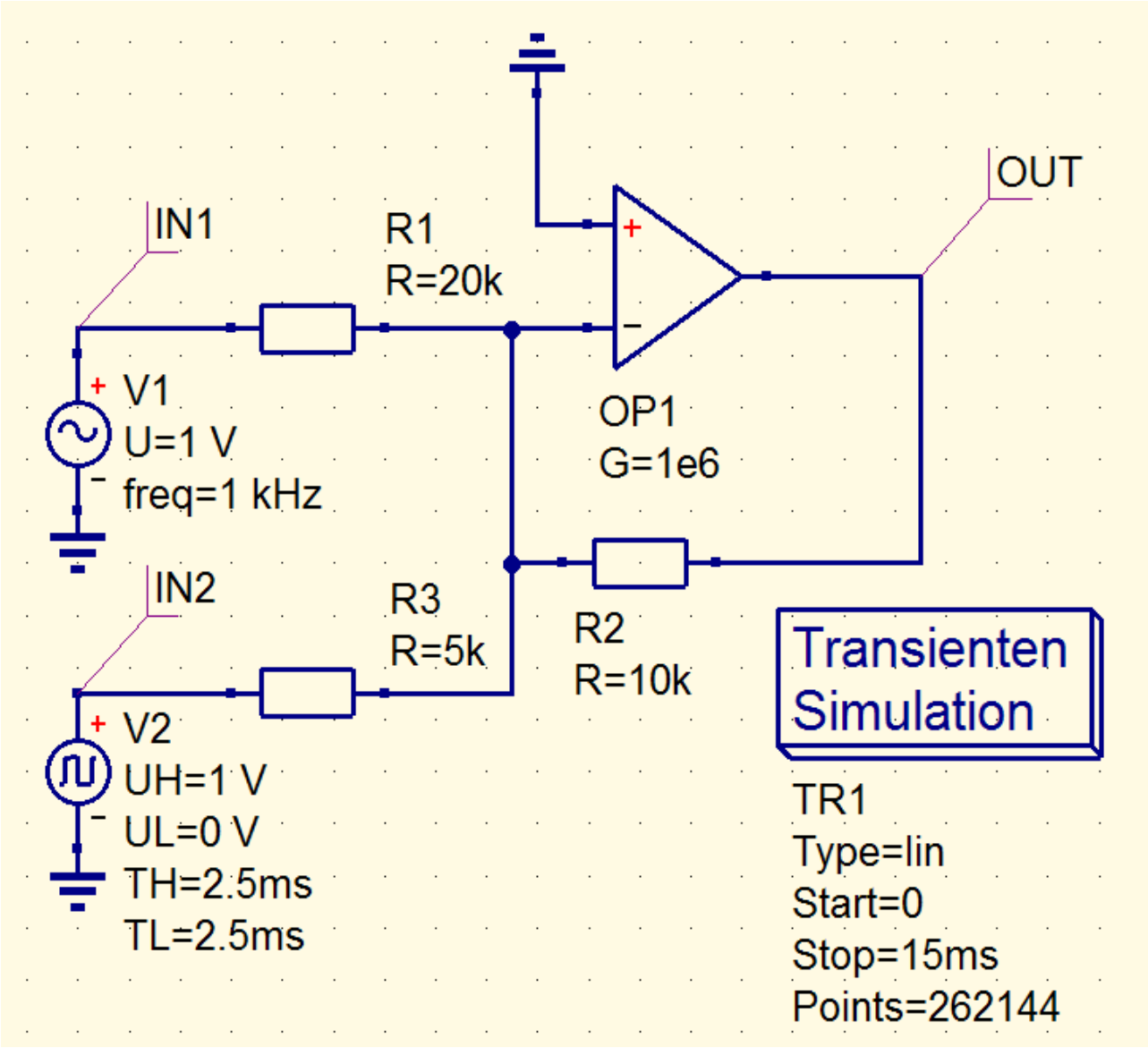
Dazu setzt man die umkehrende Schaltung ein, bei der die verschiedenen Eingangsströme (hier: über R1 und R3) in R2 aufsummiert werden. Die entstehende Gesamtspannung an R2 ergibt dann die Ausgangsspannung.

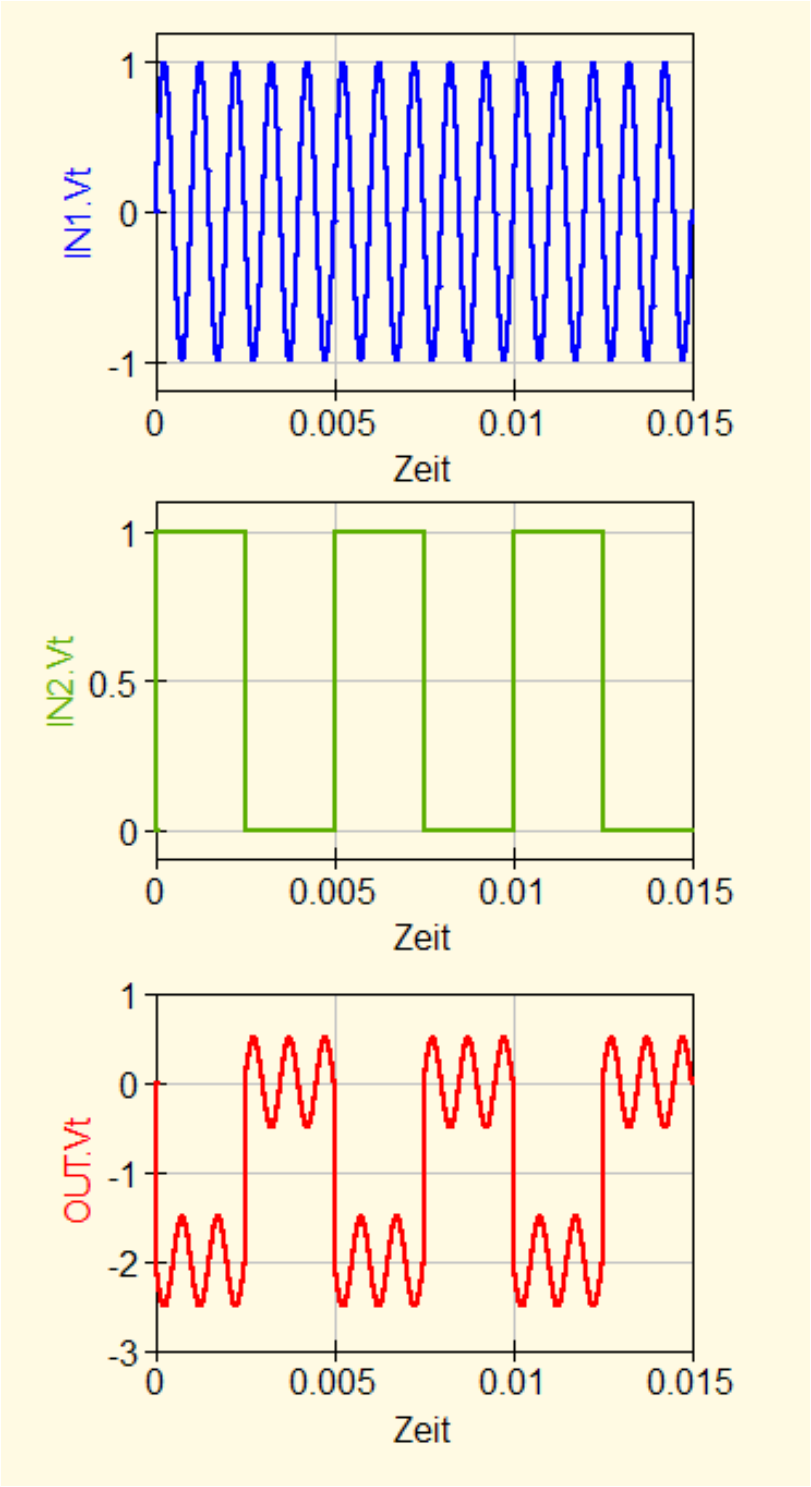
Gleichzeitig kann man unterschiedliche Verstärkungswerte für die Eingangssignale einstellen. Hier ist es

V1 = -R2 / R1 = -10k / 20k = -0,5fach für die Sinusspannung V1

und

V2 = -R2 / R3 = -10k / 5k = -2fach für die Rechteckspannung V2





Der Spitzenwert des Sinussignals an Eingang IN1 beträgt 1V

Das Rechtecksignal bei IN2 hat die Amplitudenwerte „Null“ und „+1V“

Am Ausgang muss der Rechteck-Anteil wegen der (-2)fachen Verstärkung nun die Amplitudenwerte „Null“ und „-2V“ besitzen.

Der Sinusanteil wurde auf einen Spitzenwert von 0,5 V reduziert und ist gegenphasig.

7.4. Import eines OPV-Spice-Models aus dem Internet

Das ist ein Fall, der häufig vorkommt -- schließlich lässt sich nicht jedes Problem mit dem Universal-OPV lösen (….es gibt in der Zwischenzeit ja Bausteine, die bis 500 MHz einsetzbar sind!).

Schritt 1:
Wir beschaffen uns als Beispiel das gezippte Spice-Modell des TL072 aus der Texas Instruments (= TI) Homepage, entpacken es und speichern es anschließend. So sieht es aus:

```
* TL072 OPERATIONAL AMPLIFIER "MACROMODEL" SUBCIRCUIT
* CREATED USING PARTS RELEASE 4.01 ON 06/16/89 AT 13:08
* (REV N/A)      SUPPLY VOLTAGE: +/-15V
* CONNECTIONS:   NON-INVERTING INPUT
*               | INVERTING INPUT
*               | | POSITIVE POWER SUPPLY
*               | | | NEGATIVE POWER SUPPLY
*               | | | | OUTPUT
*               | | | | |
.SUBCKT TL072    1 2 3 4 5
*
  C1   11 12 3.498E-12
  C2    6  7 15.00E-12
  DC    5 53 DX
  DE   54  5 DX
  DLP  90 91 DX
  DLN  92 90 DX
  DP    4  3 DX
  EGND 99  0 POLY(2) (3,0) (4,0) 0 .5 .5
  FB    7 99 POLY(5) VB VC VE VLP VLN 0 4.715E6 -5E6 5E6 5E6 -5E6
  GA    6  0 11 12 282.8E-6
  GCM   0  6 10 99 8.942E-9
  ISS   3 10 DC 195.0E-6
  HLIM 90  0 VLIM 1K
  J1   11  2 10 JX
  J2   12  1 10 JX
  R2    6  9 100.0E3
  RD1   4 11 3.536E3
  RD2   4 12 3.536E3
  RO1   8  5 150
  RO2   7 99 150
  RP    3  4 2.143E3
  RSS  10 99 1.026E6
  VB    9  0 DC 0
  VC    3 53 DC 2.200
  VE   54  4 DC 2.200
  VLIM  7  8 DC 0
  VLP  91  0 DC 25
  VLN   0 92 DC 25
.MODEL DX D(IS=800.0E-18)
.MODEL JX PJF(IS=15.00E-12 BETA=270.1E-6 VTO=-1)
.ENDS
```

Aber so dürfen wir es nicht für die Arbeit in qucsstudio verwenden, denn das gibt unendliche Probleme!

Wir müssen

a) Alle Zeilen, die mit einem **Sternchen beginnen, komplett herauslöschen**.

b) Alle Zeilen, in denen die Bauteile samt ihrem Wert sowie ihrer Knotenangabe eingetragen sind, beginnen hier mit einem Leerzeichen. **Das ist verboten -- also weg damit!**

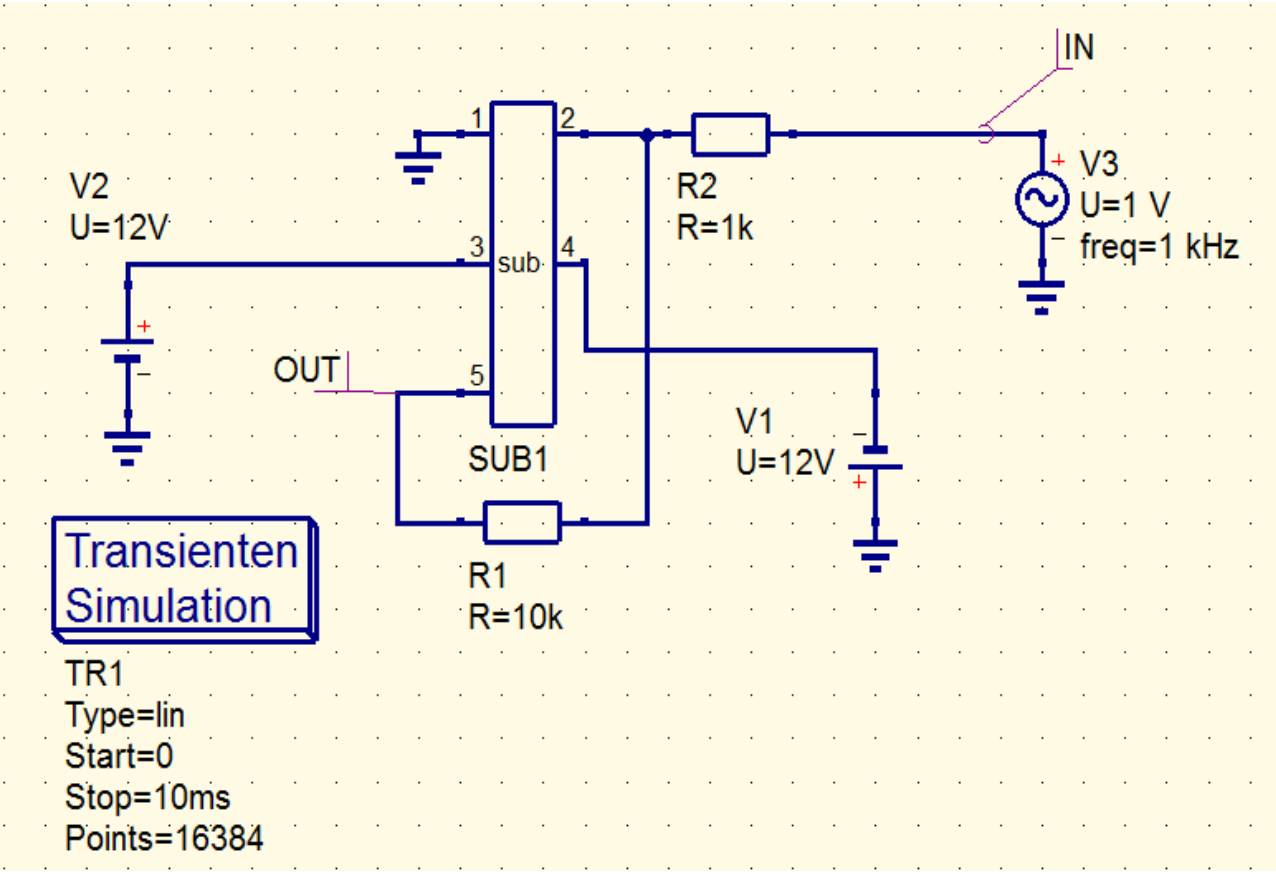
c) Und am Ende muss die Datei als „**TL072.cir**“ im vorgesehenen Projektordner gespeichert werden (..bitte wieder aufpassen und „ANSI“ sowie „Alle Dateien“ beim Editor einstellen).

So muss sie vor dem Speichern aussehen:

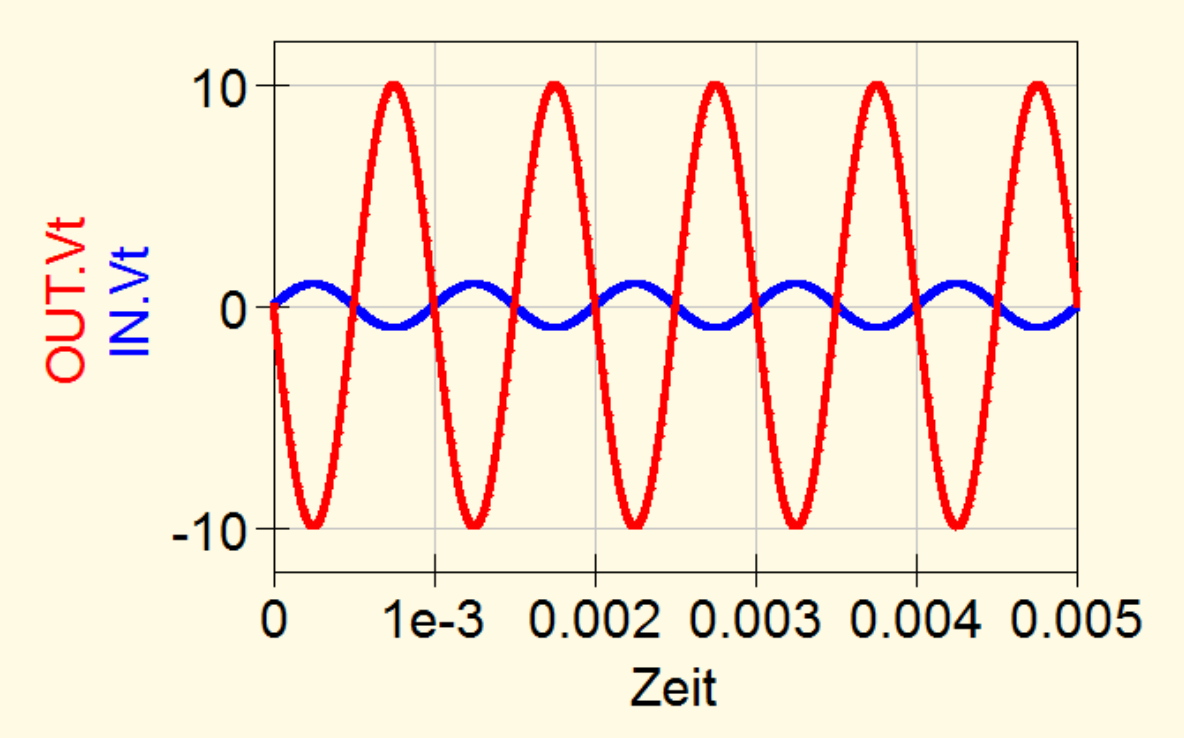
```
.SUBCKT TL072    1 2 3 4 5
C1   11 12 3.498E-12
C2    6  7 15.00E-12
DC    5 53 DX
DE   54  5 DX
DLP  90 91 DX
DLN  92 90 DX
DP    4  3 DX
EGND 99  0 POLY(2) (3,0) (4,0) 0 .5 .5
FB    7 99 POLY(5) VB VC VE VLP VLN 0 4.715E6 -5E6 5E6 5E6 -5E6
GA    6  0 11 12 282.8E-6
GCM   0  6 10 99 8.942E-9
ISS   3 10 DC 195.0E-6
HLIM 90  0 VLIM 1K
J1   11  2 10 JX
J2   12  1 10 JX
R2    6  9 100.0E3
RD1   4 11 3.536E3
RD2   4 12 3.536E3
RO1   8  5 150
RO2   7 99 150
RP    3  4 2.143E3
RSS  10 99 1.026E6
VB    9  0 DC 0
VC    3 53 DC 2.200
VE   54  4 DC 2.200
VLIM  7  8 DC 0
VLP  91  0 DC 25
VLN   0 92 DC 25
.MODEL DX D(IS=800.0E-18)
.MODEL JX PJF(IS=15.00E-12 BETA=270.1E-6 VTO=-1)
.ENDS
```

Schritt 2:
Qucsstudio wird gestartet und dann das OPV-Projekt aufgerufen. Über „Projekt“ in der Menüleiste kommen wir zu „Importiere Daten“ und dann muss man natürlich den Pfad zur eben abgelegten Datei „TL072.cir“ korrekt einstellen. Mit „Konvertieren“ ist man schon am Ziel, denn das Programm meldet die erfolgreiche Konversion und man kann diese Maske schließen.

Schritt 3:
Dafür wurde jetzt automatisch im Projekt ein neues File „**TL072.sch 5-Port**“ erstellt und das öffnen wir. Darin wartet „Subcircuit“ mit fünf Beinen auf uns und wir können die gewohnte Schaltung eines umkehrenden OPVs mit zehnfacher Verstärkung bei einem Eingangswiderstand von 1kΩ zeichnen.



Wir steuern mit einer Sinusspannung (Spitzenwert = 1V / Frequenz = 1 kHz) an und simulieren von 0….10ms mit 16384 Punkten. Die Schaltung arbeitet mit +12V und -12V als Versorgung.



Na also…

7.5. Aktive Filterschaltung mit OPV

Für Filterzwecke (nicht nur im Audiobereich!) werden immer häufiger **die aktiven Filter** den passiven Filtern vorgezogen, da sie folgende Vorteile aufweisen:

-- Verwendung von Operationsverstärkern, die nur mit R und C gegengekoppelt sind
-- viel billiger, da keine (teuren) Spulen nötig sind und alle Abgleicharbeiten wegfallen
-- als SMD - Version oder IC sehr kleine Abmessungen
-- Verstärkung in weiten Grenzen einstellbar
-- bei starker Gegenkopplung und kleiner Verstärkung fast keine nichtlinearen Verzerrungen
-- wegen der Fortschritte in der OPV - Technik nun bis ca. 100MHz realisierbar

Die erforderlichen theoretischen Grundlagen und alle Berechnungshinweise finden sich z. B. im Standardwerk „Halbleiter-Schaltungstechnik“ von Tietze-Schenk (= Springer-Verlag). In der Zwischenzeit gibt es auch viele Tutorials und Shareware- oder Freeware-Entwurfsprogramme im INTERNET (z. B. in der Homepage von Burr - Brown oder von Microchip oder von Texas Instruments).

Anwendungsbeispiel: Sallen – Key - Tiefpass vierten Grades

Die erforderliche Schaltung und die nötigen Bauteilewerte liefern die erwähnten Filterprogramme nach Eingabe folgender Parameter:

Filterart: **Lowpass**

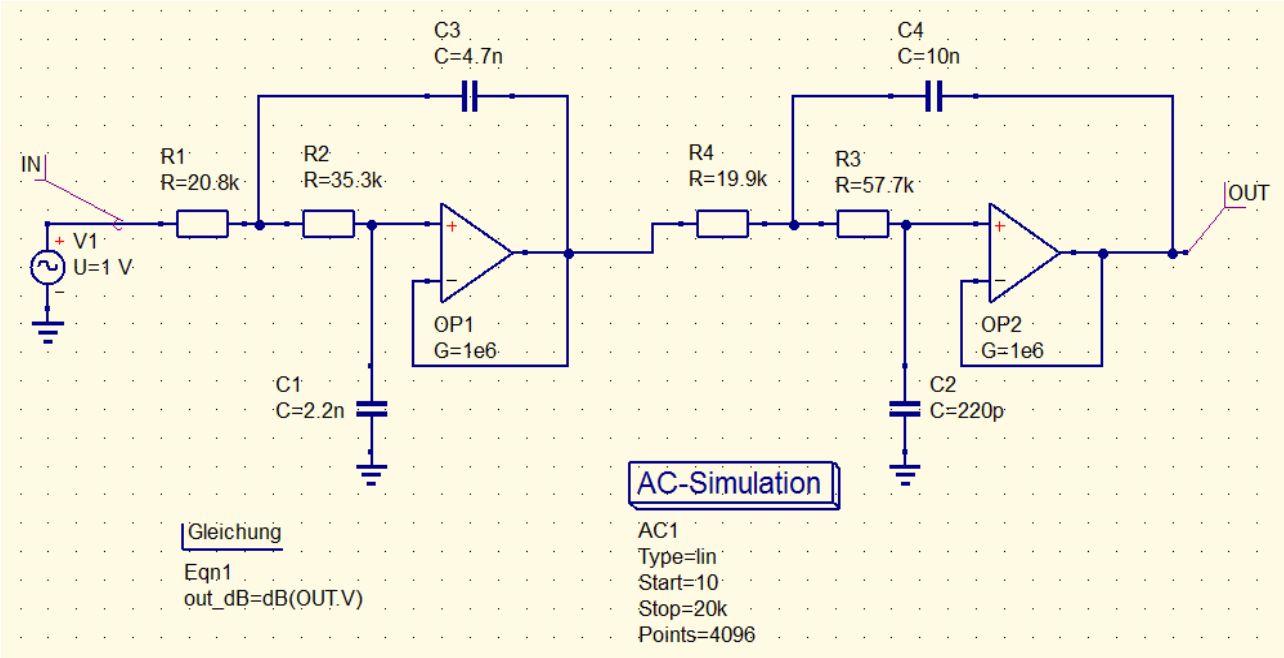
Filtertyp: **Tschebyschef**

Grenzfrequenz: **3400 Hz**

passband ripple“: **0,5 dB**
(= Maximale Welligkeit der Dämpfung im Durchlassbereich)

Filtergrad: **4**
(Er bestimmt die Anzahl der erforderlichen Bauteile und legt fest, wie steil der Übergang vom Durchlass- in den Sperrbereich erfolgt).

Und so sieht die Schaltung aus, wenn wir zur Simulation den idealen OPV verwenden:

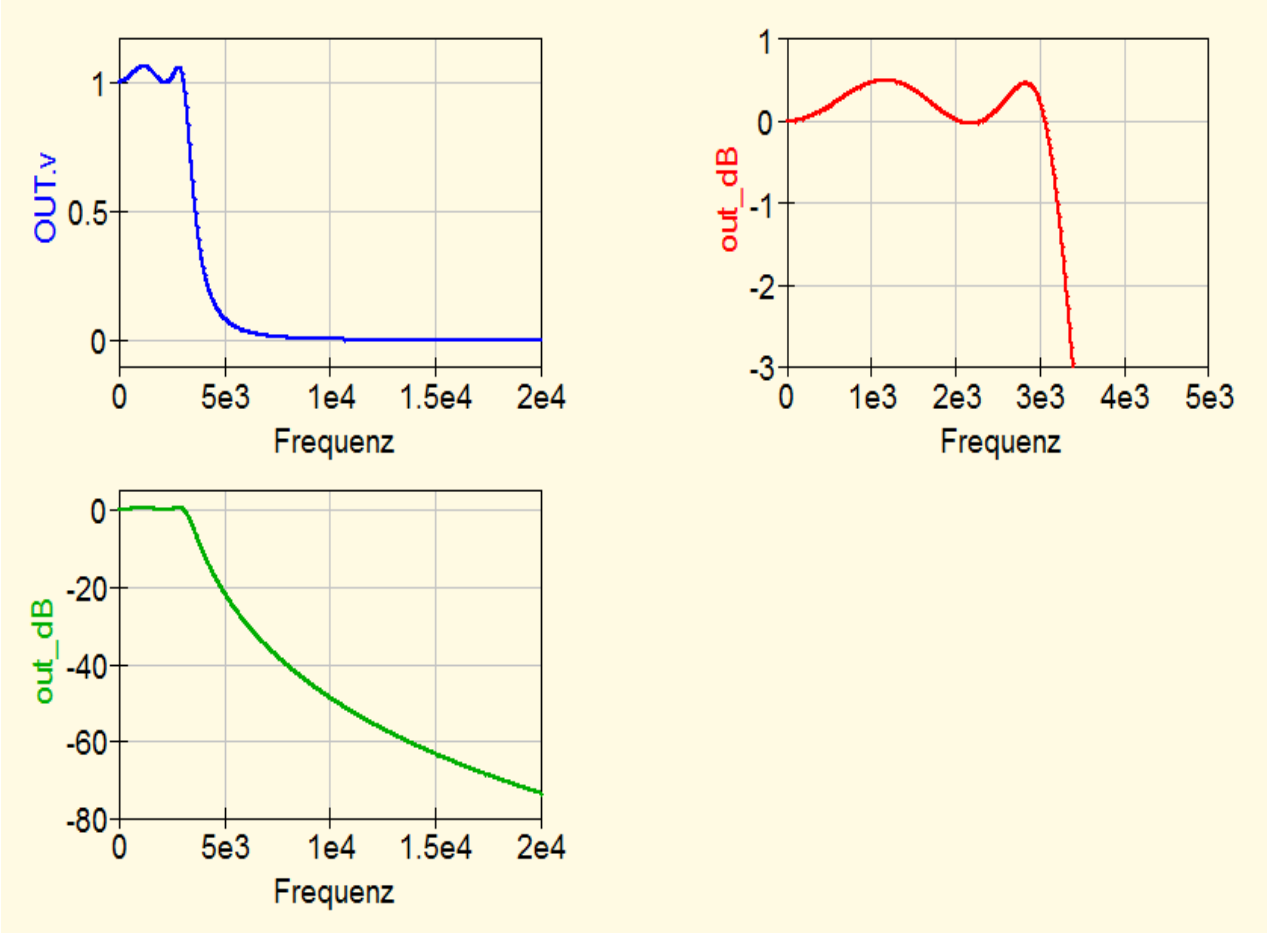


Das Ergebnis bei „OUT“ stellen wir in drei unterschiedlichen Diagrammen dar:

a) Linear bis 20 kHz

b) Logarithmisch bis 20 kHz und dB-Skala. Da sieht man die „Weitab-Selektion“ besser.

c) Ebenfalls logarithmisch, aber nur bis 5 kHz. Dafür dehnen wir die vertikale Achse, um den Bereich von -3 dB bis +1 dB. Dann erkennen wir nämlich die „Tschebyschef-Wellen“ im Durchlass - Bereich.

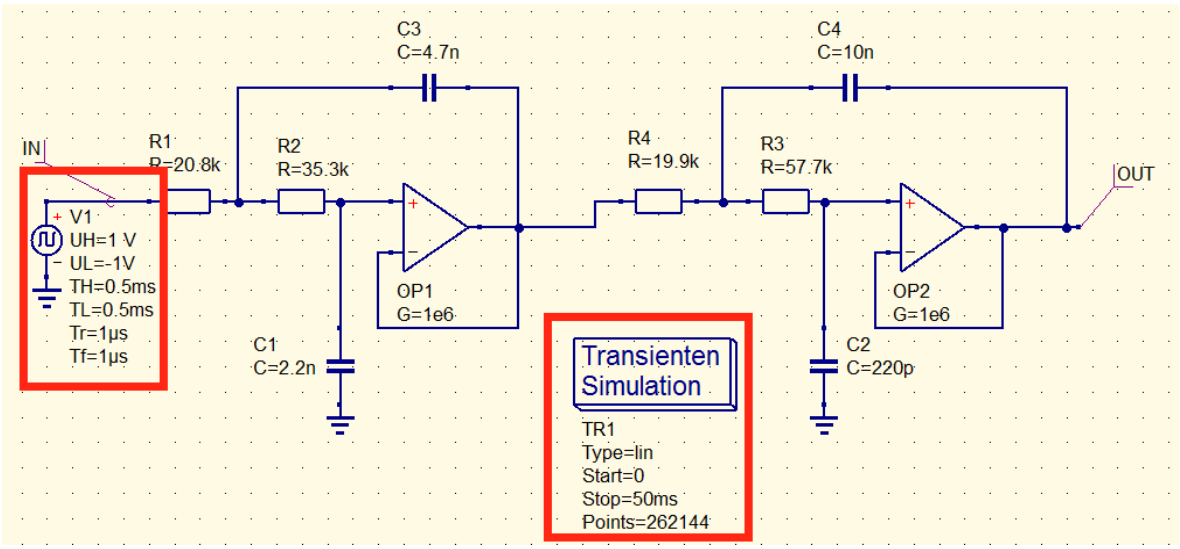


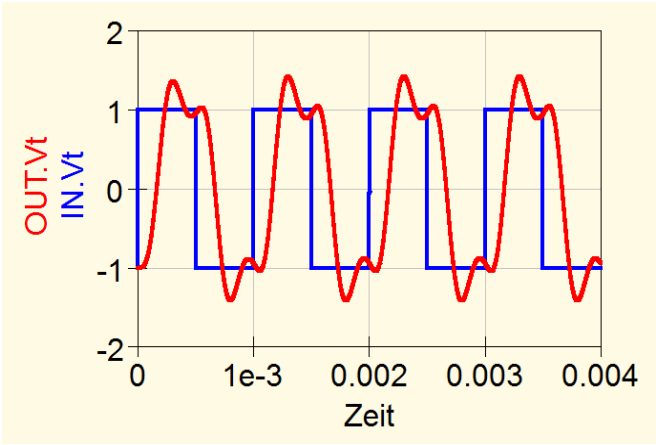
(Eine kleine Testfrage:
Wissen Sie noch, wie man die **Kurvenfarbe** ändert? Hier ist die Kurzanleitung:
Erst rechts auf das Diagramm klicken und „Eigenschaften editieren“ wählen. Dann das Häkchen bei „Farbe“ entfernen und anschließend gleich wieder anbringen. Dadurch wird das Farbfenster schwarz und wenn man nun auf dieses Fenster klickt, erscheint die Farbpalette. Da sucht man den gewünschten Farbton aus, klickt ihn an und bestätigt mit OK).
==========

Nun wollen wir uns an einem praktischen Fall die Filterwirkung der Schaltung ansehen und speisen den Eingang mit einem **Puls-Signal:**

f =	**1kHz**
Negativer Spitzenwert =	**-1V**
Positiver Spitzenwert =	**+1V**
Pulsbreite =	**500µs**
Periodendauer =	**1ms**
Anstiegs- und Abfallzeit je	**1µs**

Man muss natürlich nun eine Rechteck-Quelle verwenden und sie richtig programmieren. Bei der erforderlichen Transienten-Simulation wählt man eine Simulation von **0….50 ms** und (für eine eventuelle spätere erfolgreiche FFT) **262 144 Punkte**. Das ergibt eine Frequenzauflösung im Spektrum von **b = 1/50 ms = 20 Hz.**





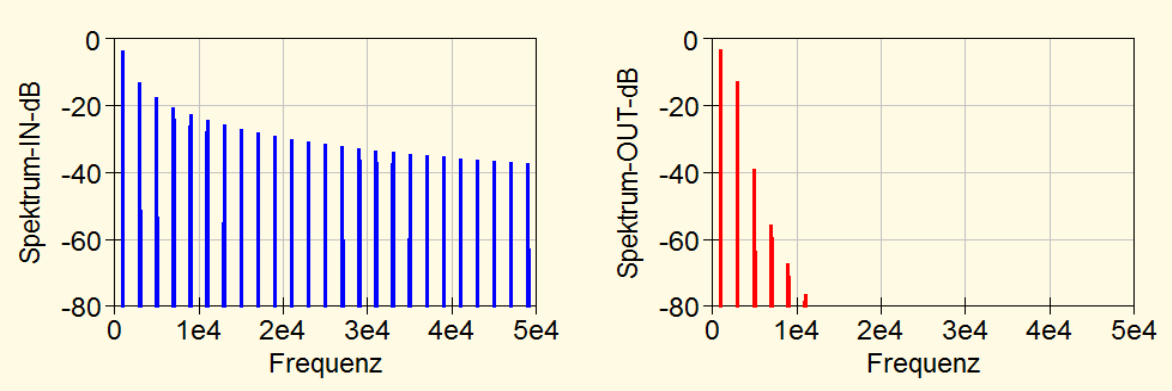
So sehen Eingangs- und Ausgangssignal aus. Man erkennt deutlich, wie am Ausgang der Schaltung

a) die **Oberwellen des Pulses wegen der Filterwirkung gedämpft** sind oder sogar fehlen, aber auch

b) wegen der Phasen- und Gruppenlaufzeitverzerrungen eines Tschebyschef-Filters **Signalverzerrungen und Überschwinger** auftreten.

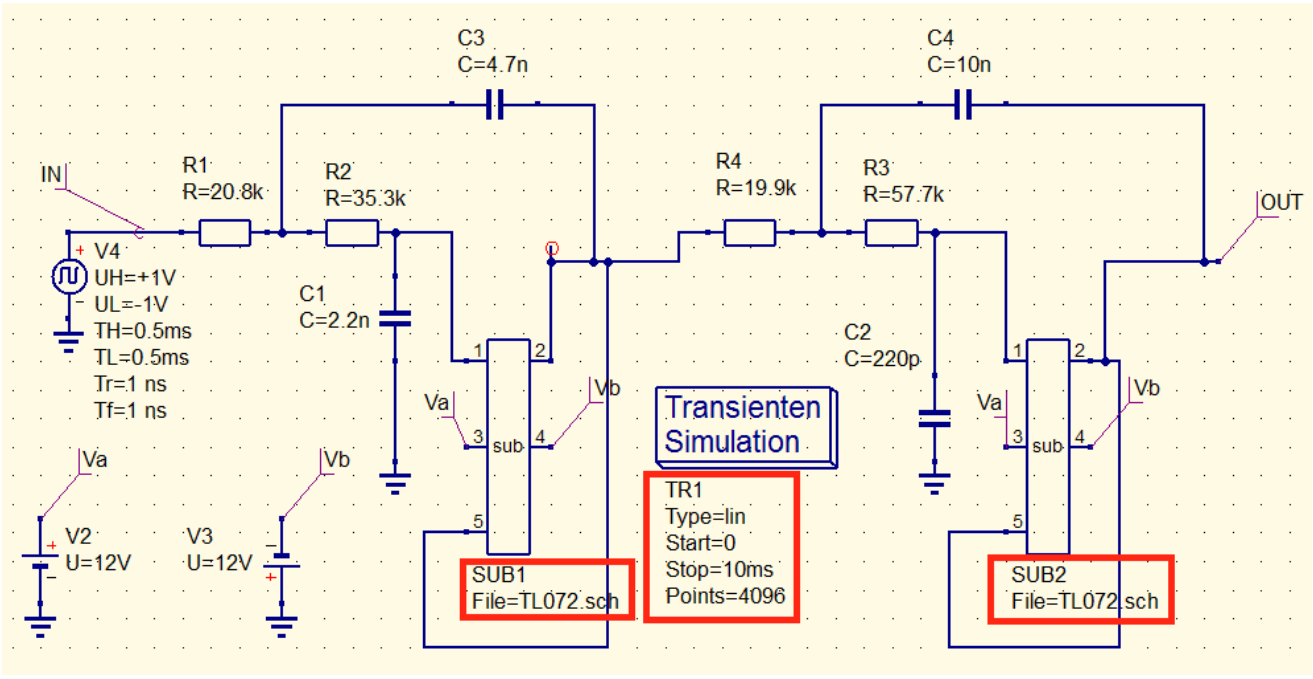
Die Filterwirkung lässt sich auch sehr schön durch einen **Vergleich der Spektren beider Signale über die FFT zeigen** (maximal dargestellte Frequenz: f = 5 kHz).

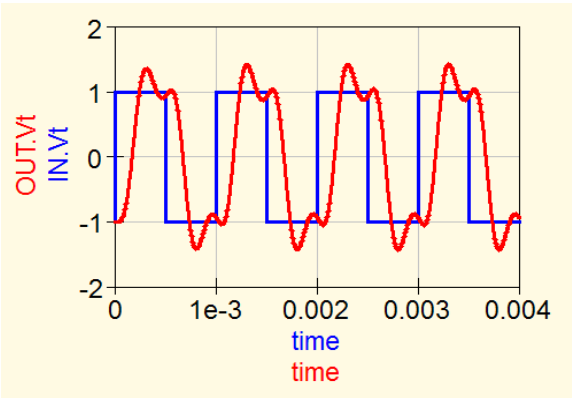
Erinnerung: das Spektrum lässt sich(z. B. für das Eingangssignal) durch folgenden Graphen erzeugen: dB(time2freq(IN.Vt,3)). Dabei legt die Ziffer „3“ zuerst ein Hanning-Window über den zeitlichen Verlauf von „IN.Vt“.



Testen wir doch mal eine Version der Schaltung mit dem Spice Modell des TL072 (Siehe Kapitel 6.4.)

Da sollte man aber gleich die **Zahl der Simulationspunkte auf den Minimalwert von 4096 zurücknehmen,** denn diese konvertierten Modelle sind sehr aufwendig und entsprechend lang dauert dann die Berechnung!
Gespeist wird mit +12V und -12V -- aber da sollte man wieder mit Labels arbeiten.





Die Simulation bringt keine Überraschungen, denn der TL072 tut seinen Dienst wie verlangt.

Noch einige Tipps für die Umsetzung solcher aktiver Filter in eine praktische Schaltung (= PCB = Printed Circuit Board):

a) Bei den **Widerstandswerten sollte man 1 MegaOhm nicht überschreiten**. Darüber gibt es Stabilitätsprobleme und das Rauschen dieser Widerstände steigt mit ihrem Wert an.

b) Kondensatoren mit **Werten unter 10 pF haben ihren Platz nur in der HF – und Mikrowellentechnik**. Grundsätzlich braucht man kleine Toleranz und hohe Güte. Geeignet sind Glimmerkondensatoren, Keramikkondensatoren aus dem Werkstoff NP0 sowie Polystyrol- und Polypropylen-Kondensatoren. **Aber Werte über 1µF sollte man vermeiden**.

c) **Elektrolytkondensatoren sind grundsätzlich tabu und unbrauchbar** (= schlechte Güte, große Streuung).

Auch ein weiteres Problem scheint wenig bekannt zu sein:
Vom Ausgang des OPVs geht ja ein Gegenkopplungskondensator zurück in Richtung Eingang. Leider sind beim Kondensator **Strom und Spannung um 90 Grad phasenverschoben** und das hat bei manchen OPV-Typen üble Folgen:

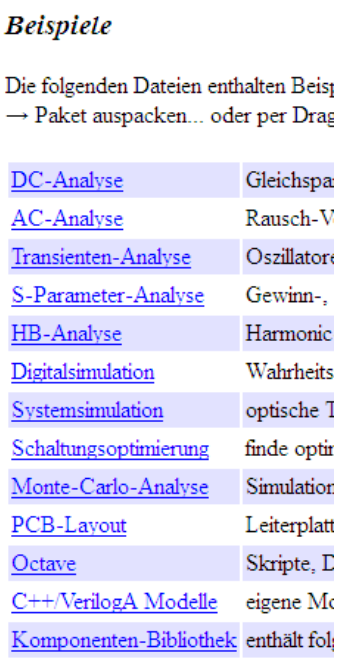
Zur Verminderung der Stromaufnahme weisen sie oft Gegentakt-Endstufen auf, denn die arbeiten mit sehr kleinem Ruhestrom. Und wenn nun beim Kondensator gerade das Strommaximum geliefert werden soll, ist die zugehörige Ausgangsspannung gerade Null. **Die Folge dieses Problems ist eine starke Verzerrung des Ausgangssignals im Bereich um den Nulldurchgang der Ausgangsspanung -- dieses Stück fehlt dann! Dieser Effekt tritt dann meist bei Kapazitätswerten über 50 nF ….100 nF auf.**

Die Abhilfe ist einfach: ein **zusätzlicher Ohmscher Widerstand mit 1k….3kΩ vom Ausgang des OPV gegen Masse,** der ja keine Phasenverschiebung zwischen Strom und Spannung aufweist.
So wird über seinen „In Phase Current“ der erwähnte Effekt kompensiert, denn die Gesamt-Phasenverschiebung in der Endstufe beträgt nun keine 90 Grad mehr, sondern geht in Richtung „In Phase“.

8. Simulation von Digitalschaltungen

8.1. Verwendung von Beispielen aus der Qucsstudio-Homepage

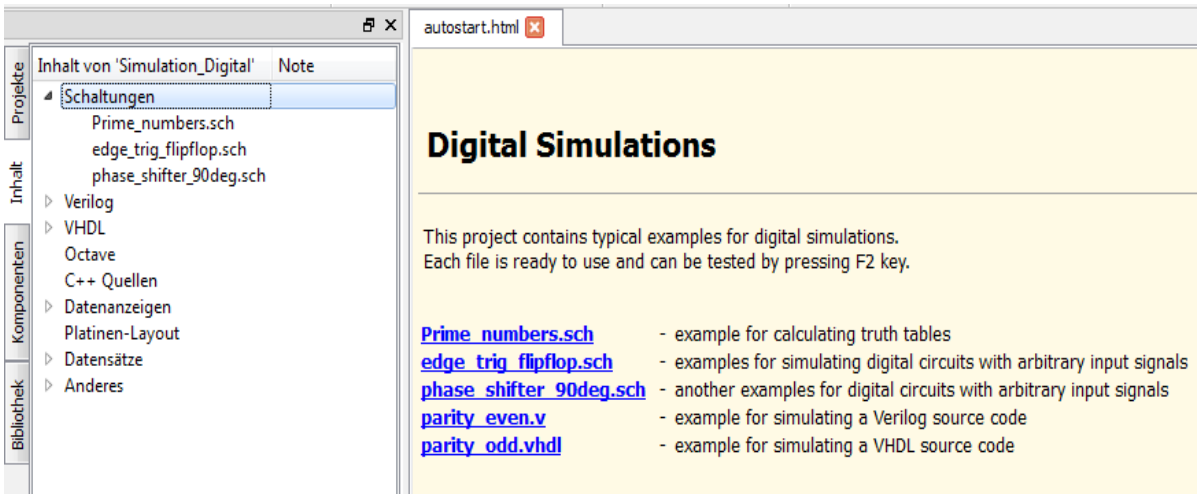
Schritt 1:



In der Homepage der Software findet sich das nebenstehende Menü „**Beispiele**“. Die aufgeführten Dateien lädt man sich einzeln auf den eigenen Rechner herunter und verschiebt anschließend diese komplette Sammlung in die Qucsstudio-Projektverwaltung -- bei mir ist das „c:\ Benutzer \ admin \ .qucs“.

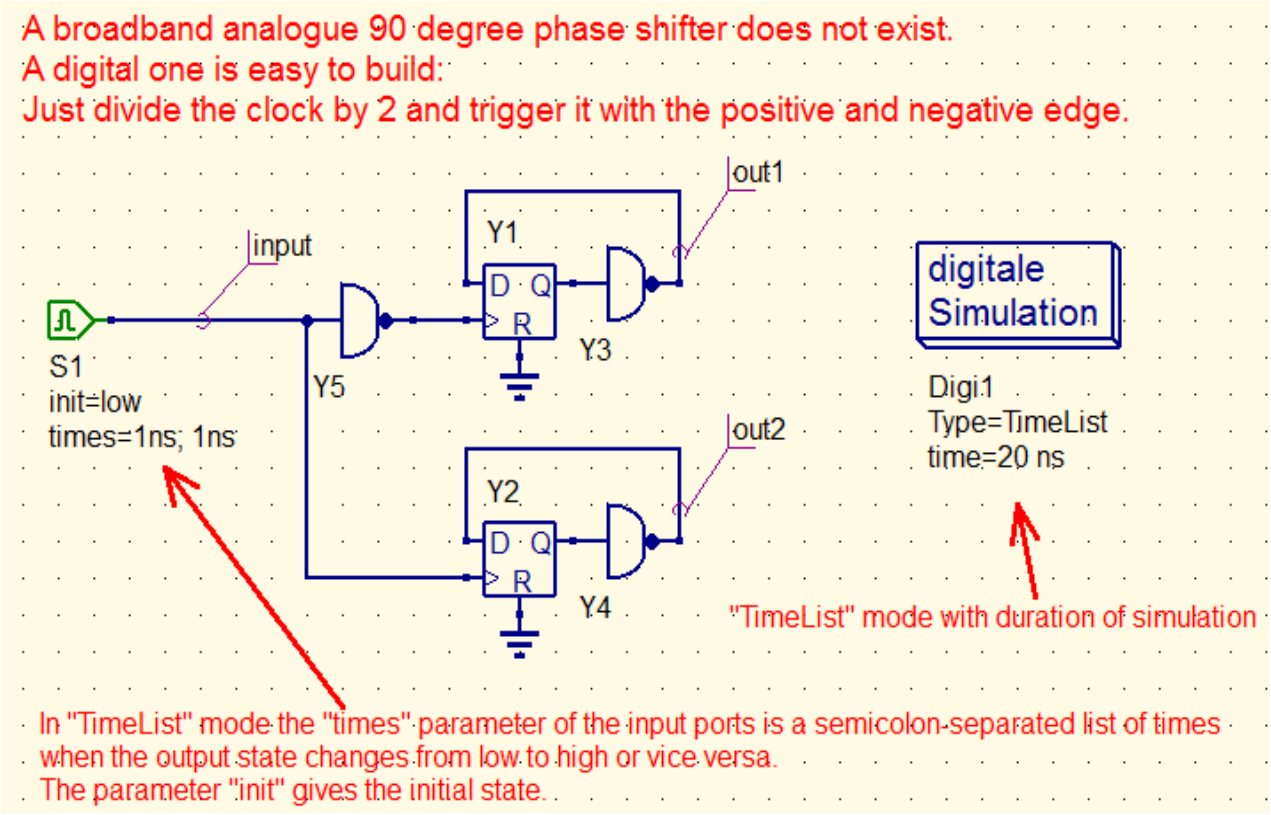
Schritt 2:
Dann wird qucsstudio gestartet und am linken Rand das oberste Menü „**Projekte**“ gewählt.
Da brauchen wir aus der Windows-Menüleiste die Karteikarte „**Projekt**“ und darin „**Projekt auspacken**“ zu wählen. Unser Rechner listet dann sofort die vorhin verschobene Beispielsammlung auf und darin können wir „**Simulation_Digital**“ auswählen. Das Kommando „**Öffnen**“ führt dann sofort zum Ziel und wir finden „**Simulationen_digital**“ neu in der Projektliste. Dieses neue Projekt öffnen wir.

Schritt 3:
Da haben wir schon, was wir möchten:



Nehmen wir uns das Beispiel „**phase_shifter_90deg.sch**“ vor und öffnen es.

8.2. Untersuchung des Beispiels „phase_shifter_90deg.sch“



Alle verwendeten Bauteile finden sich in „Komponenten / Digitalkomponenten“ und wir finden

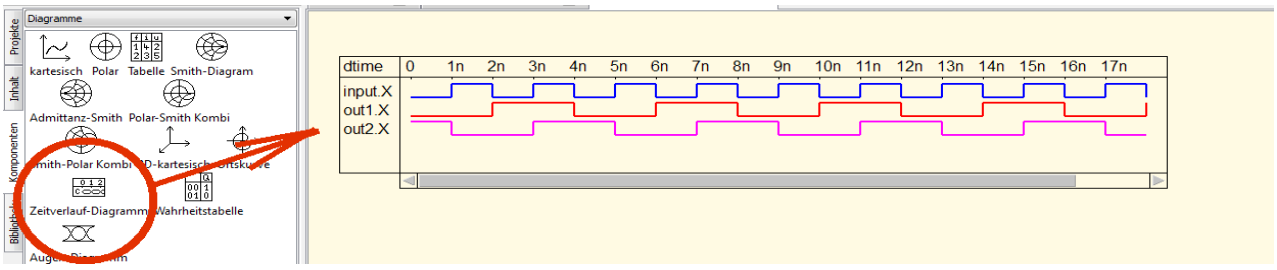
a) Als „S1“ eine Digitalsignal-Quelle

b) „Y3“, „Y4“ und „Y5“ stellen Umkehrstufen dar.

c) Und „Y1“ bzw. „Y2“ sind natürlich D-Flipflops

Die Schaltung enthält bereits alle erforderlichen Erläuterungen und Einstellungen.
Bitte die Hinweise bei der Signalquelle und bei der digitalen Simulation genau studieren.

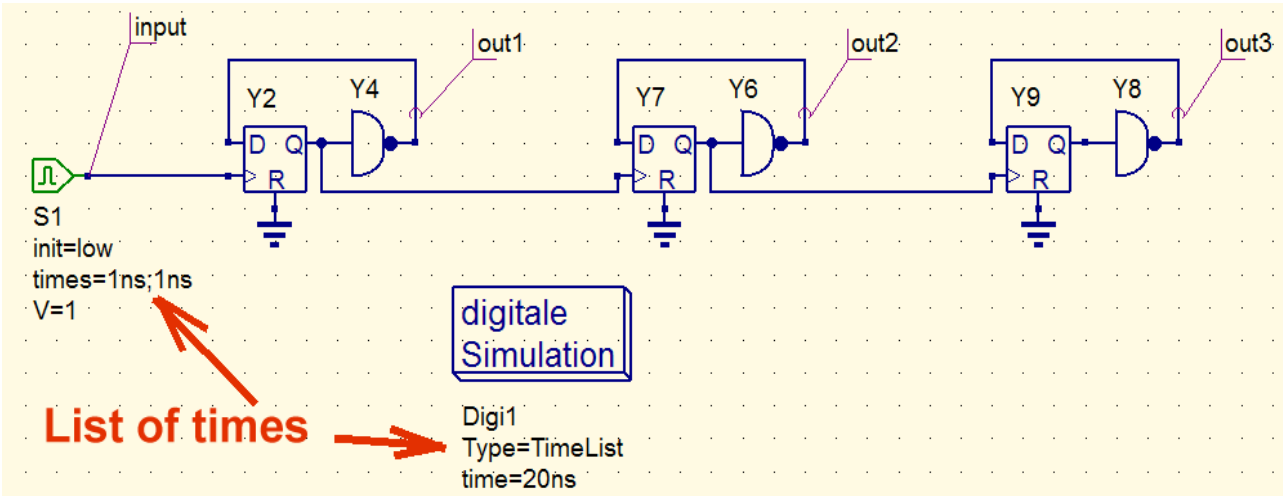
Dann wird simuliert und anschließend (nach einem Rechtsklick auf das Diagramm, gefolgt von „Eigenschaften editieren“) auf Liniendicke 2 umgestellt:



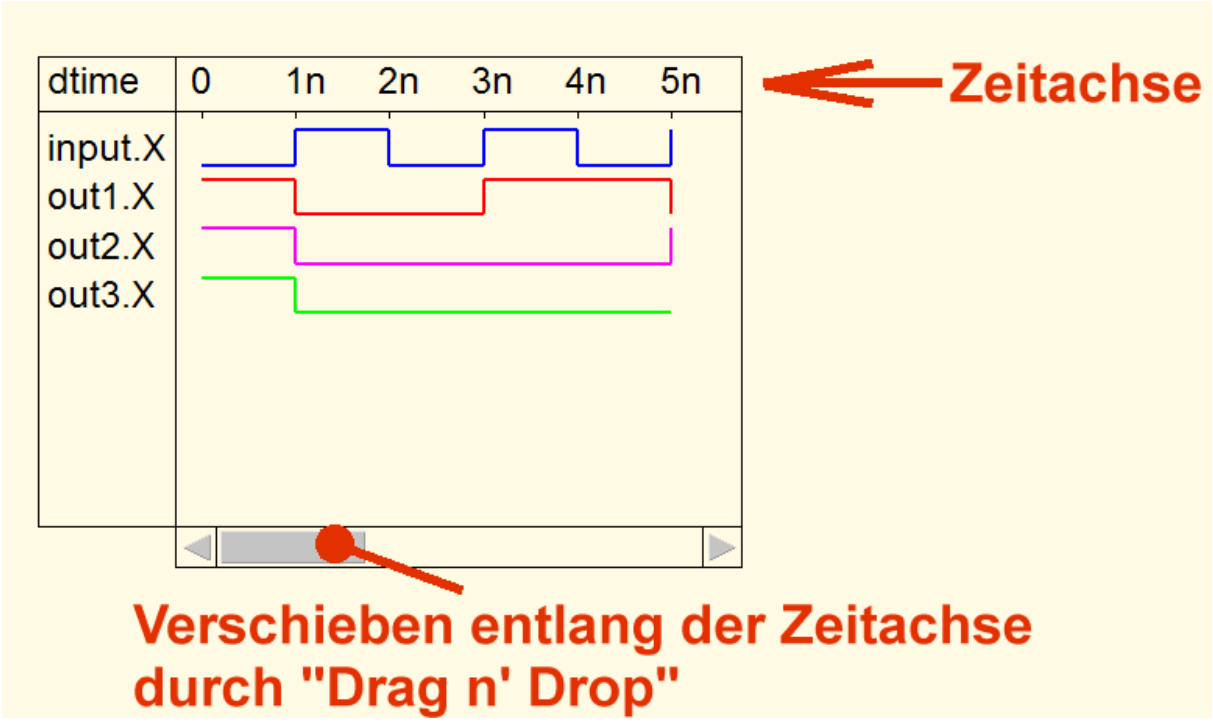
Im linken Kreis findet sich die Information, welches Diagramm für die Ausgabe der Digitalsignale gestartet werden muss. Es ist „**Zeitablauf-Diagramm“.**

8.3. Ein dreistufiger Frequenzteiler mit D-Flipflops

Da wir beim vorigen Beispiel gerade zwei D-Flipflops auf dem Bildschirm haben, speichern wir einfach die Schaltung unter einem neuen Namen (hier: „frequency_divider.sch“). Darin wird eines der beiden Flipflops herausgelöscht, das zweite dagegen gemäß dem folgenden Schaltbild zweimal kopiert. Auch den Erläuterungstext können wir entfernen. Die drei Flipflops verdrahten wir nun und benennen die Labels der Ausgangssignale um:



Zur „List of times“ ist folgende Erklärung nötig;
Der erste Wert gibt die „Startverzögerung“ an. Getrennt durch einen Strichpunkt folgt die Angabe des „Ticks“, also des zeitlichen Rhythmus für den Pegelwechsel beim Eingangssignal. Simuliert wird für eine Zeit „time = 20ns“.
Gestartet wird mit Low-Pegel beim Eingangssignal -- das erzwingt das Kommando „init=low“ bei der digitalen Signalquelle. Und so sieht das Ergebnis aus, wenn gleich (nach einem rechten Mausklick und „Eigenschaften editieren“) auf eine Liniendicke von 2 umgestellt wird:



Wichtig:
Es wird nicht die komplette Zeit von 0….20ns dargestellt, sondern mit dem „Reiterlein“ unterhalb des Diagramms kann man an der Zeitachse entlang fahren. Bitte ausprobieren!

9. Arbeiten mit S-Parametern

9.1. Grundlagen der S-Parameter

In der Nachrichtentechnik werden bei der Planung größerer Systeme (Eingabe der Information / Sender / Übertragungsmedium / Empfänger / Ausgabe der Information) viele Baugruppen benötigt. Es ist -- speziell bei hohen Frequenzen! -- deshalb üblich, bei diesen Baugruppen das **Prinzip der Leistungsanpassung** anzuwenden, um

a) die größte mögliche Leistung von einem Baustein an den nächsten weiterzugeben und

b) dadurch auch den größten Abstand des Nutzsignals zu den unvermeidlichen und unerwünschten Störungen (Rauschen, Verzerrungen, Intermodulationsprodukte, Übersprechen etc.) zu erzielen und so die beste, in diesem Fall mögliche Übertragungsqualität zu sichern.

Deshalb gilt im kompletten System derselbe „Systemwiderstand“ -- für die Verbindungskabel ebenso wie für die Eingangs- und Innenwiderstände der Baugruppen. Üblich ist hierbei der Wert von

Z = 50Ω

….wobei aber die Radio-, Fernseh- und Videotechnik leider stur bei ihren mal eingeführten 75Ω bleibt…
Jeder einzelne Baustein wird dabei durch seine **S-Parameter** beschrieben. Dazu wird zuerst der Eingang (= Port 1) des Bausteins an einen Signalgenerator mit dem Innenwiderstand Z = 50Ω über ein Kabel mit Z = 50Ω angeschlossen, wobei der Baustein-Ausgang einfach mit Z = 50Ω abgeschlossen wird. Somit speist die „Hinlaufende Welle“= „**Incident Wave**“ (mit der Amplitude $U_{incident}$ = Uo / 2) den Baustein-Eingang und verursacht bei fehlerhafter Anpassung eine „**Reflektierte Welle = Reflected Wave**“ -- also ein Echo. Die Leistungen beider Signale werden ins Verhältnis gesetzt, das Ergebnis als „**Eingangs-Reflektion S11**“ bezeichnet und das Ganze in dB ausgedrückt.

(Diese Bezeichnung **S11** ist eigentlich eine Abkürzung und bedeutet
S = „Scattering Parameter = Streu-Parameter
1 = am Eingang 1, also an Port 1 wird das reflektierte Signal gemessen
1 = zum Eingang 1, also zu Port 1 wird das Generatorsignal (= incident wave) mit der Amplitude $U_{incident}$ = Uo/2 geschickt.)

$$S11 = 10 \bullet \log\left(\frac{P_{reflected}}{P_{incident}}\right) = 10 \bullet \log\left(\frac{\left(\frac{U_{reflected}^{\ 2}}{Z}\right)}{\left(\frac{U_{incident}^{\ 2}}{Z}\right)}\right) = 10 \bullet \log\left(\frac{U_{reflected}^{\ 2}}{U_{incident}^{\ 2}}\right) = 20 \bullet \log\left(\frac{U_{reflected}}{U_{incident}}\right)$$

in dB.
Kurz und prägnant:

S11 = Eingangs-Reflektion = input reflection = informiert über die Abweichung des Eingangswiderstandes von den geforderten 50Ω (…im Idealfall möglichst keine Abweichungen von 50Ω und das ergibt einen hohen negativen dB-Wert für S11)

Bitte merken:

a) **Vollständige Leistungsanpassung** haben wir bei R = Z (z. B. 50Ω) -- da gibt es **kein Echo**. Dazu gehört ein extrem niedriger negativer Wert bei S11 in dB, theoretisch wären das „Minus Unendlich dB“ bei absolut perfekter Anpassung.

b) **Komplette Totalreflektion** und damit **keinerlei Leistungsabgabe haben wir bei einem Leerlauf oder Kurzschluss**. Dazu gehört S11 = +1 (beim Leerlauf) und S11 = -1 (beim Kurzschluss). Der Betrag von S11 in dB geht dann gegen Null.

Bei „**Oneport“ - Bausteinen** (wie z. B. Antennen oder Widerständen oder Dioden oder Kondensatoren etc.) ist S11 der **einzige messbare S-Parameter**. Bei einem „**Twoport**“ (= Verstärker, Filter, Attenuator etc.) gibt es aber noch einen Ausgang (= Port 2) und die Hinlaufende Welle erzeugt auch dort ein Signal. Also definiert man einen weiteren S-Parameter **S21** und tauft ihn „**Forward Transmission**“.

S21 bedeutet: „S-Parameter mit der Bezeichnung Forward Transmission“.
Er gibt die Leistungsverstärkung des Bausteins in dB an, wobei die Ausgangsleistung am Abschlusswiderstand von Port 2 gemessen und ins Verhältnis zur Hinlaufenden Welle an Port 1 gesetzt wird. Er soll deshalb z. B. bei Verstärkern nicht unter 1 = Null dB sinken.

Dann wird die ganze Sache umgedreht und der Generator samt Kabel an den Ausgang (Port 2) angeschlossen. Der Eingang (Port 1) wird nun mit den Systemwiderstand Z= 50Ω abgeschlossen, dann wird gemessen. Damit erhält man zwei weitere Parameter:

S22 = Ausgangs-Reflektion = output reflection = informiert über die Abweichung des Innenwiderstandes von den geforderten 50Ω. (…im Idealfall möglichst keine Abweichungen von 50Ω und das ergibt einen hohen negativen dB-Wert für S22)

S12 = Rückwärts-Transmission = reverse transmission = informiert über die Rückwirkungen vom Ausgang zurück zum Eingang (…sollte z. B. bei Verstärkern ebenfalls möglichst Null und damit ein hoher negativer dB-Wert sein…sonst schwingt die Kiste…)

==========

9.2. Praxis-Beispiel: Tschebyschef-Tiefpass mit einer Grenzfrequenz von 11 MHz

9.2.1. Zuerst etwas über Filterschaltungen

Filter dienen dazu, bestimme Frequenzbereiche in einem Signal zu unterdrücken und andere möglichst ungeschwächt durchzulassen. Deshalb unterscheidet man

Tiefpässe (= nur tiefe Frequenzen dürfen bis zu einer bestimmten Grenzfrequenz passieren),

Hochpässe (= nur hohe Frequenzen dürfen erst oberhalb einer bestimmten Grenzfrequenz passieren),

Bandpässe (= nur ein schmales Frequenzband wird durchgelassen) und

Bandsperren (= alles wird durchgelassen -- bis auf eine schmale Lücke).

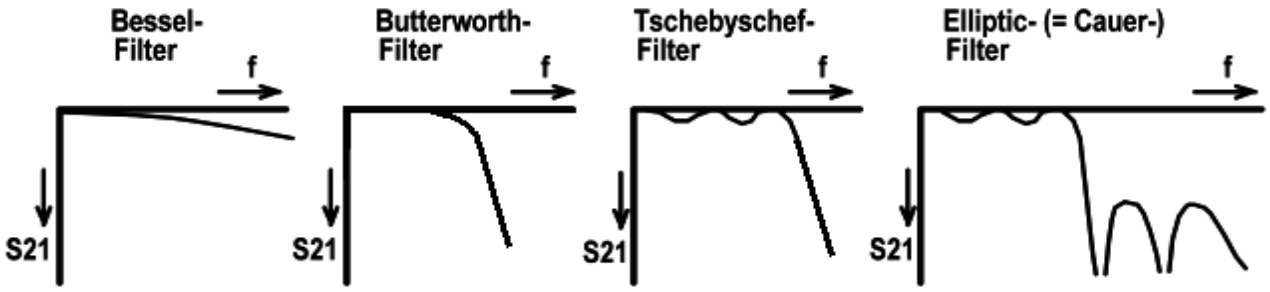
Sie werden meist als Twoports mit dem Systemwiderstand Z = 50 Ω mit passender Software entworfen, die in der Zwischenzeit längst kostenlos im Internet zu finden ist (= „Filter Calculator“). Jedoch muss man etwas über die 4 wichtigsten Filtertypen und ihre Eigenheiten wissen.

a) **Bessel-Filter**
Bei ihnen wird besonders auf geringste Phasen- und Gruppenlaufzeitverzerrungen beim Übergang vom Durchlassbereich (= pass band) in den Sperrbereich (= stop band) geachtet. Nur so ist eine korrekte Übertragung von Rechtecksignalen (und anderen, nicht sinusförmigen Signalformen) mit ihrem hohen Anteil an Harmonischen sichergestellt. Die Kehrseite ist ein äußerst müder Übergang vom Durchlass- in den Sperrbereich und eine nur langsam und monoton ansteigende Sperrdämpfung.

b) **Butterworth-Filter**
Der Übergang vom Durchlass- zum Sperrbereich ist viel steiler als beim Bessel-Filter mit gleichem Filtergrad, um möglichst schnell die unerwünschten Signalanteile zu sperren. Das führt zu stärkeren Gruppenlaufzeit-Verzerrungen, wenn es auf den Sperrbereich zugeht. Mit steigender Frequenz im Sperrbereich steigt die Dämpfung monoton bis Unendlich an (= S21 nimmt monoton ab).

c) **Tschebyschef-Filter**
Der „Knick“ beim Übergang vom Durchlass- in den Sperrbereich wird wesentlich schärfer als beim Butterworth-Filter ausgeführt und das führt zu **S21-Wellen (= „ripple“) im Durchlassbereich**. Das ergibt für nichtsinusförmige Signale starke Verzerrungen der Kurvenform, sobald man sich der Grenzfrequenz nähert, da hier die Gruppenlaufzeit-Verzerrungen nochmals deutlich schlimmer sind.
Wird vor allem bei sinusförmigen Signalen angewandt, um die Oberwellen möglichst stark zu bedämpfen.
Die Dämpfung steigt im Sperrbereich monoton bis Unendlich an (= S21 nimmt monoton ab) und **dieses Sperrverhalten entspricht exakt dem Butterworth-Filter bei gleichem Filtergrad.**

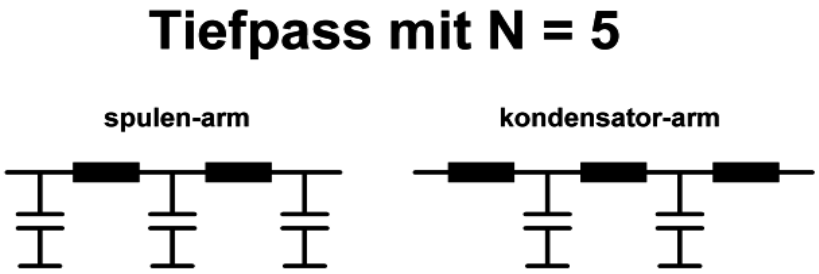
d) **Elliptische Filter (= Cauer Filter)**
Gleiches Verhalten mit Wellen im Durchlassbereich wie bei Tschebyschef-Filtern, aber **die Steigung im Sperrbereich wurde (bei gleichem Filtergrad) nochmals versteilert**. Das führt im Sperrbereich zu „Einbrüchen“ bei der Sperrdämpfung. Dort wird also nur ein bestimmter Mindestwert der Dämpfung garantiert.
Hier sind die prinzipiellen S21-Verläufe für die obigen Filtertypen dargestellt. Bitte dabei beachten:
a) Alle „S21-Hügel“ im Durchlassbereich von Tschebyschef- und Elliptic-Filtern haben dieselbe Amplitude.
b) Die Maxima der „Sperrbereichs-Dämpfungseinbrüche“ beim Elliptic- Filter sind ebenfall gleich hoch.



Für die **Steilheit des Übergangs** vom Durchlassbereich (Pass Band) in den Sperrbereich (stop band) ist neben dem eben besprochenen Filtertyp auch der **Filtergrad („Order“)** verantwortlich. **Je höher der Filtergrad, desto steiler wird der Übergang.** Er gibt den Grad der Polynome in der Übertragungsfunktion des Filters an, aber schon beim Tiefpass hat er auch eine ganz praktische Bedeutung:

Die Anzahl der nötigen Bauteile (die Energie speichern können) entspricht exakt dem Filtergrad!

Sehen wir uns das bei einem Tiefpass (= LPF = Low Pass Filter) für N = 5 an. Der muss natürlich 5 Spulen oder Kondensatoren aufweisen, aber da kommen wir nun zum letzten Unterscheidungspunkt: man kann ihn entweder „**spulenarm**“ oder „**kondensatorarm**“ ausführen (…Hinweis: Spulen sind IMMER die wesentlich teureren Bauteile….):



….und nie vergessen:
Jeder Filter wird von links mit einer Quelle (Innenwiderstand Z = 50Ω) gespeist, rechts dagegen wird er mit einem Widerstand von 50Ω abgeschlossen (= z. B. durch den Eingang der nächsten Stufe)

Übrigens:
Die Angabe der „**Grenzfrequenz**“ bedeutet immer, dass dort die **Signalamplitude gegenüber dem Passband um 3dB abgenommen hat (= S21 ist um 3dB kleiner geworden).**
Aber bei Tschebyschef- und Elliptic-Filtern gibt man bisweilen auch gern die „Ripple-Grenzfrequenz“ an, bei der die Dämpfung zum ersten Mal den maximalen S21-Ripple-Wert überschreitet.
Bei **Bandpässen und Bandsperren** gibt es dann logischerweise eine **untere und eine obere Grenzfrequenz**, da ja dort ein komplettes Frequenzband durchgelassen oder gesperrt wird.

9.2.2. Pflichtenheft für das Projekt

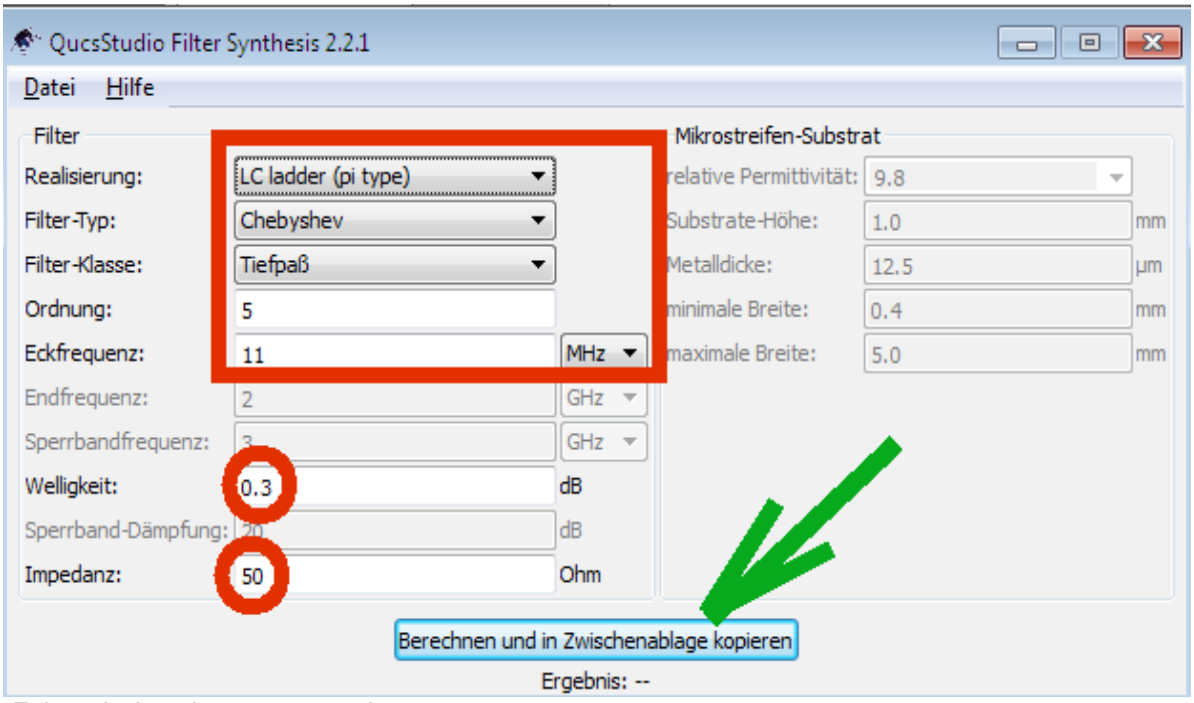
Wir wollen für ein Kommunikationssystem mit Z = 50 Ω einen Tiefpass mit N = 5 entwerfen, der alle Signale oberhalb der in UKW-Empfängern üblichen Zwischenfrequenz von f = 10,7 MHz unterdrückt. So etwas geht grundsätzlich immer mit dem **Pflichtenheft** los:

Systemwiderstand:	**Z = 50 Ω**
Filtertyp:	**Tschebyschef**
Filtergrad (Order):	**N = 5**
Welligkeit (Ripple) im Durchlassbereich:	**0,3 dB**
3dB-Grenzfrequenz:	**11 MHz**
Gemessene Spulengüte:	**Q = 70 bei 11 MHz**

In qucsstudio ist ein passender **Filter-Calculator** eingebaut und den nehmen wir uns gleich vor.

9.2.3. Filterentwurf mit qucsstudio: Einsatz des Filtercalculators und der S-Parameter-Simulation

Dazu legen wir ein neues Projekt an und öffnen eine leere Schaltplanseite. Anschließend startet man in der Karteikarte „Werkzeuge“ die Option „Filter-Synthese“ und füllt die folgende Eingabemaske aus:

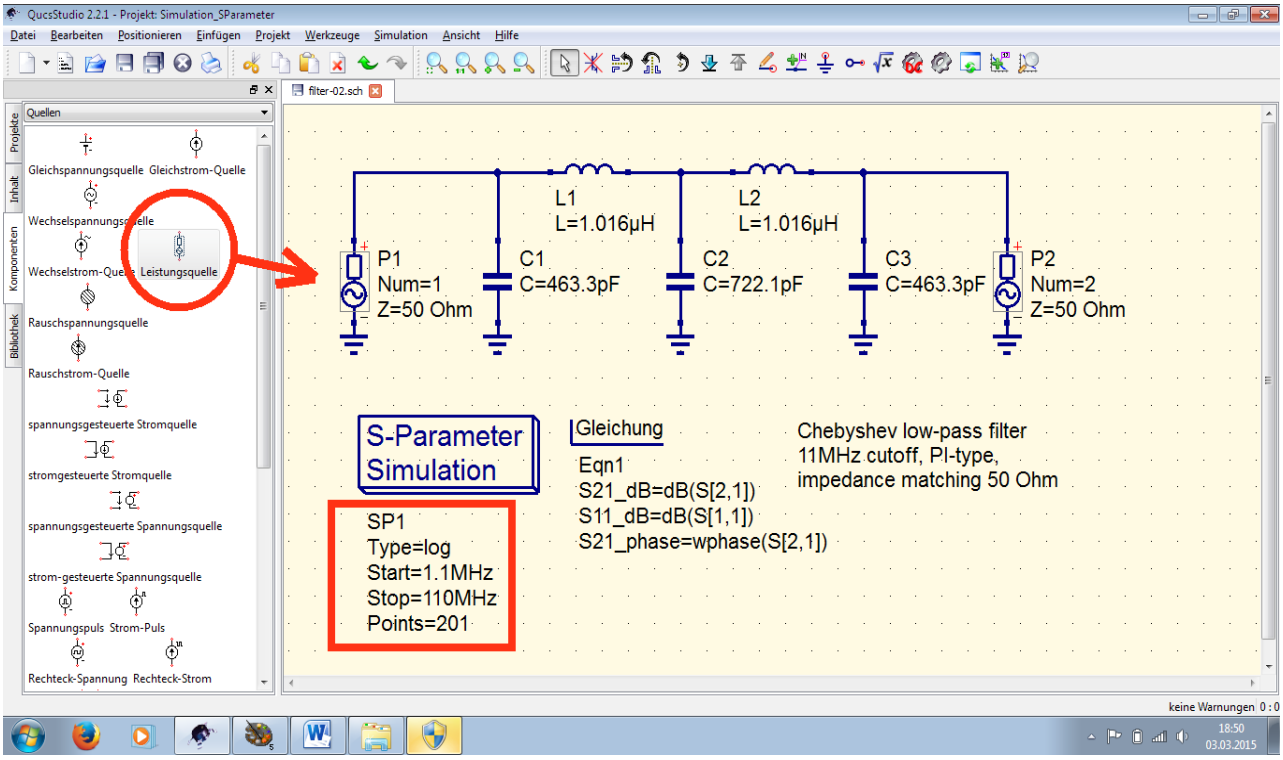


Folgende Angaben müssen rein:

Realisierung	**= LC Ladder (pi type)**
Filter-Typ	**= Chebyshev**
Filter-Klasse	**= Tiefpass**
Ordnung (= Order)	**= 5**
Eckfrequenz (corner frequency)	**= 11 MHz**
Welligkeit	**= 0.3 dB**
Impedanz	**= 50 Ohm**

Mit der “grünen Aktion” wird das Ergebnis in die Zwischenablage kopiert und ist zunächst unsichtbar…

Aus der Zwischenablage (Clipboard) heraus kann das Ergebnis im Schaltplan abgesetzt werden. Es handelt sich um die bereits komplett für die Simulation vorbereitete Schaltung.



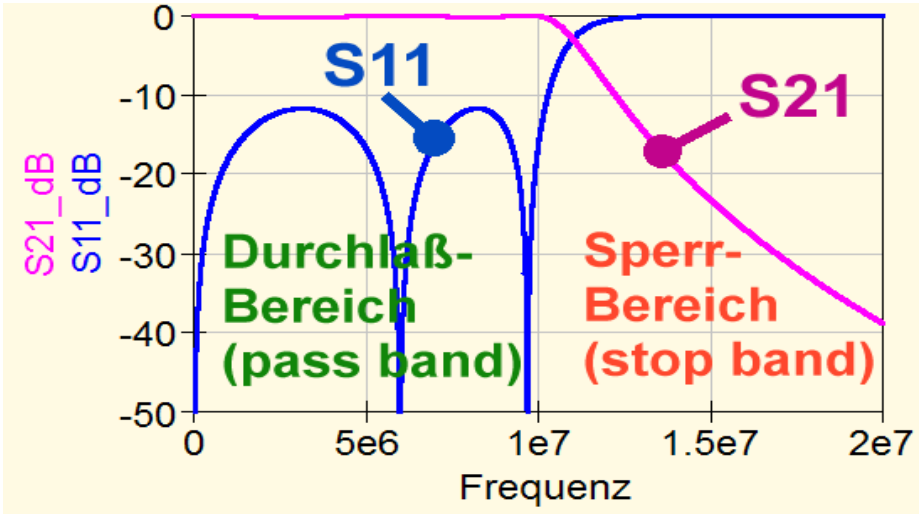
S-Parameter
Simulation

SP1
Type=lin
Start=1k
Stop=20MHz
Points=262144

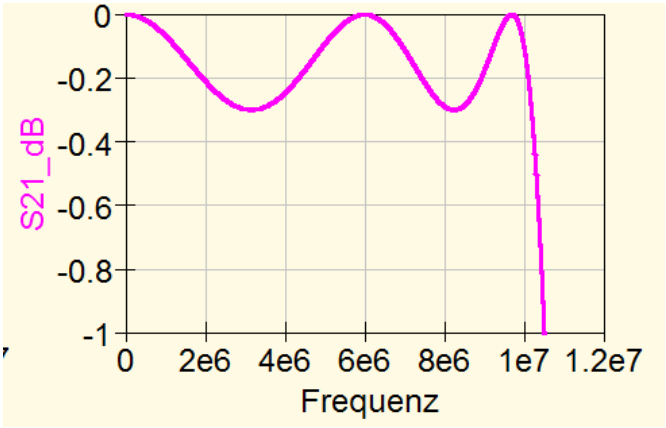
Zur Information:
Die beiden „Ports“ zur Speisung der Schaltung stammen als „Leistungsquellen“ aus dem Komponenten / Quellen-Menü. Die Gleichungen zur Berechnung der S-Parameter S11und S21 sowie zum Phasenverlauf bei S21 wurden bereits automatisch von qucsstudio vorbereitet.

Allerdings nehmen wir noch einige Änderungen bei der Sweep-Einstellung vor (rechter Mausklick und „Eigenschaften editieren“) und **arbeiten mit einem linearen Sweep von 1kHz…20MHz (bei 262144 Punkten).**

(Wer bei Null beginnen möchte, erlebt leider einen Protestmarsch des Programms….)



So sollten wir das Ergebnis hinkriegen…..



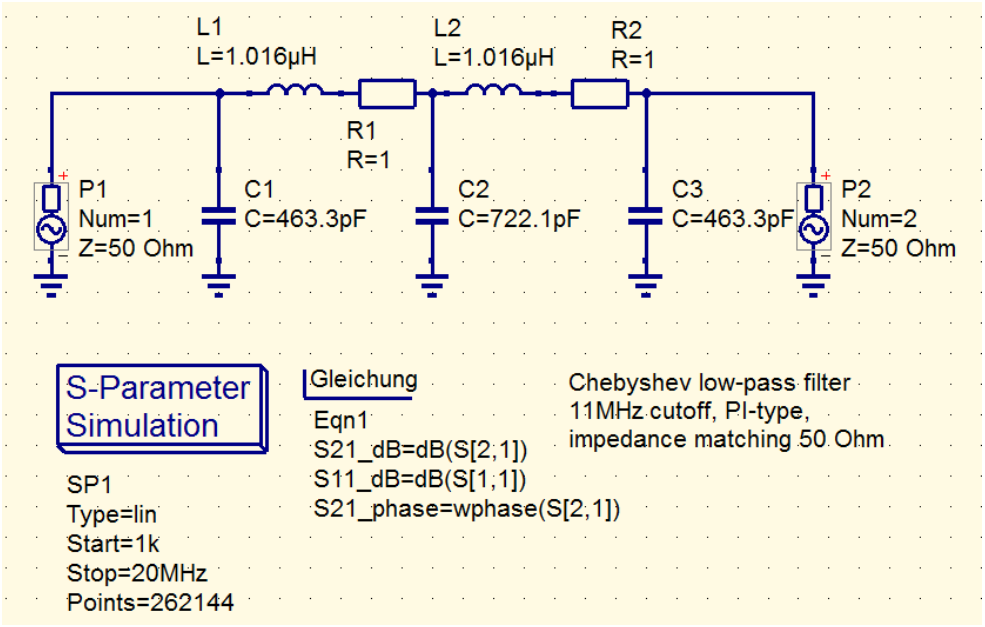
…und anschließend nochmals soweit dehnen, dass wir die Tschebyschef-Wellen von S21 zu sehen bekommen:

==========

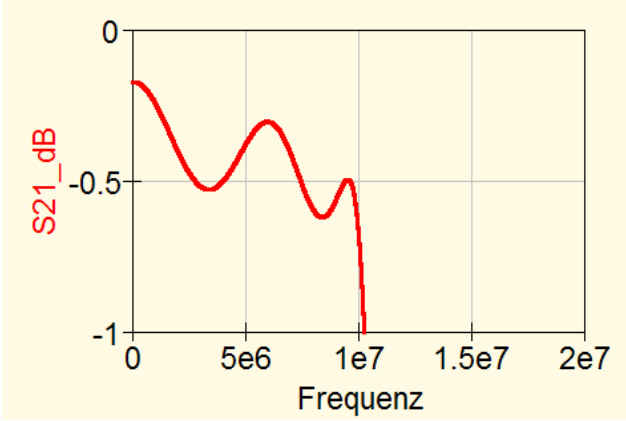
9.2.4. Berücksichtigung der Spulenverluste

Im Pflichtenheft am Anfang des letzten Kapitels stand „**Gemessene Spulengüte Q = 70 bei 11 MHz**“. Diese Verluste in den Spulen lassen sich jeweils durch einen zusätzlichen Reihenwiderstand in die Simulation einbringen: Sein Wert beträgt:

$R_r = X_L / Q = 2\pi fL / Q = 2\pi * 11MHz * 1{,}016\mu H / 70 = 1\Omega$



Bei jeder Spule wird jetzt ein solcher Widerstand im Schaltbild eingetragen und nochmals simuliert.



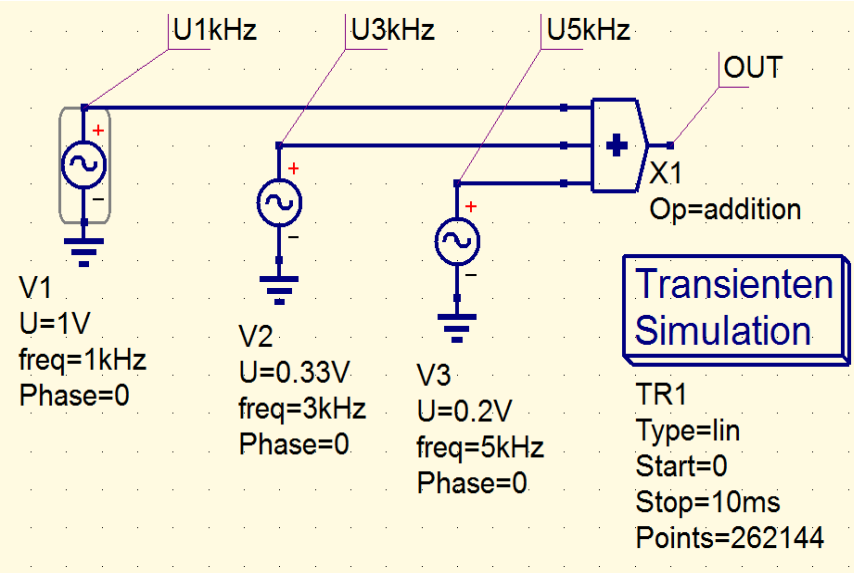
Das Ergebnis sollte man sich merken:

Die Spulenverluste führen zu einer zusätzlichen „Grunddämpfung“ im Durchlassbereich, die mit der Frequenz ansteigt.

(In der Nähe der Durchlassgrenze sind das bereits 0,5 dB…)

9.2.5. Die Sache mit der Group Delay (= Gruppenlaufzeit)

Bei der Signalübertragung haben wir es selten mit einem einzigen rein sinusförmigen Signal zu tun: entweder sind es modulierte Signale (AM, FM, QAM usw.) oder ein Signalgemisch (Sprache, Musik, Videosignale usw.) oder pulsförmige 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.

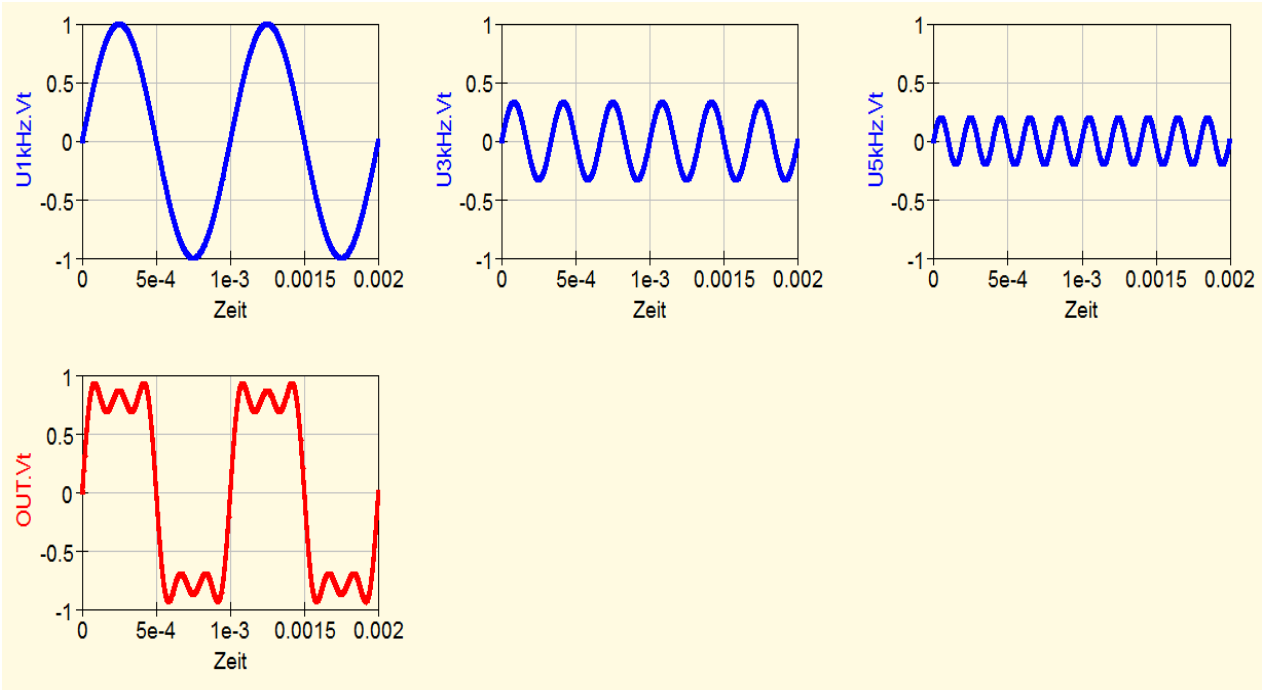
Sehen wir uns das mal an einer **Simulation** an, bei der die ersten drei Spektrallinien eines symmetrischen Rechtecksignals zusammenaddiert werden.

Es handelt sich um

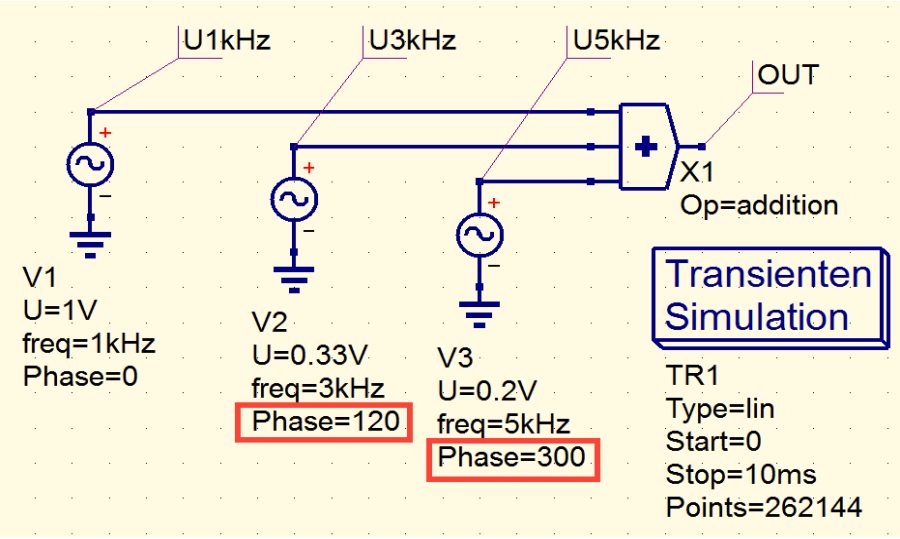
a) die Grundwelle mit U1kHz= 1V und der Frequenz f = 1kHz
b) die dritte Harmonische mit U3kHz = 1/3 x (U1kHz) und der Frequenz f = 3 x f1 = 3kHz
c) die fünfte Harmonische mit U5kHz = 1/5 x (U1kHz) und der Frequenz f = 5 x f1 = 5kHz

Alle Signale weisen dieselbe „Nullphase“ auf.

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

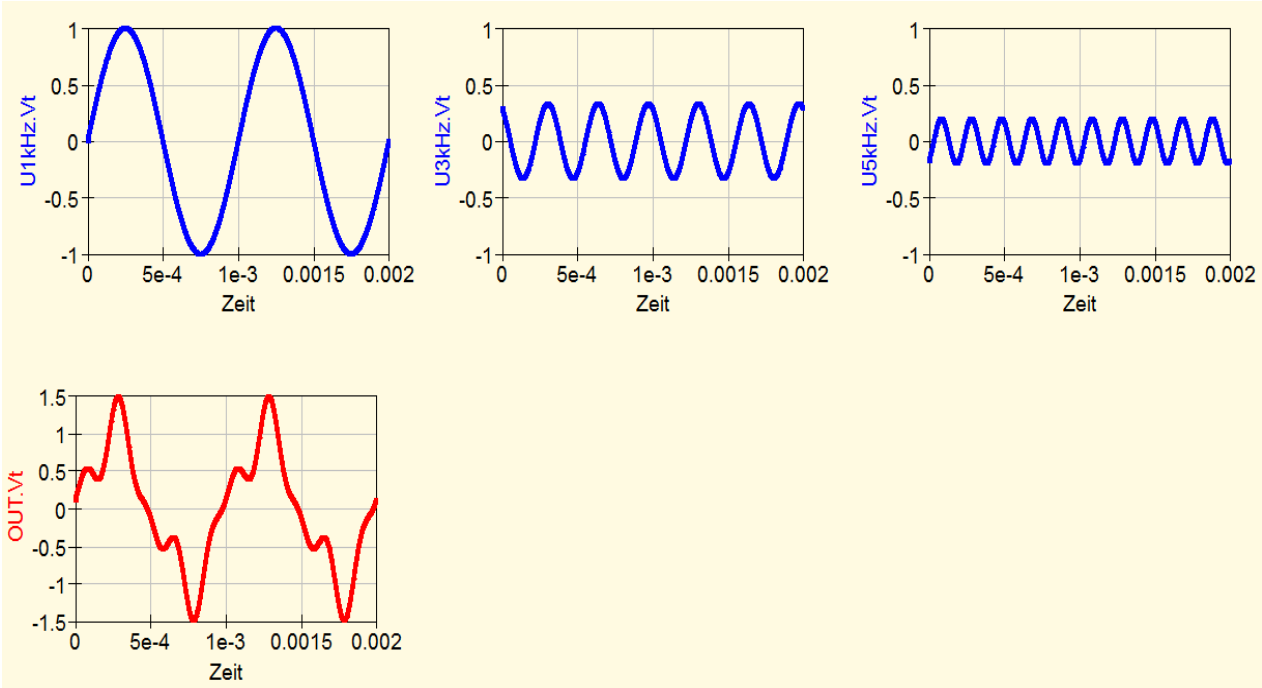


Trotz des geringen Oberwellen-Anteils ist der grundsätzliche Kurvenverlauf des Rechtecks bei der Summenspannung eindeutig zu erkennen.



Nun lassen wir die Amplituden der Harmonischen unverändert, **verschieben aber ihre Phasenlagen** gegeneinander. Dazu lassen wir die die dritte Harmonische mit -120 Grad“, die fünfte Harmonische dagegen mit -200 Grad beginnen:

Das führt zu einer totalen Veränderung der Kurvenform, bei der das ursprüngliche Rechtecksignal fast nicht mehr zu erkennen ist:

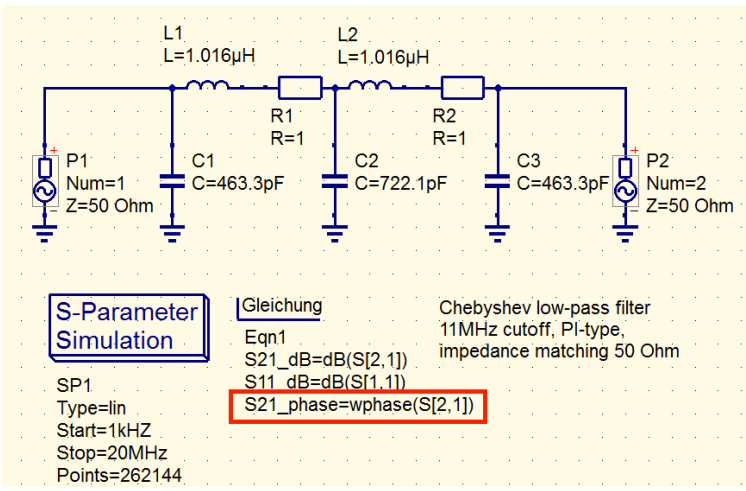


Also heißt die Forderung:
Durchläuft ein solches „Gruppensignal“ einen Baustein oder eine Übertragungsstrecke, dann dürfen zwar **Zeitverzögerungen** auftreten, aber die **ursprünglichen Phasenlagen (der Einzelsignale untereinander) müssen am Ausgang noch korrekt stimmen.**

Die korrekte Formulierung lautet damit:

Eine korrekte 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.

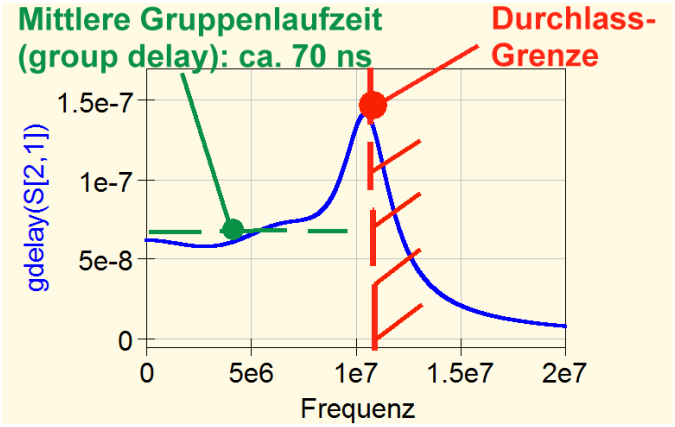


Nun wollen wir wieder unserem aktuellen Tiefpass mit Verlusten (aus dem vorigen Kapitel) auf den Zahn fühlen und das eben Besprochene an diesem Beispiel demonstrieren.

Entscheidend ist, dass bereits die Gleichung für die Bestimmung der Phase von S21

S21_phase=wphase(S[2,1])

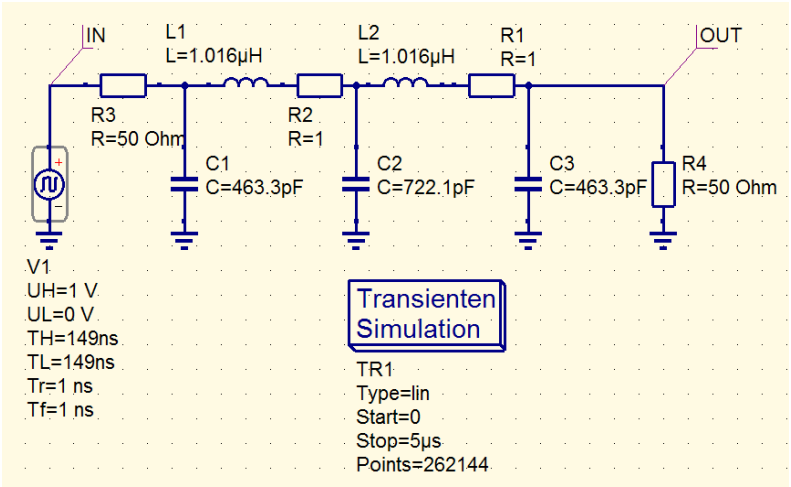
vorhanden ist, denn aus dem Phasenverlauf wird nach der Simulation die Gruppenlaufzeit berechnet.



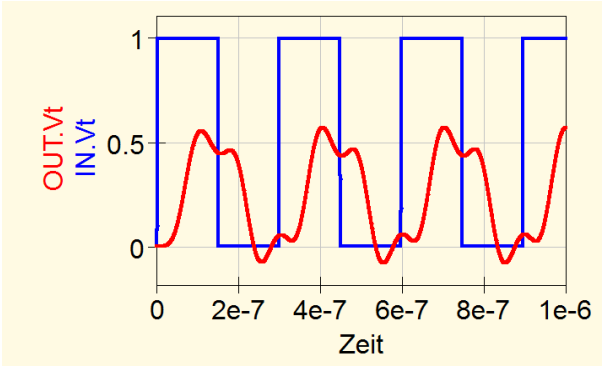
Dazu holt man sich ein kartesisches Diagramm auf die Arbeitsfläche und trägt bei den Eigenschaften des neuen Graphen

gdelay(S[2,1])

ein. Dann erhält man (nach „Anwenden“) das nebenstehende Diagramm.
Man erkennt, dass die group delay etwa bis zu 75% der Durchlass-Grenze nur um ca. 10 ns um den Mittelwert von 70ns herum schwankt. Ab da steigt sie massiv an.



Doch wir wollen uns sofort ansehen, wie sich das in der Praxis auswirkt. Wir speisen dazu den obigen TP mit einem Pulssignal, das eine Periodendauer von 300 ns aufweist (… das ergibt etwa fpulse = 3,33 MHz) und sehen uns die Ausgangsspannung an.



Dieses Bild ist fast selbst erklärend...

10. Das Smith-Diagramm („Smith chart“)

10.1. Problemstellung

In bisherigen Diagrammen von simulierten S-Parametern sahen wir immer nur den Verlauf der „Magnitude“, also des Betrags eines S-Parameters. Wenn man jedoch z. B. die Eingangsreflektion S11 oder die Ausgangsreflektion S22 verbessern möchte, möchte man genau die Gründe für die Abweichungen vom Systemwiderstand Z = 50Ω wissen -- und dafür sind meist kapazitive oder induktive Blindanteile verantwortlich. Also brauchen wir ein Diagramm, das sowohl beim Wirkwiderstand wie auch bei den Blindwiderständen die Darstellung von

Null bis Unendlich

ermöglicht.

Das erreicht man durch eine „**Koordiaten-Transformation**“ und die führt zum Smith-Diagramm (= Smith chart).

10.2. Wie entsteht ein Smith Chart?

Erinnerung:
Wenn man einen Wirkwiderstand und einen Blindwiderstand in Reihe schaltet, so muss man die

Phasenverschiebungen zwischen Strom und Spannung beim Blindwiderstand beachten und darf die beiden

nur geometrisch (= rechtwinklig) addieren.

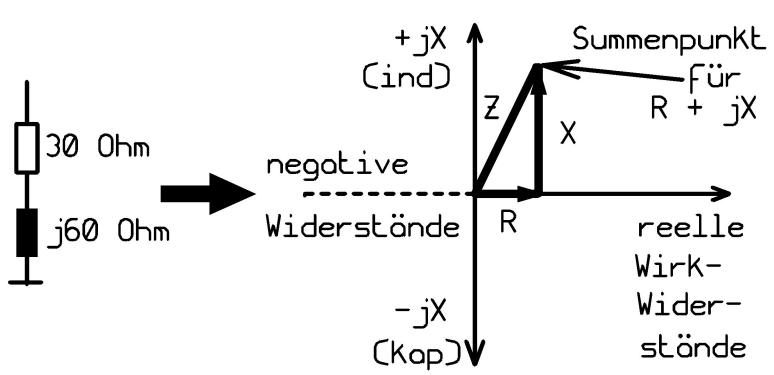
(Kennen Sie noch oder schon die beiden Sätze: „...beim Kondensa**tor** eilt der Strom **vor...**“
bzw. ....bei der **Induktivität** kommt der Strom zu **spät...**“? Die gehören nun hierher)

Nehmen wir mal als Beispiel einen Ohmschen Widerstand mit 30 Ohm und schalten ihm eine Spule in Reihe, deren Blindwiderstand bei der Arbeitsfrequenz gerade 60 Ohm beträgt.
Man kann nun ein System mit einem Achsenkreuz vorsehen, bei der die **Ohmschen Widerstände** in **waagrechter Richtung** einzutragen sind. Das ist dann die „**reelle Achse**“ und weil wir im Normalfall keine negativen Widerstände zur Verfügung haben, reicht der **waagrechte Teil rechts vom Nullpunkt** dafür aus.

Induktive Blindwiderstände werden an der **senkrechten Achse** eingetragen, die vom **Nullpunkt nach oben** zeigt, **Kapazitive Blindwiderstände** dagegen in **senkrechter Richtung vom Nullpunkt nach unten**. Das ist dann die **„imaginäre Achse“** und diese Blindwiderstände erhalten den Buchstaben

„+j“ (bei Spulen) oder **„-j“ (bei Kondensatoren)**

zur Unterscheidung vor ihren Widerstandswert gesetzt:



Den Anwender interessiert der **Summenpunkt (= die Zeigerlänge = der komplexe Gesamtwiderstand)**) und den können wir auf zwei Arten angeben:
Entweder als Summe aus Realteil (= 30 Ω) und Imaginärteil (= j60 Ω)

oder

als „Magnitude“ und Phase“.

Für die Kommunikationstechnik ist die Darstellung mit „linearer Teilung der Achsen“ nicht gut geeignet, denn wir haben es dort mit Widerstandswerten zwischen Null Ohm (= Kurzschluss) über Z = 50 Ohm (also: Anpassung) bis hin zu „Unendlich“ (= Leerlauf) zu tun, wobei auch noch die Übersichtlichkeit gewahrt sein sollte.

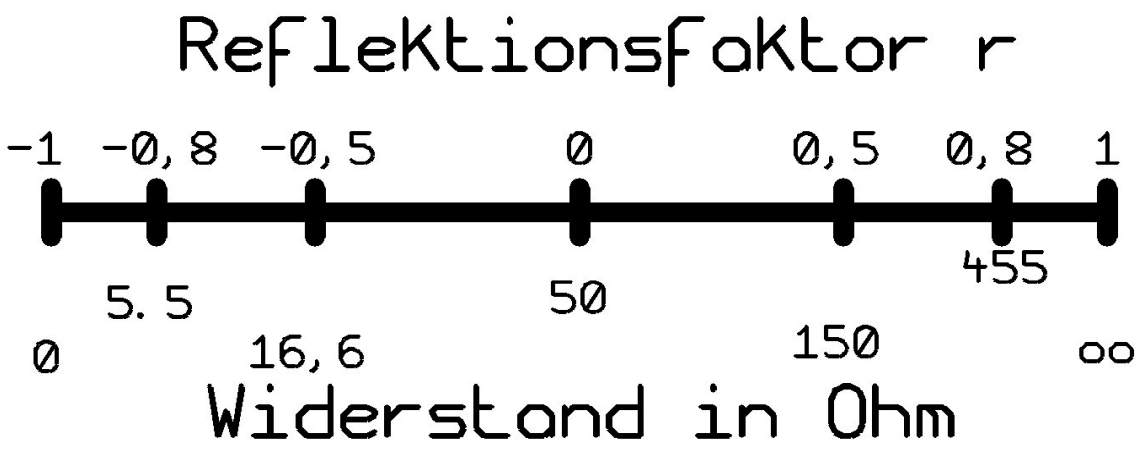
Deshalb ist es viel vernünftiger, ein Diagramm zu verwenden, das den **Reflektionsfaktor „r“**

$$r = \frac{Z_{Last} - Z}{Z_{Last} + Z}$$

als Achsenteilung verwendet und tragen die zugehörigen Widerstandswerte mal an der **reellen Achse** ein.

So sieht nun die waagrechte Achsenteilung aus:

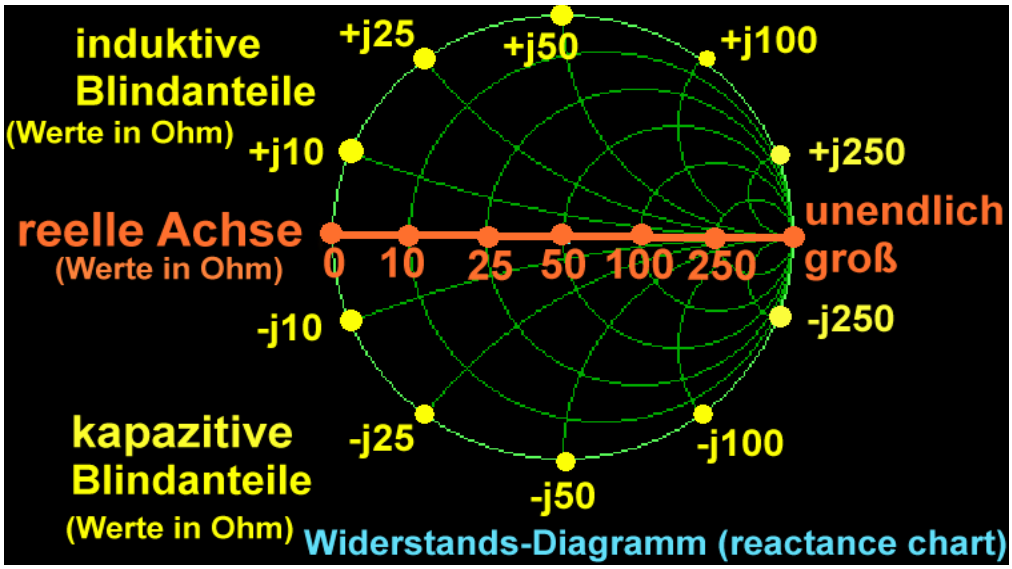
Reflektionsfaktor „r“ linear geteilt von „-1“ über „0“ bis „+1“



Widerstandswert reicht nun von „Null“ bis „Unendlich“

Das ist die Transformation für einen Systemwiderstand von 50 Ω und deshalb finden wir in der Mitte für „keine Reflektion, also r = Null“ genau einen Widerstandswert von 50Ω.
Der Wertebereich des Reflektionsfaktors r reicht dann von „-1“ (bei Kurzschluss) über „Null“ (für Anpassung) bis „+1“ (für einen Leerlauf, also Unendlich) -- und genau das brauchen wir!

Wichtig:
Blindwiderstände (die ja positives Vorzeichen für Induktivitäten, aber negatives Vorzeichen für Kapazitäten aufweisen) werden wegen der Koordinatentransformation nun nicht mehr als Geraden, sondern als **Kreise** dargestellt. Sehen wir uns das doch mal genau an:



Da muss man sich nun das Folgende für dieses **Widerstands-Diagramm** gut einprägen:

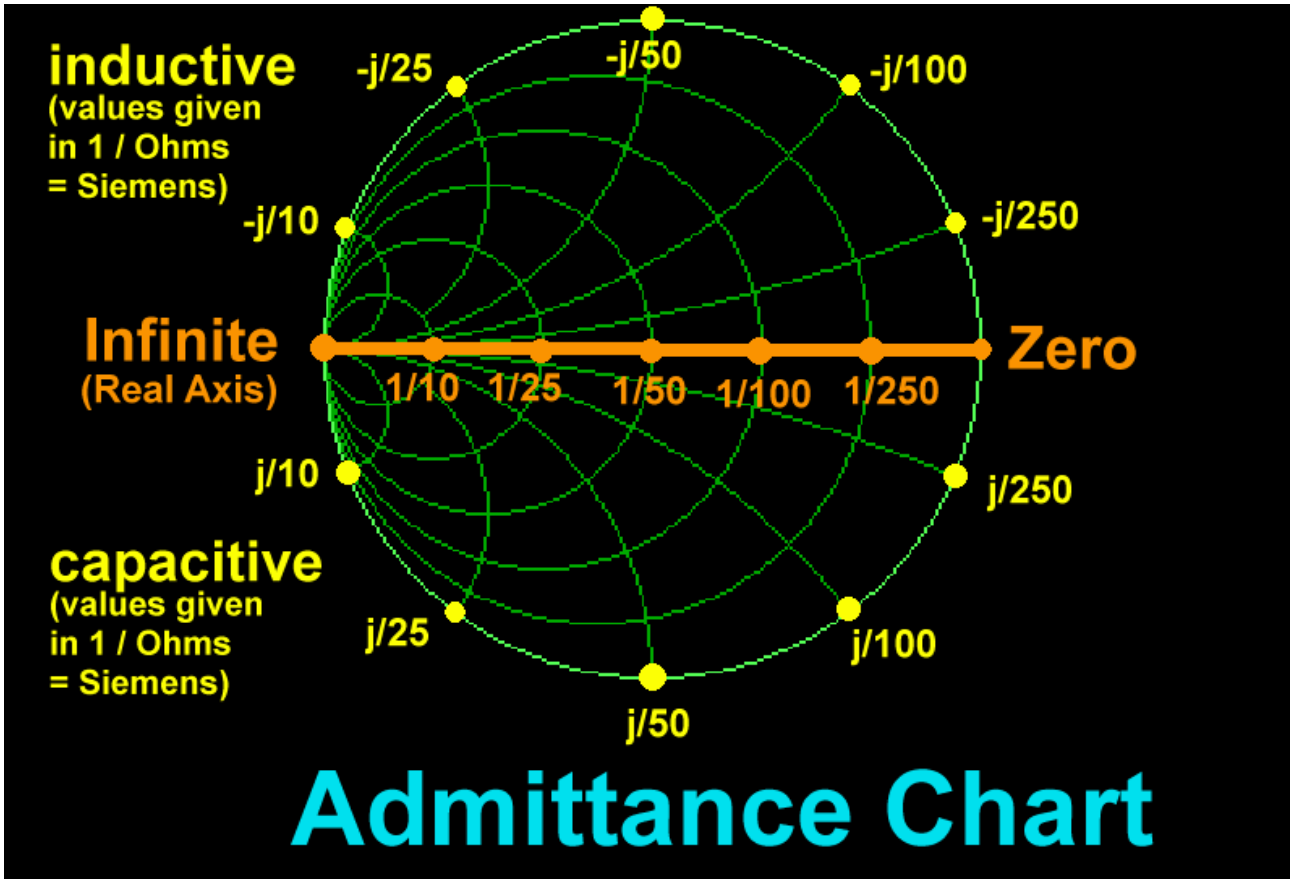
a) **Der Diagramm-Mittelpunkt entspricht genau dem Systemwiderstand von 50Ω -- also der perfekten Anpassung ohne Reflektionen. Das ist meist das Ziel des Schaltungsentwicklers….**

b) **Den Anfang der reellen Achse bildet ein Widerstand mit Null Ohm -- das ist der Kurzschluss!**

c) **Das Ende der reellen Achse auf der rechten Seite stellt den Leerlauf mit einem unendlich hohen Widerstand dar.**

Achtung:
Bei jedem beliebigen Punkt in diesem Widerstands-Diagramm hat man es immer mit einer Reihenschaltung aus Wirkwiderstand und Blindwiderstand zu tun!

Oft benötigt man jedoch eine **Parallelschaltung von Wirk- und Blindanteil**. Dafür gibt es das Leitwerts-Diagramm (A**dmittance Chart**). **Bei ihm ist alles in waagrechter Richtung gespiegelt, aber die Darstellung des Kurvenverlaufs eines S-Parameters (wie S11 oder S22) ändert sich nicht!** (...denn „Kurzschluss“ an der linken Seite und „Leerlauf“ an der rechten Seite behalten ihre Positionen bei).



==========

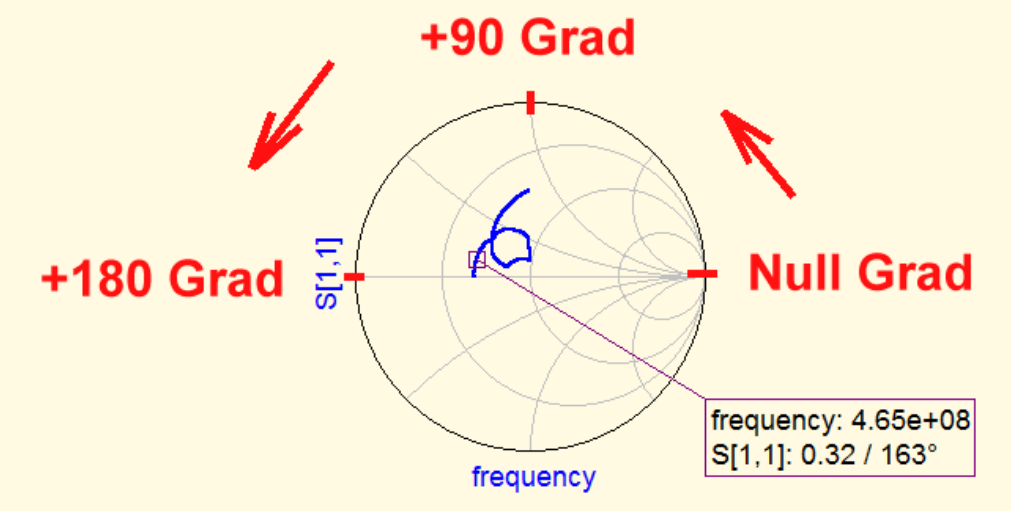
Eine putzige Sache ergibt sich dann natürlich, wenn man z. B. Anpass-Schaltungen aus einer Mischung von parallelen und in Reihe geschalteten Bauteilen entwirft. Da muss man zwar wie ein Karnickel zwischen den beiden Diagrammformen hin und herspringen, aber dann bewegt man sich beim Hinzufügen eines neuen Bauteils immer auf einem Kreis. Ist mit etwas Übung nicht schwierig und sehr effektiv – aber eine eigene Welt...

10.3. Kleine Tücken bei den Winkelangaben im Smithchart

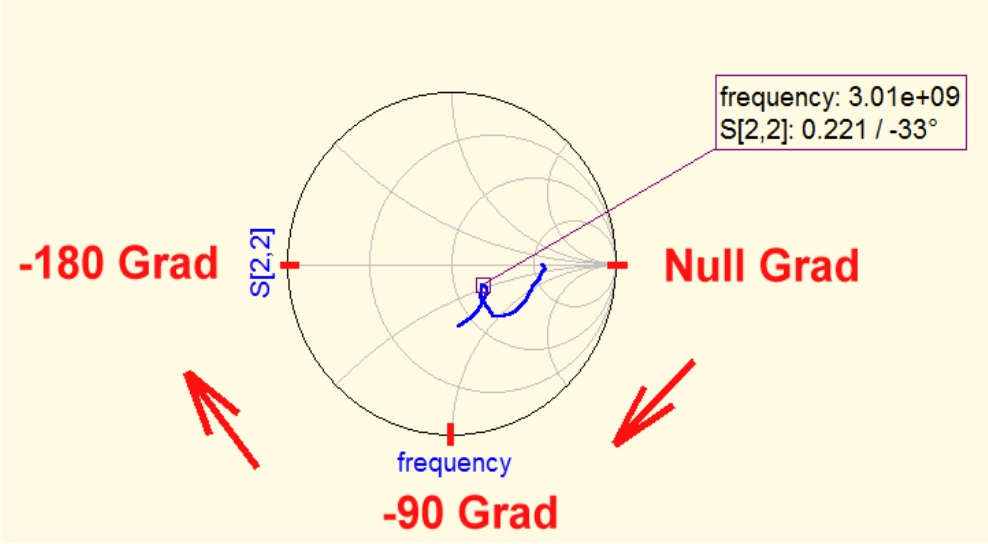
Da kommt man am Anfang leicht durcheinander, denn wir haben mit dem Smithchart ein Diagramm vor uns, bei dem zwei Dinge miteinander verknotet wurden:

a) Die zu einer Eingangs- oder Ausgangsreflektion (= S11 bzw. S22) gerhörenden **Impedanzen** sind in der **oberen Diagrammhälfte** immer **induktiv** und deshalb kann es bei der Angabe als „Magnitude und Phase“ nur Werte zwischen **Null und +90 Grad** geben.
Bei **kapazitivem** Verhalten befinden wir uns in der **unteren** Diagrammhälfte und da gibt es nur Phasenwinkel zwischen **Null und -90 Grad.**

b) Die Grundstruktur und **die Anzeige** des Smithcharts basiert aber auf dem **Reflektionsfaktor „r“** und da muss man deutlich aufpassen:



So sieht das für einen dargestellten **S Parameter „S11“** aus. Auch hier arbeitet man mit Betrag und Phase, **aber der Nullpunkt liegt rechts und der Phasenwinkel wird von dort aus positiv gegen den Uhrzeigersinn von Null bis +180 Grad gezählt.**
Also haben wir es – wenn man die S11-Werte später in Impedanzen umrechnet – mit induktiven Blindanteilen beim Eingangswiderstand zu tun.



Hier zum Vergleich eine Ausgangsreflektion S22 für einen MMIC.
Sie verläuft im unteren Teil des Diagramms und deshalb haben wir kapazitives Verhalten.

Die Phase von „r“ (bzw. hier: S22) wird in diesem Teil von Null bis -180 Grad gezählt und die Zählrichtung bzw. Anzeige ist der Uhrzeigersinn.

Die Frage „Wie kann ich die Impedanz für einen bestimmten Punkt im Diagramm ablesen?“ werden wir gleich im nächsten Kapitel beantworten.

10.4. Wie erhalte ich die Impedanz als Wirk- und Blindwiderstand (= Serienschaltung) bei einem Diagramm-Punkt im Smithchart?

Dazu nehmen wir uns einen bekannten und klassischen MMIC als Beispiel vor, den es schon lange gibt (= INA03184), und simulieren die S-Parameter. Dazu findet man im Internet in der Agilent-Avago-Homepage folgende Datei mit dem Namen „**a03184ia.s2p**“:

==========

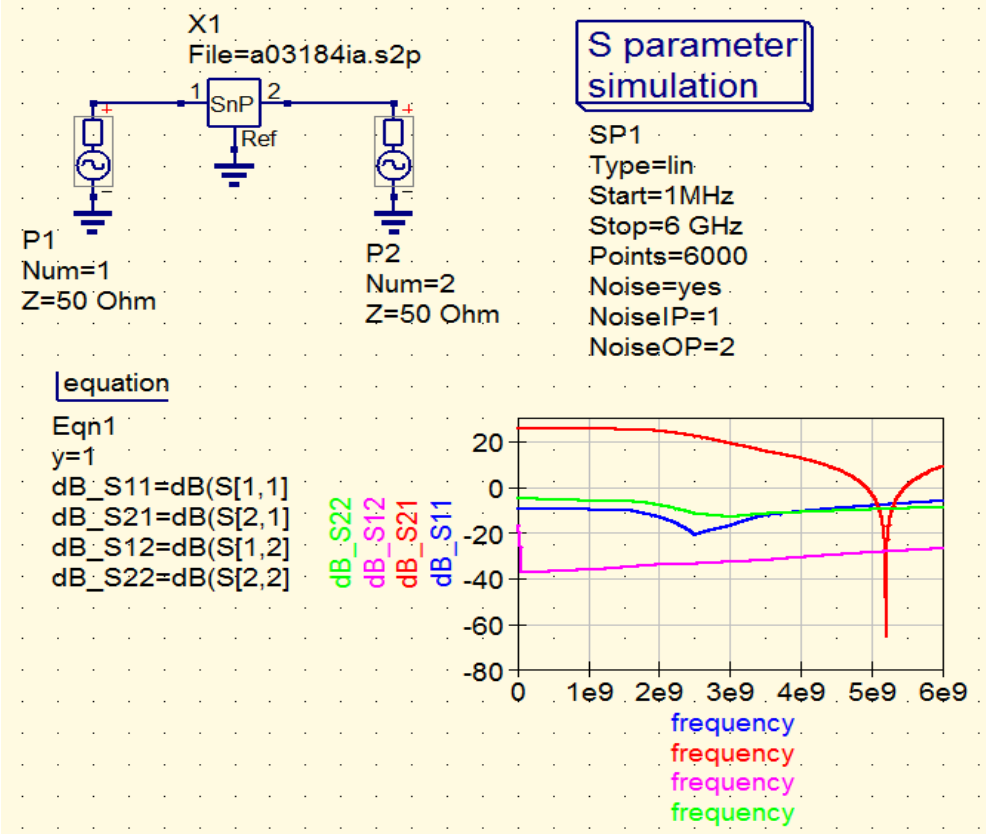
! INA-03184 S PARAMETERS
! Id = 10 mA LAST UPDATED 07-22-92

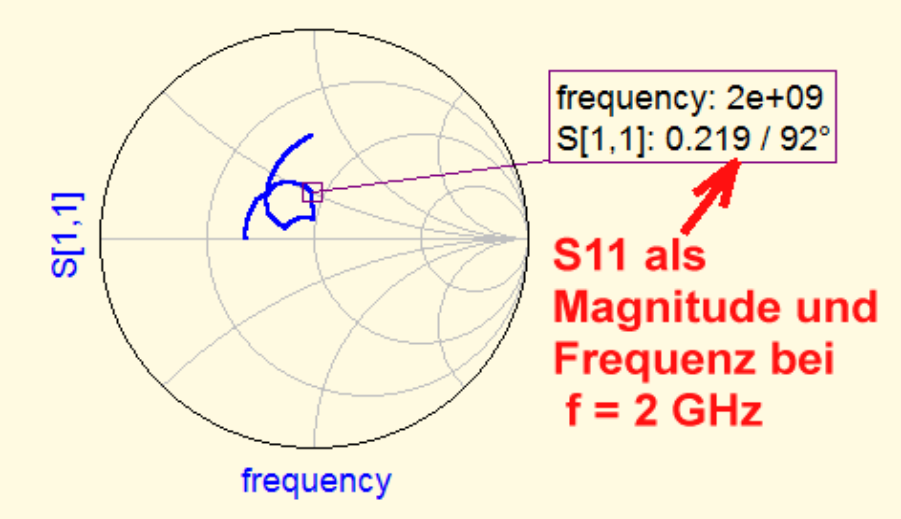
 # ghz S ma r 50

0	**.32**	**180**	**19.2**	**0**	**0.14**	**0**	**0.55**	**0**
0.05	**.32**	**179**	**19.14**	**-3**	**.014**	**3**	**.55**	**0**
0.10	**.32**	**176**	**19.05**	**-7**	**.014**	**4**	**.57**	**-3**
0.20	**.32**	**172**	**19.05**	**-14**	**.014**	**6**	**.55**	**-5**
0.40	**.32**	**165**	**18.78**	**-29**	**.014**	**10**	**.53**	**-11**
0.60	**.32**	**158**	**18.71**	**-43**	**.015**	**11**	**.51**	**-14**
0.80	**.32**	**151**	**18.53**	**-57**	**.015**	**13**	**.51**	**-17**
1.00	**.32**	**144**	**18.18**	**-72**	**.016**	**21**	**.50**	**-20**
1.20	**.30**	**135**	**18.27**	**-86**	**.016**	**25**	**.50**	**-23**
1.40	**.31**	**126**	**18.10**	**-102**	**.017**	**30**	**.49**	**-29**
1.60	**.30**	**117**	**17.92**	**-117**	**.018**	**38**	**.48**	**-34**
1.80	**.26**	**102**	**17.49**	**-135**	**.019**	**44**	**.45**	**-41**
2.00	**.22**	**92**	**16.62**	**-153**	**.020**	**49**	**.40**	**-50**
2.50	**.09**	**91**	**12.88**	**168**	**.021**	**57**	**.26**	**-48**
3.00	**.14**	**160**	**8.79**	**134**	**.023**	**65**	**.22**	**-33**
3.50	**.24**	**151**	**5.92**	**108**	**.025**	**69**	**.26**	**-33**
4.00	**.29**	**139**	**4.18**	**87**	**.029**	**81**	**.28**	**-43**

==========

Sie wird entweder von dort geholt oder man kopiert sie aus der obigen Liste in ein Textfile und speichert das als „a03184ia.s2p“ im zugehörigen qucsstudio-Projektordner (….**Achtung: nicht vergessen, auf „ASCII“ und „Alle Dateien“ einzustellen.**..). Dann führen wir eine S-Parameter-Simulation im Bereich von 0....6 GHz durch.





Dann öffnen wir ein **„Impedanz-Smithchart“** und lassen uns darin **S11** darstellen. Mit einem **Frequenzmarker** kennzeichnen wir den Punkt für eine **Frequenz von f = 2 GHz.**
(Der Marker lässt sich nach einem linken Mausklick markieren und dann entweder mit den waagrechten Pfeiltasten an der Kurve entlang schieben oder bei gedrückter linker Maustaste zum gewünschten Frequenzpunkt „ziehen“).

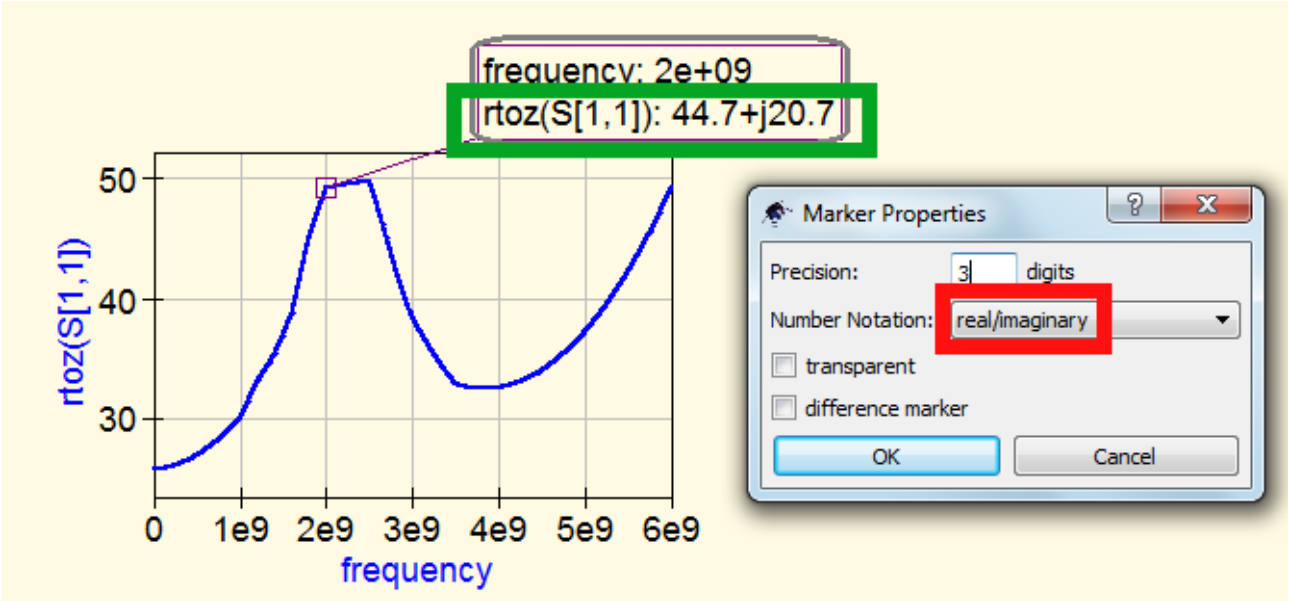
Anschließend öffnet man zusätzlich ein kartesisches Diagramm und schreibt in die Zeile für „Graph Properties“:

rtoz(S[1,1])

Auch auf dieser Kurve positionieren wir einen Marker, markieren ihn durch einen linken Mausklick und schieben ihn auf f = 2 GHz. Anschließend klicken wir „mit rechts“ auf das Markerfeld, um die Properties zu editieren. Darin stellen wir unter „number notation“ auf „**real / imaginary**“ und erhalten dann nach OK die Ersatz-Serienschaltung der Eingangsimpedanz zu

44,7Ω + j20,7Ω

und das bedeutet: **44,7Ω in Reihe mit 1,65 nH.**



10.5. Wie erhalte ich den Wirk- und Blindanteil des Eingangs-Leitwertes (= Parallelschaltung = Admittanz) bei einem Diagramm-Punkt im Smithchart?

Aufgabe:
Stellen Sie nun S11 in einem Admittanz-Smithchart (= Leitwerts-Diagramm) dar.

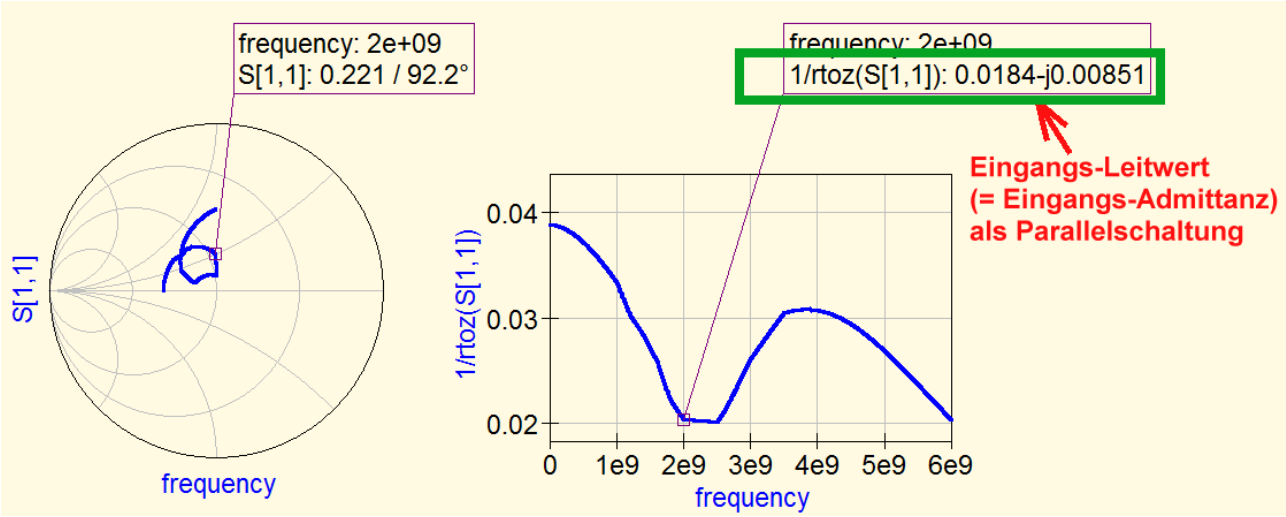
Markieren Sie darin den Punkt für S11 bei f = 2 GHz

Bestimmen Sie nun den Wirk- und Blindleitwert dieser Parallelschaltung.

Lösung:
Zuerst holt man sich ein Admittanz-Smithchart aus dem Diagramm-Vorrat und lässt darin S11 anzeigen.
Dann wird ein Marker auf die Kurve gesetzt und auf f = 2 GHz geschoben.

Anschließend brauchen wir ein kartesisches Diagramm und sorgen dafür, dass darin die **Admittanz als Kehrwert der Impedanz** präsentiert wird. Das erfolgt durch folgende Zeile bei „Graph Properties“:

1/rtoz(S[1,1])



Aus dem rechten Bild entnimmt man:

Die Eingangsimpedanz kann so bei f = 2 GHz auch als **Parallelschaltung** eines

Wirkleitwertes mit 0,0184 Siemens (= 54,34Ω)

und

eines induktiven Blindleitwerts von -j0,00851 Siemens dargestellt werden. Der entspricht bei f = 2 GHz einem induktiven Widerstand von 117,5Ω und einer Induktivität von 9,35 nH.

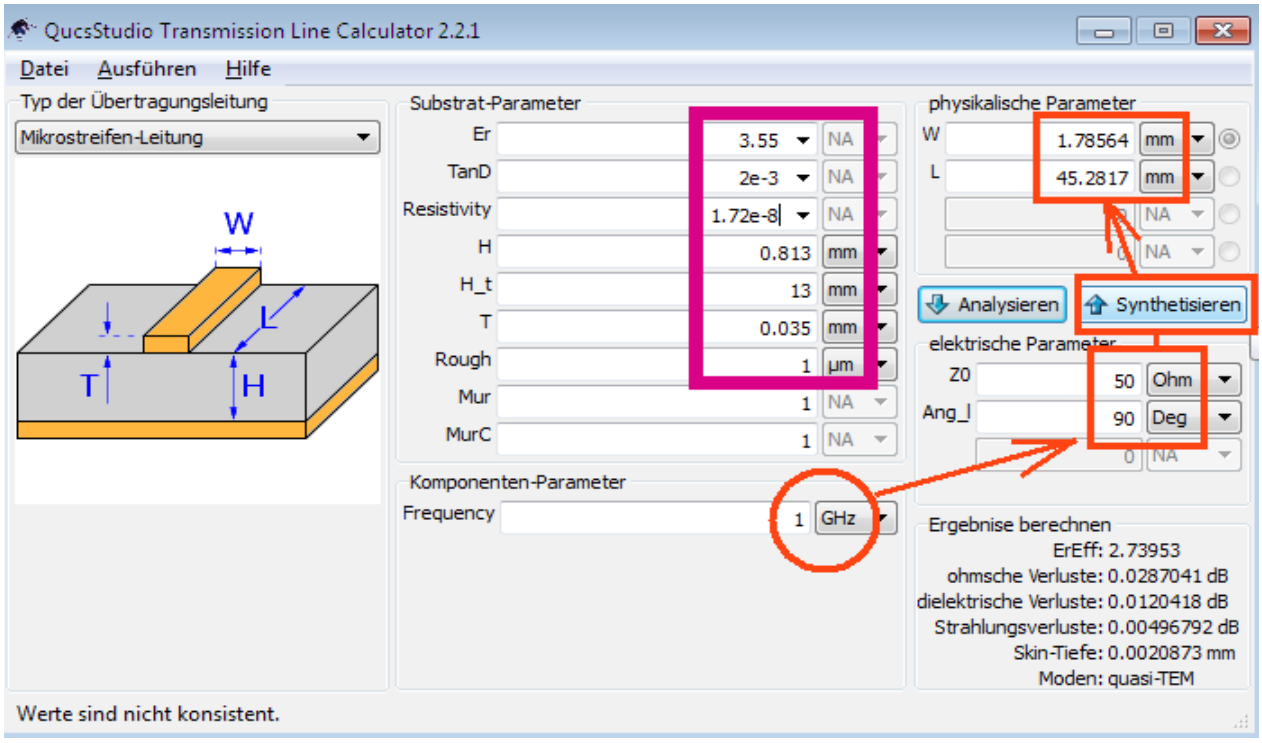
Bitte nachmachen und nachrechnen...

11. Microstrips (= Streifenleitungen)

Für die korrekte Signalübertragung sind wir auf HF-Leitungen mit dem richtigen Wellenwiderstand (üblich: Z = 50Ω) angewiesen. Da jedoch die HF-Schaltungen stets auf Leiterplatten (PCBs) aufgebaut werden, genügt dort ein Leiterbahn mit einer bestimmten Breite, um eine solche 50Ω - „Microstrip“ zu realisieren. Die Unterseite des Boards ist dabei eine **durchgehende Metallfläche, die als „Ground Plane“** (= Masse – Ebene) dient. Für die erforderliche Breite dieser Leitung sind natürlich die Platinendicke und die Werkstoffdaten maßgebend -- aber dafür haben wir unter „**Werkzeuge**“ und „**Leitungsberechnung**“ einen hübschen Calculator, der die Arbeit übernimmt.

Ihr Autor arbeitet meist mit dem bewährten hochwertigen und trotzdem preisgünstigen Platinenmaterial „RO4003“ der Firma Rogers. Da sind beim Calculator folgende Einträge erforderlich:

Dielektrizitätskonstante	**Er = 3.55**
Verlustfaktor	**TAND = 0.002**
Spezifischer Widerstand (resistivity) des verwendete Kupfers:	**1,72e-8**
Platinendicke	**H = 32 mil = 0,813mm**
Deckelabstand zur Platine	**H_t = 13mm**
Metallisierung	**Kupfer mit Dicke T= 0,035mm**
Rauhigkeit der Cu-Leiterbahn	**Rough = 1µm**



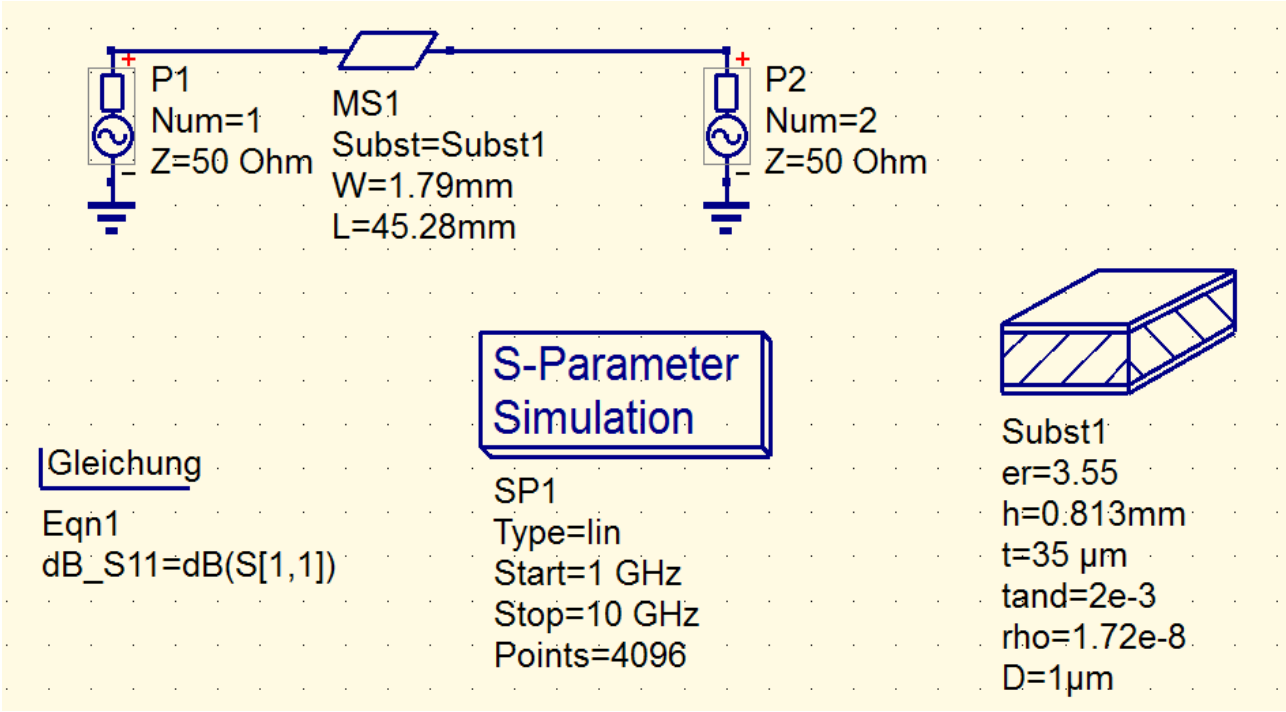
Erst wird kontrolliert, ob der Calculator auf „**Mikrostreifen-Leitung**“ eingestellt ist. Anschließend übernimmt man der Reihe nach die obigen Werte in den violett markierten Rahmen.

Dann stellt man als **Designfrequenz „1 GHz“** ein, gefolgt vom gewünschten Widerstand **Z = 50Ω** und einer elektrischen Leitungslänge von **90 Grad**. Ein Klick auf „**Synthetisieren**“ liefert die gewünschten Leitungsdaten:

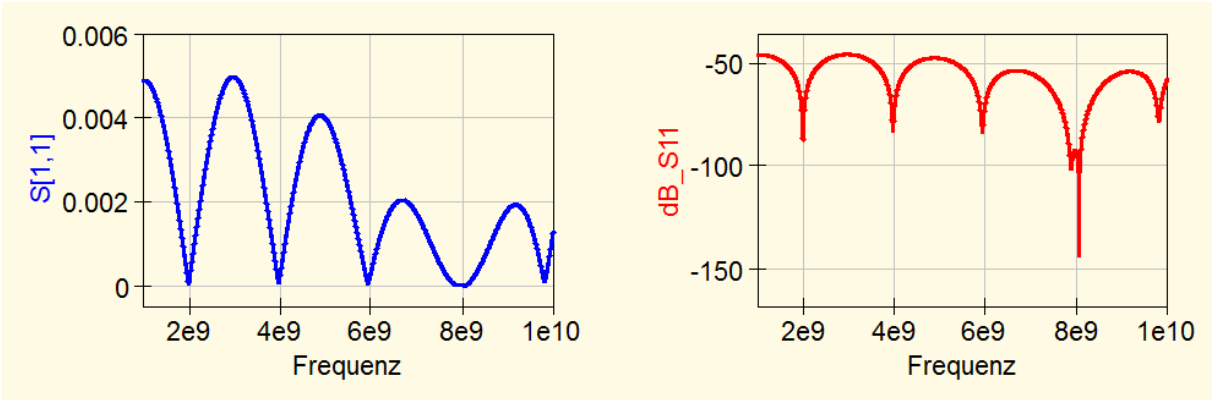
Leiterbreite	**W = 1,79 mm (…genauer kann man nicht ätzen…)**
Leitungslänge für 90 Grad	**L = 45,28 mm bei 1 GHz**

Anschließend sollte man sich (…Vertrauen ist gut, Kontrolle ist besser…) die Mühe machen, dieses Ergebnis durch eine kurze Simulation zu prüfen. Dazu speist man die Leitung mit einem Port, schließt sie am Ende mit 50Ω ab und simuliert anschließend S11 (linear und in dB).
Dann braucht man natürlich die Komponenten „**Mikrostreifen-Leitung MS1**“ und „**Substrat**“,deren Menüs ausgefüllt werden müssen. Sie finden sich in „**Komponenten / Übertragungsleitungen**“.

Warnung:
Dabei gibt es beim Ausfüllen der Substrat-Daten eine hübsche kleine Falle, an der man am Anfang fast verzweifelt:
Man klickt rechts auf das Substratsymbol und wählt „Eigenschaften editieren“. Soweit stimmt alles noch.
Dann ruft man die erste Eigenschaft (= Dielektrizitätskonstante er) auf und stellt auf er=3,55 ein.
Aber jetzt darf man nicht (wie gewohnt) auf „Anwenden“ klicken, sondern MUSS diesen neuen Wert mit der Eingabetaste („Enter“) bestätigen. Nur so wird er übernommen und es wird auch gleich auf die nächste Zeile weitergeschaltet. Muss man wissen…



So sieht das Ergebnis aus und damit wollen wir zufrieden sein, denn im linken Bild sieht man, dass die maximale Reflektion nicht über 0,5% hinausgeht…



12. Praxis-Projekt: MMIC-Breitbandverstärker bis 3 GHz

12.1. Touchstone-Files (S2P-Files)

Jede HF-Halbleiter-Herstellerfirma hält für ihre Bauteile solche Files für die Anwender bereit. Dabei handelt es sich um Datensätze mit S-Parametern, die von einem Vektoriellen Network-Analyser im angegebenen Arbeitspunkt (und im „Kleinsignal-Betrieb“) direkt am Bauteil gemessen wurden.

Ein MMIC ist ein „**Monolithic Microwave Integrated Circuit**“, also ein vollständig integrierter und lauffähiger Verstärker. Er stellt ein „**Zweitor“ (Twoport)** mit einem Eingang und einem Ausgang dar, deshalb werden im Touchstone-File die Parameter

S11 / S21 / S12 / S22

genau in dieser Reihenfolge für steigende Frequenz aufgelistet. Sehen wir uns das für den vorgesehenen ABA52563 mal an. Dieses File findet sich am Ende des ABA52563-Datenblattes und muss dann durch Herauskopieren, Einfügen in ein neues Textfile und Nachbearbeitung am PC in folgende Form gebracht werden:

! ABA-52563 S PARAMETERS

 # ghz s ma r 50

0.05	**0.01 146.6**	**12.10 -2.6**	**0.03 0.3**	**0.15 -2.4**
0.10	**0.01 134.0**	**12.11 -4.8**	**0.03 -0.3**	**0.15 -5.1**
0.20	**0.01 -40.6**	**12.16 -9.6**	**0.03 0.1**	**0.15 -9.6**
0.30	**0.01 -53.2**	**12.19 -14.5**	**0.03 1.2**	**0.15 -13.0**
0.40	**0.02 -56.7**	**12.19 -19.5**	**0.03 2.4**	**0.14 -15.7**
0.50	**0.03 -141.5**	**12.26 -24.8**	**0.03 1.0**	**0.15 -15.7**
0.60	**0.03 -128.1**	**12.24 -29.8**	**0.03 3.1**	**0.15 -17.6**
0.70	**0.04 -127.5**	**12.21 -34.9**	**0.03 4.3**	**0.15 -20.3**
0.80	**0.04 -126.7**	**12.18 -39.8**	**0.03 6.1**	**0.15 -22.5**
0.90	**0.05 -123.9**	**12.16 -44.7**	**0.03 7.4**	**0.15 -24.2**
1.00	**0.05 -125.0**	**12.13 -49.7**	**0.03 11.7**	**0.15 -26.4**
1.20	**0.05 -123.4**	**12.10 -59.6**	**0.03 10.8**	**0.15 -29.4**
1.40	**0.06 -127.4**	**12.05 -69.4**	**0.03 12.4**	**0.15 -32.4**
1.60	**0.06 -133.8**	**12.04 -79.6**	**0.03 13.0**	**0.15 -35.3**
1.80	**0.06 -136.7**	**12.00 -89.8**	**0.04 14.7**	**0.15 -37.8**
2.00	**0.07 -142.5**	**11.94 -100.4**	**0.03 14.3**	**0.15 -38.3**
2.20	**0.07 -143.9**	**11.87 -111.2**	**0.04 16.7**	**0.15 -37.8**
2.40	**0.08 -146.1**	**11.75 -121.9**	**0.04 16.2**	**0.15 -37.3**
2.60	**0.09 -148.4**	**11.56 -133.2**	**0.04 17.3**	**0.14 -36.9**
2.80	**0.09 -149.5**	**11.33 -144.5**	**0.04 15.6**	**0.14 -36.4**
3.00	**0.10 -152.7**	**10.95 -156.1**	**0.04 15.8**	**0.13 -35.9**
3.20	**0.10 -158.7**	**10.51 -167.5**	**0.04 15.6**	**0.13 -35.4**
3.40	**0.11 -163.2**	**9.97 -178.7**	**0.04 15.5**	**0.13 -34.9**
3.50	**0.11 -167.6**	**9.67 175.9**	**0.05 16.0**	**0.13 -34.6**
4.00	**0.12 165.9**	**8.25 150.6**	**0.05 12.0**	**0.13 -33.4**
4.50	**0.16 138.3**	**6.98 126.3**	**0.05 12.7**	**0.14 -37.1**
5.00	**0.19 122.8**	**5.71 105.0**	**0.06 9.5**	**0.12 -48.4**
5.50	**0.25 112.3**	**4.85 86.7**	**0.07 6.0**	**0.12 -63.0**
6.00	**0.30 99.3**	**4.14 70.4**	**0.07 1.0**	**0.11 -83.5**

Erläuterungen:
a) Zeilen, die mit einem Ausrufungszeichen (!) beginnen, stellen **Kommentare** dar und werden später ignoriert.

b) Der Gartenzaun (#) gibt an, in welcher Form die Parameter-Infos angeliefert werden. Die Angabe

ghz s m a r 50

bedeutet:

„**Die Zeilen beginnen immer mit der Frequenzangabe in GHz**“ (Hier sind es die Daten von 50 MHz bis 6 GHz).

Dann folgen die S-Parameter in der Reihenfolge

S11 S21 S12 S22

Jeder Parameter wird als Wertepaar aus **„m“ (= Magnitude, also Amplitude) und „a“ (= Angle, also Winkel)** angegeben.

Der Hinweis „**r 50“** besagt, dass der **Systemwiderstand Z einen Ohmscher Widerstand mit 50Ω darstellt.“**

Aufgabe: Kopieren Sie nur die obigeTabelle in ein einfaches Textfile und speichern Sie es in Ihrem Projekt-Ordner (…befindet sich im „.qucs“-Directory) unter dem Namen

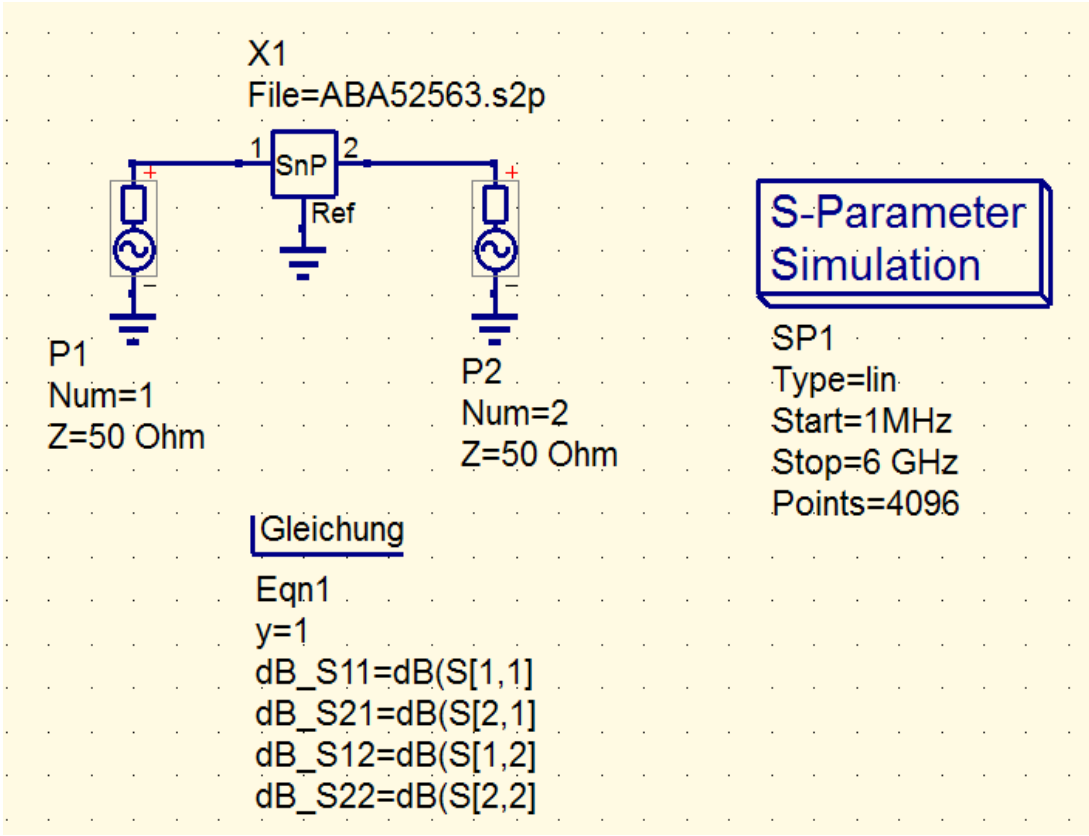
ABA52563.S2P

Bitte vor dem Abspeichern natürlich wieder an „ANSI“ und „alle Dateien“ denken….

==========

12.2. Simulation der S-Parameter bis 6 GHz

Dafür ist dieses Schaltbild erforderlich.



a) Die **Ports P1 und P2** finden wir als „Leistungsquellen“ über „Komponenten / Quellen“

b) Das **Zweitor X1 („Twoport“)** stammt als „S-Parameter-Datei“ aus „Komponenten / Geräte“

c) Die „**S-Parameter-Simulation**“ hat ihre Heimat in „Komponenten / Simulationen“

d) Und die Programmierung der **S-Parameter-Ausgabe in dB** erfordert natürlich wieder die zugehörigen Gleichungen

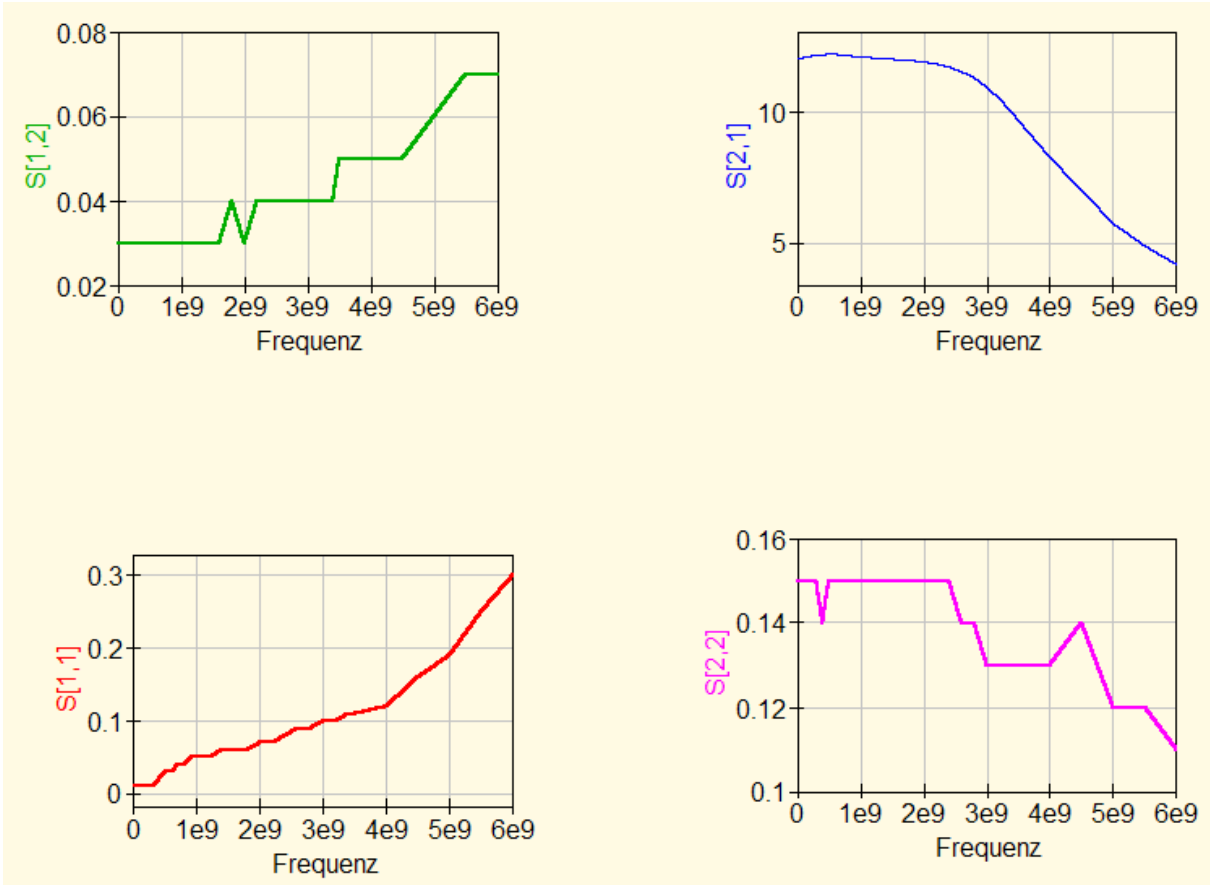
Beim Twoport X1 müssen wir natürlich „mit rechts“ auf das Symbol klicken, um an „Eigenschaften editieren“ heran zu kommen. Darin wird der korrekte Name unseres S-Parameter-Files angegeben.

Bei der Simulation sehen wir einen linearen Sweep von 1 MHz bis 6 GHz mit 4096 Punkten vor.
Dann wird die Simulation gestartet.

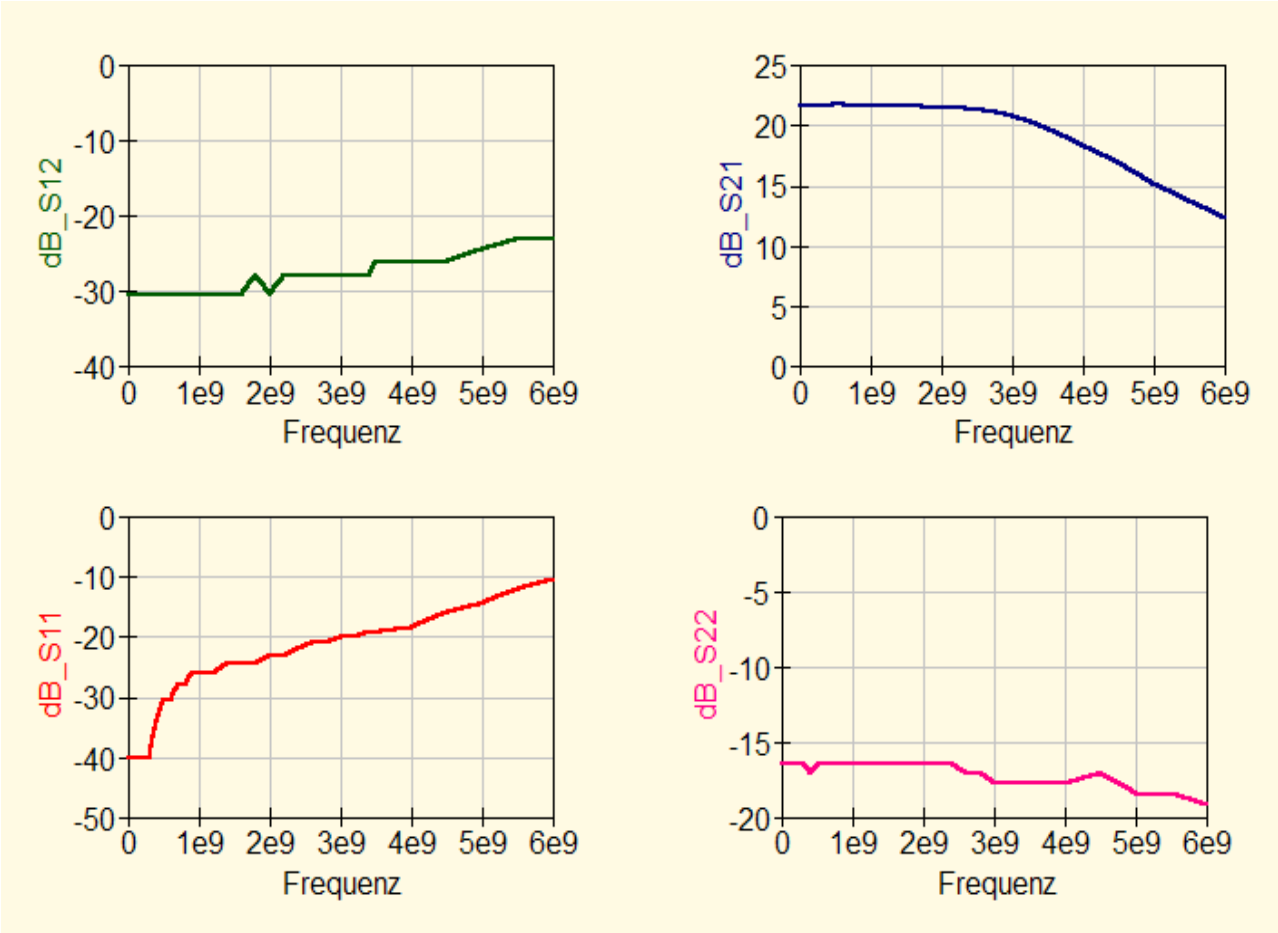
frequency	S[1,1]	S[1,2]	S[2,1]	S[2,2]
1e6	0.01 / 159°	0.03 / 0.888°	12.1 / -0.444°	0.15 / 0.246°
1.02e6	0.01 / 159°	0.03 / 0.888°	12.1 / -0.445°	0.15 / 0.245°
1.05e6	0.01 / 159°	0.03 / 0.887°	12.1 / -0.446°	0.15 / 0.244°
1.07e6	0.01 / 159°	0.03 / 0.887°	12.1 / -0.447°	0.15 / 0.242°
1.09e6	0.01 / 159°	0.03 / 0.887°	12.1 / -0.448°	0.15 / 0.241°
1.11e6	0.01 / 159°	0.03 / 0.887°	12.1 / -0.449°	0.15 / 0.24°
1.14e6	0.01 / 159°	0.03 / 0.886°	12.1 / -0.45°	0.15 / 0.239°
1.16e6	0.01 / 159°	0.03 / 0.886°	12.1 / -0.451°	0.15 / 0.237°
1.18e6	0.01 / 159°	0.03 / 0.886°	12.1 / -0.452°	0.15 / 0.236°
1.21e6	0.01 / 159°	0.03 / 0.886°	12.1 / -0.453°	0.15 / 0.235°
1.23e6	0.01 / 159°	0.03 / 0.885°	12.1 / -0.454°	0.15 / 0.234°

Als erste Übung können wir uns in bekannter Weise das neu berechnete (und interpolierte) S2P-File über die (schon bekannte) **Tabelle** ausgeben lassen.
Mit dem Reiter am linken Rand lassen sich dann alle 4096 Frequenzen „abfahren“.

Die nächste Aufgabe ist die lineare Darstellung der S-Parameter aus dem Touchstone-File in vier getrennten Diagrammen:



Meist wird die Darstellung dieser **S-Parameter in „dB**“ gefordert. Deshalb haben wir ja unser Schaltbild gleich um die „Gleichungs-Sammlung zur dB-Darstellung“ erweitert. Also gilt es, nochmals die Schaltung unter einem neuen Namen abzuspeichern, zu simulieren und sich um die Anpassung der vier Diagramme an die neue Aufgabe zu kümmern.



Aber nach etwas Mühe und Aufwand bei jedem einzelnen ausgegebenen Diagramm (Farben / Strichstärke / Frequenzachse / senkrechte Achse) sieht das doch recht hübsch aus und liefert alle Informationen:

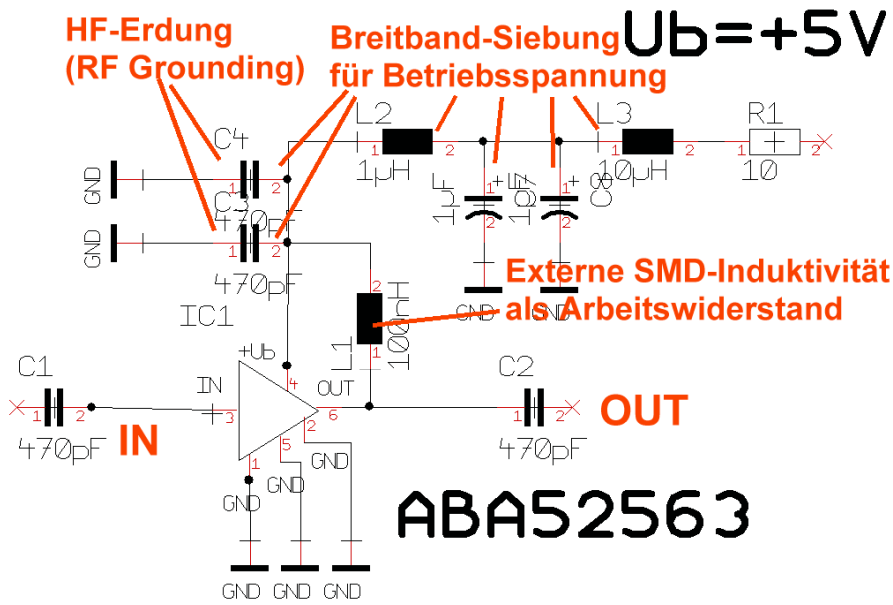
Man sieht, dass z. B. die Verstärkung (ausgedrückt durch S21) fast bis 3 GHz konstant ist.

Die Eingangs-Reflektion S11 bleibt in diesem Bereich unter -20 dB und das bedeutet -- speziell bei den tieferen Frequenzen! -- einen Eingangswiderstand von fast genau 50 Ω

Die Rückwirkungen (S12) sind kleiner als -20 dB und damit ist zu hoffen, dass die Schaltung nicht schwingen wird (…Hinweis: darüber entscheidet in diesem Fall auch die richtige Auslegung der Leiterplatte).

Nur die Ausgangs-Reflektion S22 könnte wirklich besser sein -- das sind keine 50Ω mehr!

12.3. Untersuchung der praktischen MMIC-Gesamtschaltung mit dem ABA52563



Die auf einer Leiterplatte aufgebaute praktische Schaltung sieht natürlich ganz anders aus und es wird einem als Anfänger fast schwindelig. Aber wir wollen gleich klären, wie wir mit der Simulation an die HF-Eigenschaften herankommen.

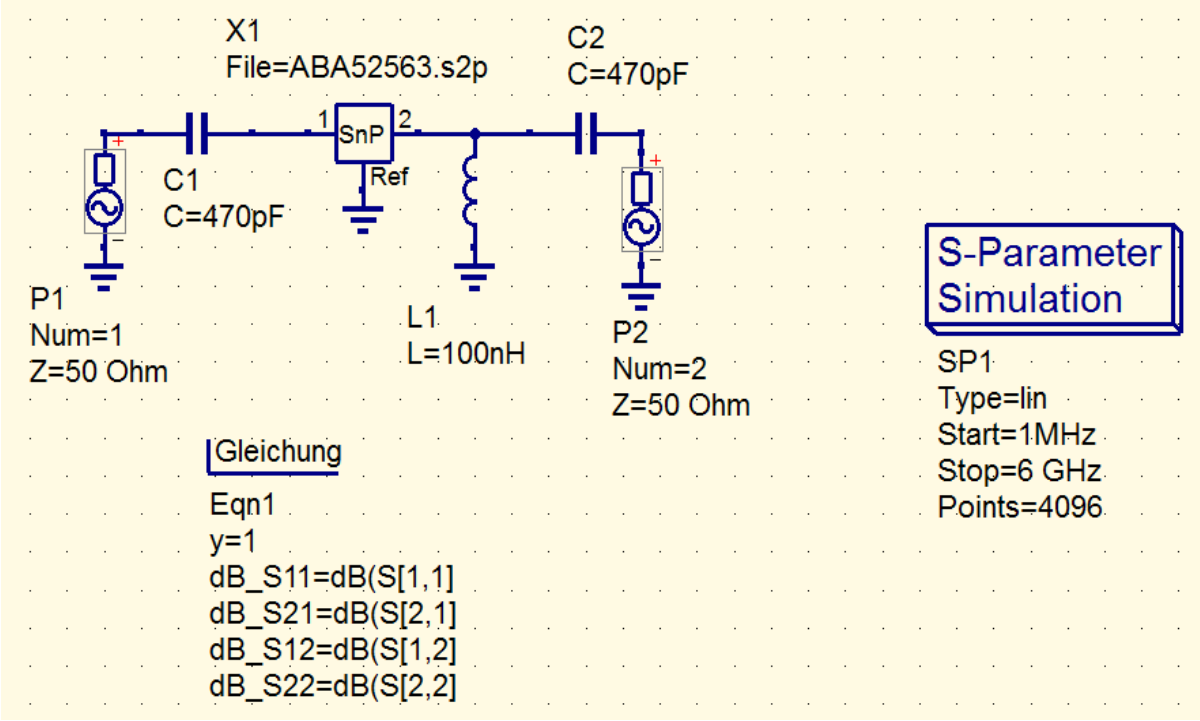
Da gelten einige einfache Spielregeln:

a) **Für Hochfrequenz stellen Kondensatoren mit Werten über 1nF praktisch einen HF-Kurzschluss dar.**

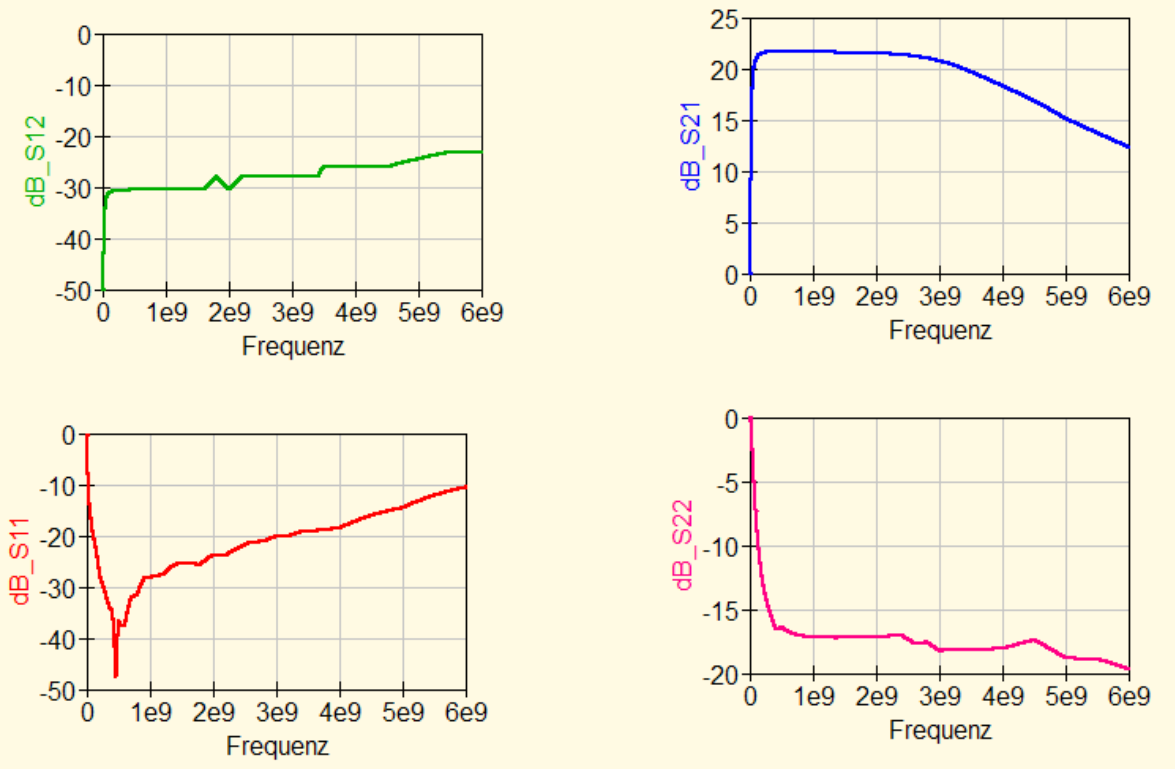
b) **Auch die Versorgungsanschlüsse liegen für HF auf Masse.**

c) **Spulen und Kondensatoren mit kleineren Werten müssen in der Simulation berücksichtigt werden.**

So sieht das dann aus:



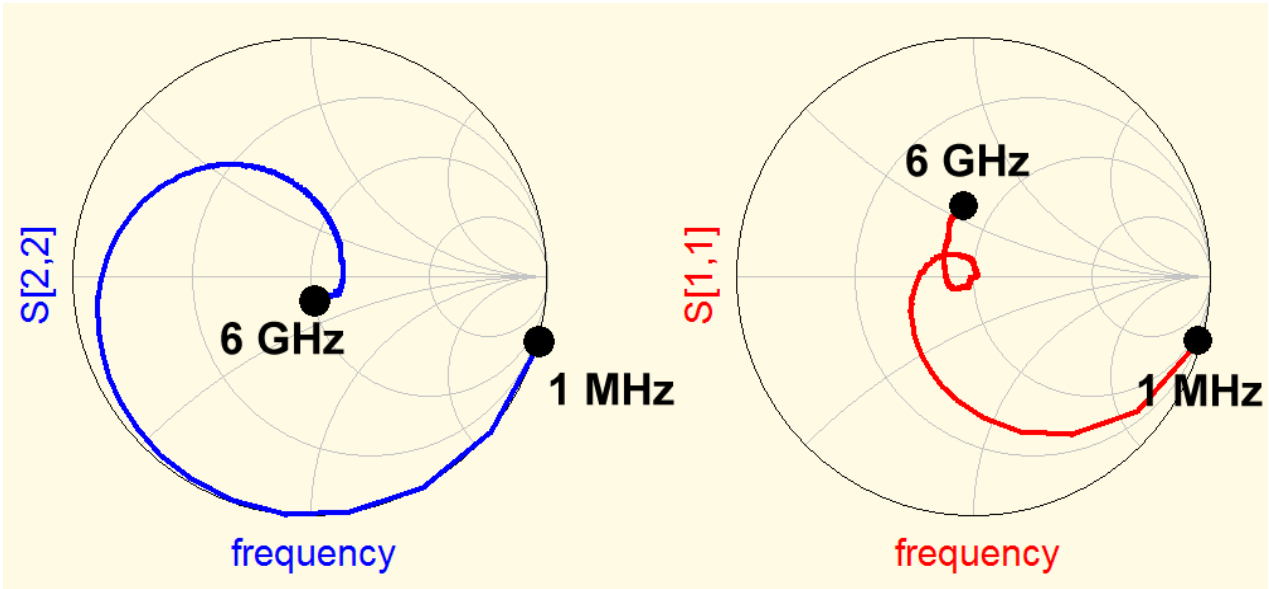
Dann holen wir uns gleich wieder die „S-Parameter in dB:



Da haben sich (wegen des zu tiefen Frequenzen hin abnehmenden Blindwiderstandes der 100nH – Induktivität als Arbeitswiderstand) vor allem die Verläufe von S11 und S22 geändert.

12.4. S11 und S22 der MMIC-Schaltung im Smithchart

Kein Problem: man simuliert, holt sich anschließend das Smithchart aus dem Diagramm-Vorrat und ruft S11 auf. Dann wird das Ganze für S22 wiederholt. Dargestellt wird mit Liniendicke „2“ und in zwei unterschiedlichen Farben:



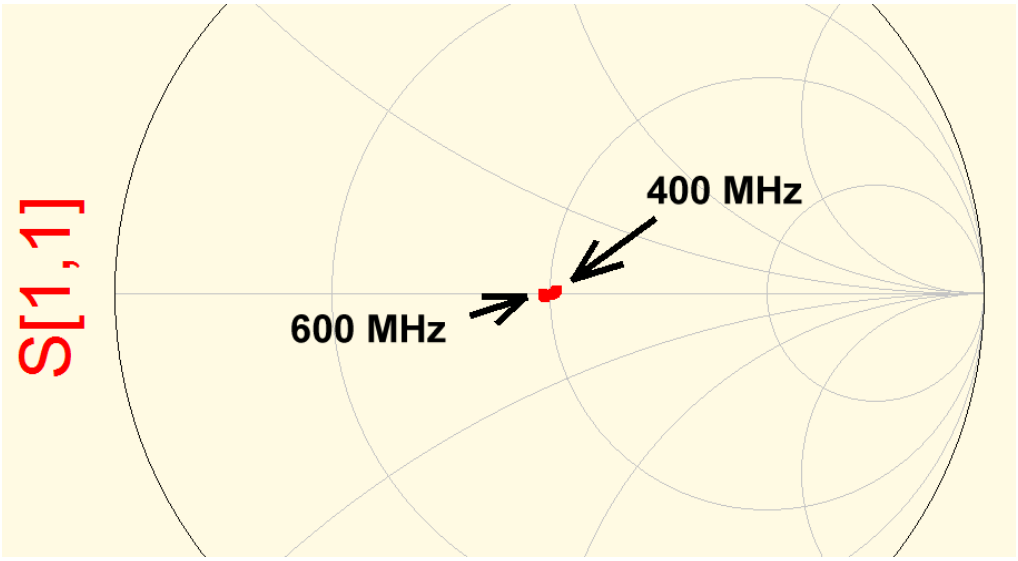
Man sieht:

a) Die Kurven drehen sich mit steigender Frequenz **IMMER im Uhrzeigersinn**.

b) Der Startpunkt liegt bei beiden Kurven in der **Nähe des „Unendlich“-Punktes**. Ist ja auch logisch, denn bei tiefen Frequenzen weist der Koppelkondensator am Eingang bzw. Ausgang noch einen hohen Blindwiderstand auf.

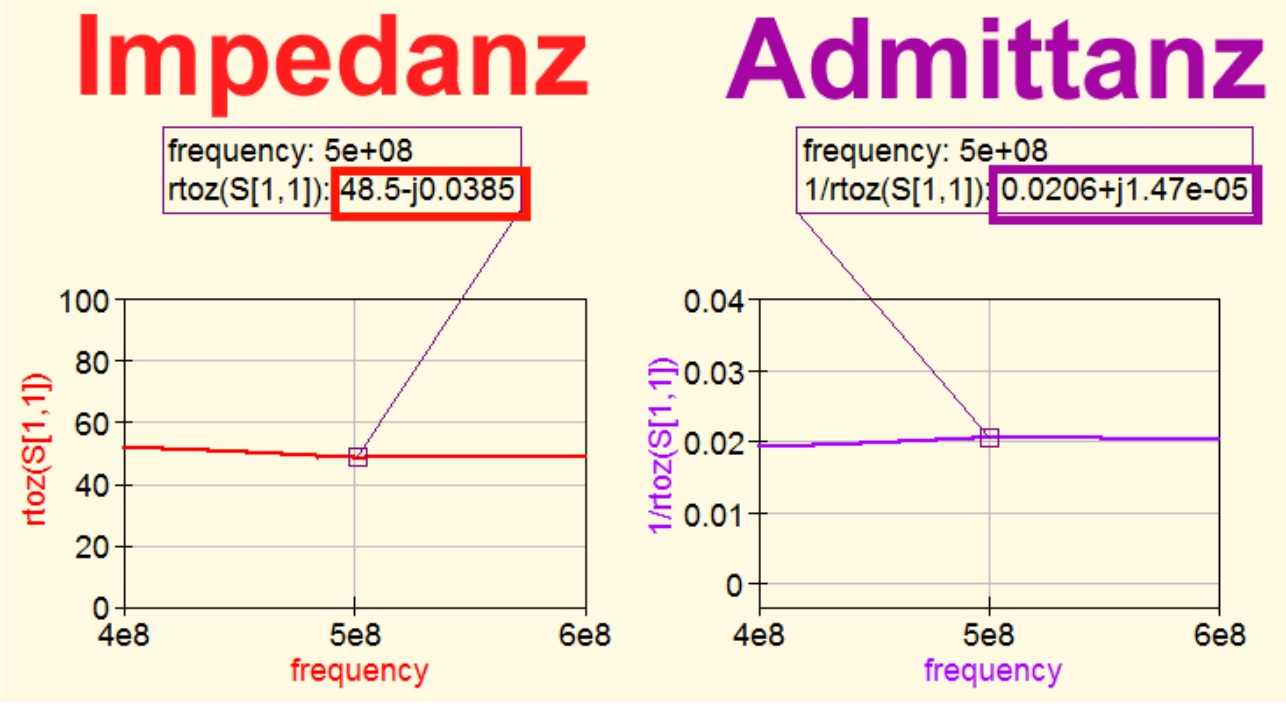
c) **Bei S22 wandert die Kurve mit steigender Frequenz erst mal in Richtung des „Kurzschluss“-Punktes, sobald der Blindwiderstand des Ausgangs-Trennkondensators C2 = 470 pF abnimmt**. Der Grund ist der lange recht niedrige Blindwiderstand der 100nH-Drossel, die als Arbeitswiderstand dient.

Bei S11 gibt es eine Frequenz, bei der die Kurve genau durch den Diagramm-Mittelpunkt (= perfekte Anpassung an 50Ω) geht. Wenn man in Kapitel 8.3.3. nachschaut, dann sieht man, dass das irgendwo bei 500 MHz der Fall ist. Möchte man den genauen Verlauf von S11 um diesen Punkt wissen, dann simuliert man einfach nochmals nur von 400 MHz bis 600 MHz – die Ergebnisdiagramme bleiben dabei in ihrer Einstellung unverändert.



Nun sieht man die Details.

Wer möchte, der kann sich nun (...wie wir das im Smithchart-Kapitel gelernt haben...) die Impedanz bzw. Admittanz zu **S11 für die Frequenz f = 500 MHz** ausgeben lassen. Dazu grenzt man den Frequenzbereich an der waagrechten Achse auf **400....600 MHz** ein und stellt bei den Marker-Properties auf "**real / imaginary**" um.



12.5. Praktische Ausführung des Verstärkers

12.5.1. Ein Blick auf die Leiterplatte

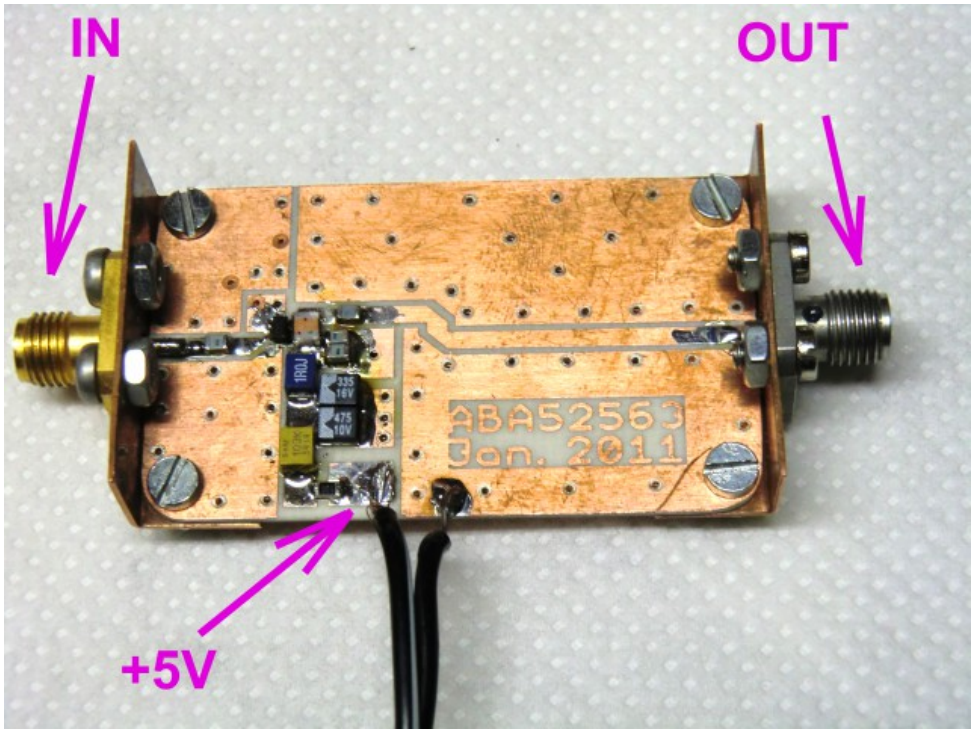
Das Konstruktionsprinzip für Mikrowellenplatinen ist recht einfach:
Auf der **Unterseite der zweiseitig kaschierten Platine** befindet sich stets eine **durchgehende Massefläche --** sie stelle die „unendlich gut leitende Erde“ dar.
Auf der **Oberseite** werden **HF-Signale auf Microstrip- Streifenleitungen oder in „Grounded Coplanar Waveguides“** geführt. Die sind für Koppelkondensatoren oder für MMICs durch einen kurzen Spalt (= „**Gap**“) unterbrochen.

Wichtig:
Die einzelnen Schaltungsteile werden stets durch völlig **voneinander getrennte Masse-Inseln** über **Durchkontaktierungen (Vias) mit der durchgehenden unteren Masse-Ebene verbunden und so geerdet**.

Speziell die Eingangs-Masse-Inseln dürfen nirgends mit Inseln verbunden sein, die zum Ausgang gehören. Dabei dürfen keine massiven Drähte verwendet werden, sondern es dürfen nur metallisierte Bohrungen (Vias“) zum Einsatz kommen.

Bei einzelnen Musterplatinen empfiehlt sich die Verwendung von **versilberten Kupfer-Hohlnieten** mit 0,8mm Durchmesser. **Mehr Durchkontaktierungen helfen mehr**, denn dadurch schalten sich die vielen kleinen Induktivitäten, die die Durchkontaktierungen darstellen, parallel und die **Gesamtinduktivität wird kleiner**.

Und wem die Sache mit dem „Drahtverbot“ anstelle von Hohlnieten merkwürdig vorkommt: das hängt mit dem **Skineffekt** zusammen. Bei hohen Frequenzen dringt der Strom nur noch wenige Mikrometer in einen elektrischen Leiter ein und bei einem eingelöteten massiven Draht ist deshalb der direkte Weg nach unten zur Masse versperrt! Der arme Strom muss dann auf der Inseloberfläche bis zum Rand der Insel und anschließend auf der Unterseite der Kupferschicht wieder bis zum Drahtstück zurück. Erst von dort geht es nach unter zur „Erde“ und dieser verlängerte Weg bringt nicht nur Verlustwiderstand, sondern meist auch zusätzliche Induktivität. **Das kann bei sehr hohen Frequenzen (z. B. bei einigen GHz) plötzlich sehr leicht einige dB an zusätzlicher Dämpfung ergeben oder zur Schwingneigung führen!**
Fertig bestückt und zur S Parameter-Messung vorbereitet sieht die Platine dann so aus:



Links und rechts sind Kupferwinkel als Träger für die SMA-Buchse beim Eingang und Ausgang angeschraubt. Die Mittelleiter dieser Buchsen liegen flach auf der Platine auf und ihr Ende ist unter 45 Grad angefeilt. So ergibt sich ein möglichst reflektionsarmer Übergang beim Auflöten auf die Streifenleitung.

Diese Methode empfiehlt sich besonders für den ersten schnellen Test und funktioniert problemlos bis zu etlichen GHz.

Erst nach der Fertigstellung der kompletten Entwicklung und Beseitigung der letzten Fehler lässt man die Platine im gefrästen Alugehäuse verschwinden, schraubt die Platine fest und schließt die SMA-Buchsen für die HF sowie die SMB-Buchse für die Stromversorgung an. So sieht das ganze Gebilde dann aus, bevor der Deckel drauf kommt und gemessen wird:

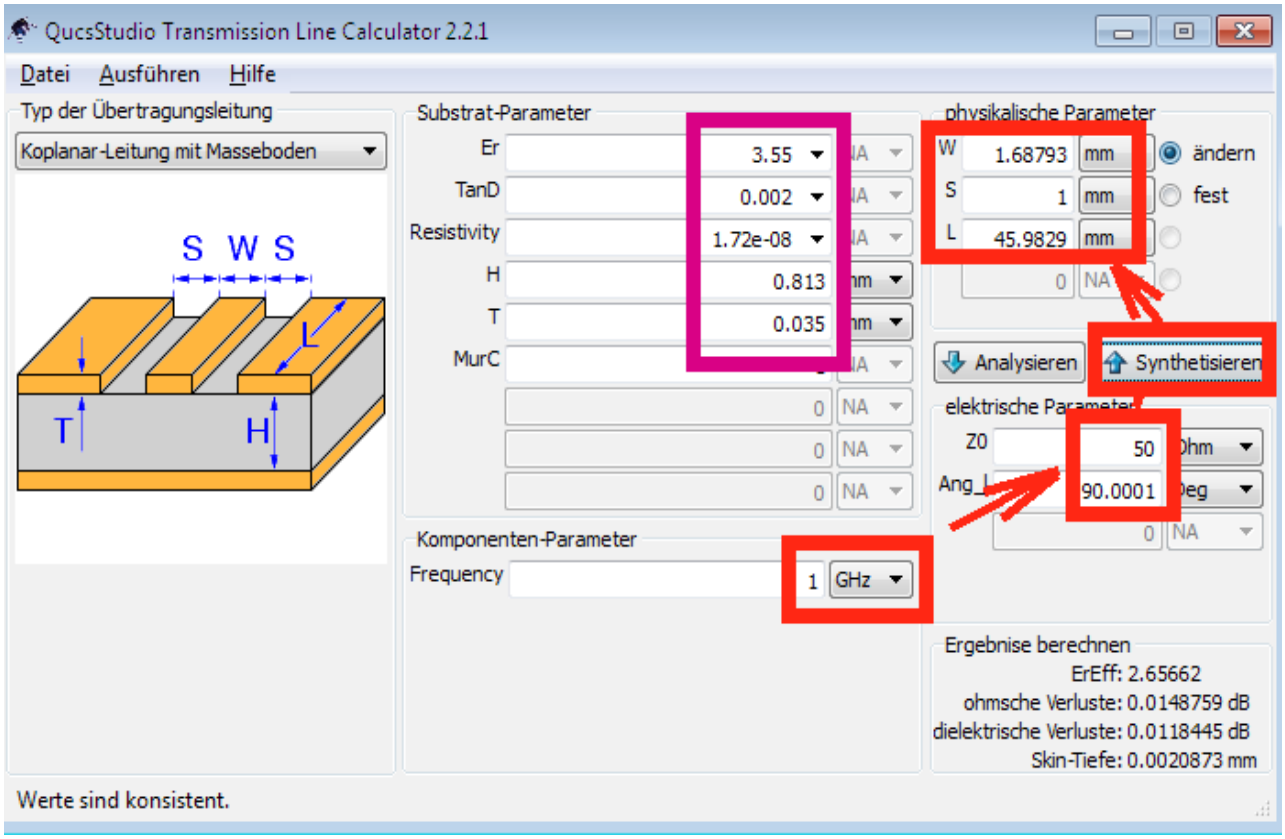
Wer sich über die „nach links auf der Platine verschobene Schaltung“ wundert:

Die Eingangs-Streifenleitung muss so kurz wie möglich gehalten werden, denn ihre Dämpfung (in dB) verschlechtert genau um diesen Betrag das Rauschverhalten (präzise: die Noise Figure F).

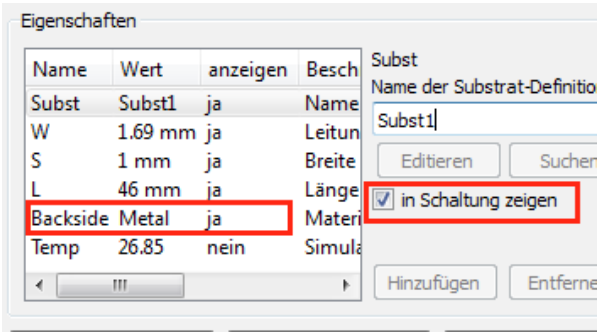
Außerdem gibt es auf diese Weise noch Platz, um dieselbe Stufe nochmals auf der Platine unterzubringen und so eine Gesamtverstärkung von über 40 dB zu erzielen (…die Einzelstufe besitzt ein S21 von ca. 21 dB).

12.5.2. Entwurf des „Grounded Coplanar Waveguides“ für die Platine

Wir benötigen für unsere Simulation noch die Mikrostreifen-Leitung mit Z = 50 Ω, die für die Koppelkondensatoren und den MMIC durch Lücken („Gaps) unterbrochen wird. In der Praxis zieht man statt der einfachen Leitung oft den „Grounded Coplanar Waveguide“ (…in unserem Calculator heißt das „Koplanar-Leiter mit Masseboden“) vor. Bei ihm ist die Streifenleitung links und rechts durch Masse-Ebenen eingerahmt. Diese Abschirmung vermindert die Empfindlichkeit gegen Störungen -- sei es Einstrahlung oder Selbst-Abstrahlung.
Also rufen wir wieder über „Werkzeuge“ und „Leitungsberechnung“ unseren Transmission Line Calculator auf und schalten auf „Koplanar-Leitung mit Masseboden“. Und dann geht es erst mal mit den Substratdaten im violetten Rahmen los, die korrekt eingetragen werden müssen.
Dann wird die Frequenz auf „1 GHz“ eingestellt, gefolgt vom Wellenwiderstand ***Z = 50Ω*** und einer **elektrischen Länge von 90 Grad**. Bevor wir jedoch auf „Synthetisieren“ klicken, entscheiden wir uns für eine **Spaltbreite („Separation“) von S = 1 mm** zwischen dem Mittelleiter und den eingrenzenden Masseflächen. Der erlösende Klick liefert dann eine **Mittelleiter-Breite von 1,69 mm**.



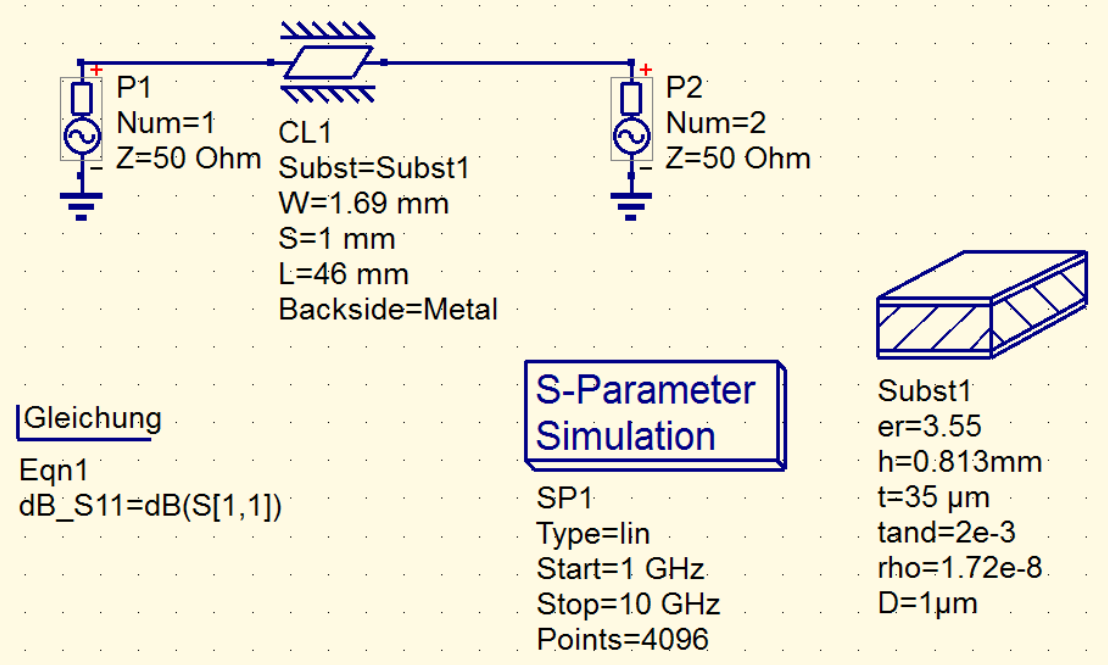
Diese Daten sollte man wieder über die schon bekannte Simulation testen:
Man schließt am Eingang einen Mikrowellen-Port an und schließt sie am Ende mit 50Ω ab. Gesweept wird von 1 GHz bis 6 GHz.



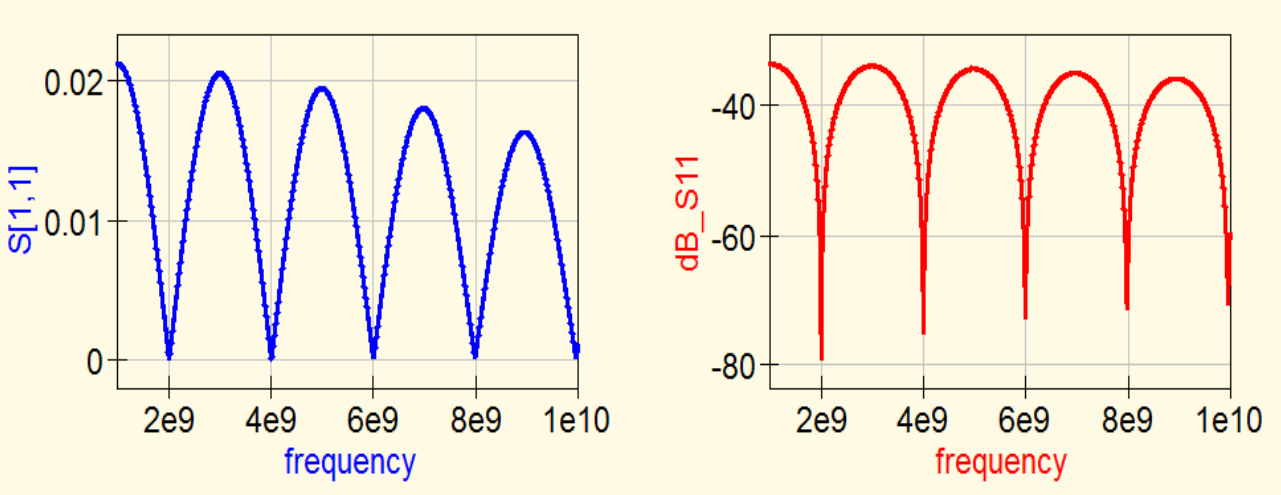
Den **Coplanar Grounded Waveguide finden wir als „Koplanar-Leitung“ unter „Komponenten / Übertragungsleitungen“.** Allerdings muss man nach dem Editieren der Eigenschaften sehr aufpassen und darf die **Umstellung der „Backside“ von Air auf Metall** nicht vergessen. Gibt sonst große Fehler…

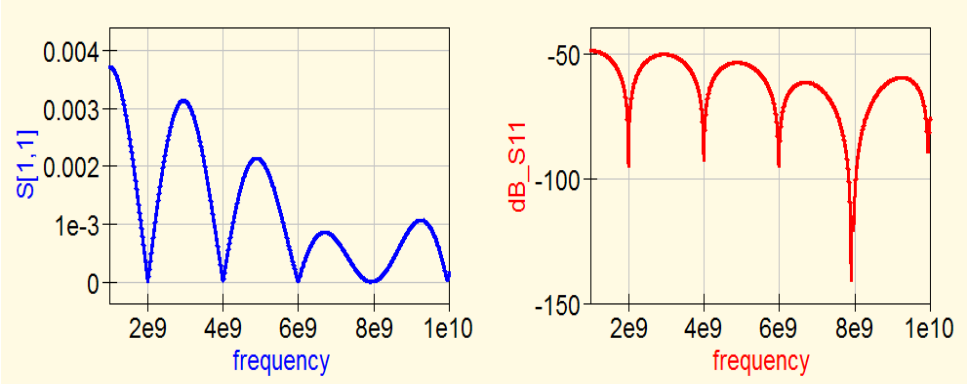
Außerdem empfiehlt es sich, alle Eigenschaften im Schaltbild sichtbar zu machen.

Damit brauchen wir folgende Schaltung:



Und die liefert folgende Simulationsergebnisse:





Das ist noch etwas unbefriedigend, denn S11 steigt bis auf -35 dB an.

Da kann man noch durch leichte Variation der Leiterbreite (Achtung: um Hundertstel Millimeter..!) nach einer Verbesserung suchen. Hier wäre das Ergebnis für **W = 1,74 mm** statt 1,69 mm: **jetzt sinkt S11 unter -50dB**.

12.6. Die Gesamtschaltung

12.6.1. Simulation der S-Parameter

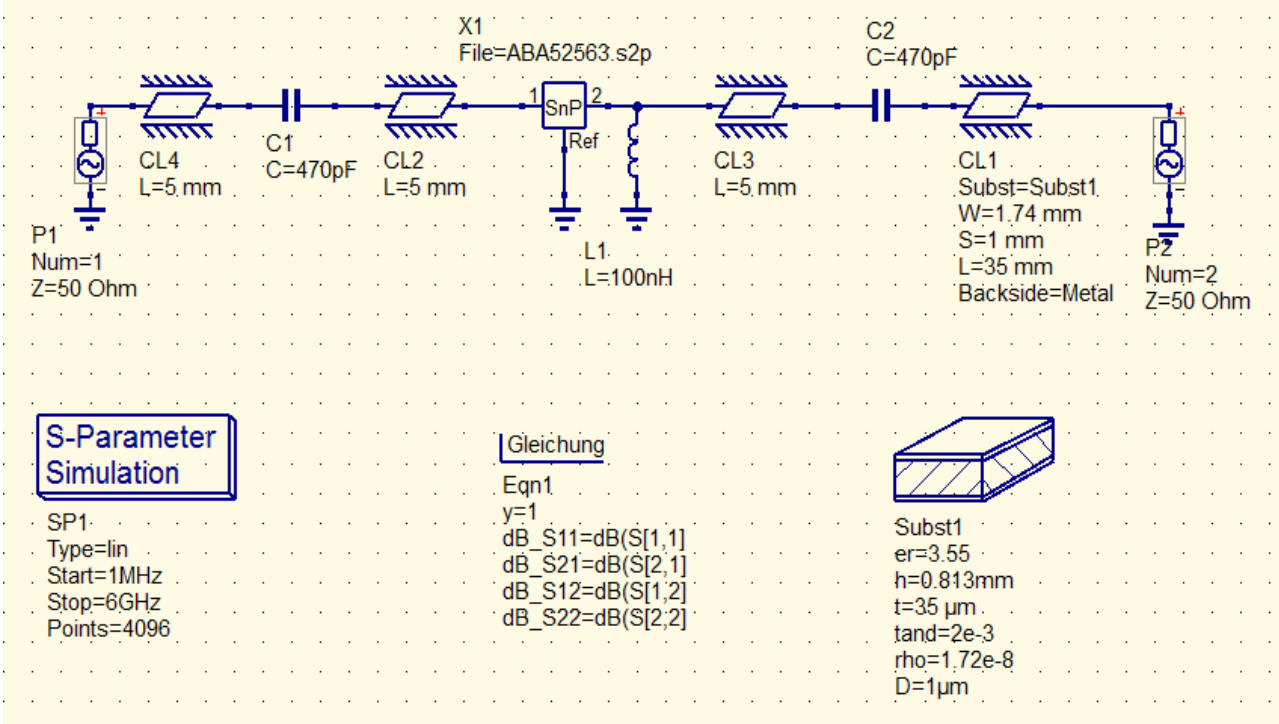
Zuerst holen wir uns die Simulationsschaltung des Verstärkers aus Kapitel 10.3 auf den Schirm und speichern sie unter einem neuen Namen. Dann wird geändert, denn entsprechend dem Platinen-Layout müssen wir nun noch vier „Grounded Coplanar Waveguide“ – Leitungsstücke in die Schaltung einfügen. Ihre Längen betragen:

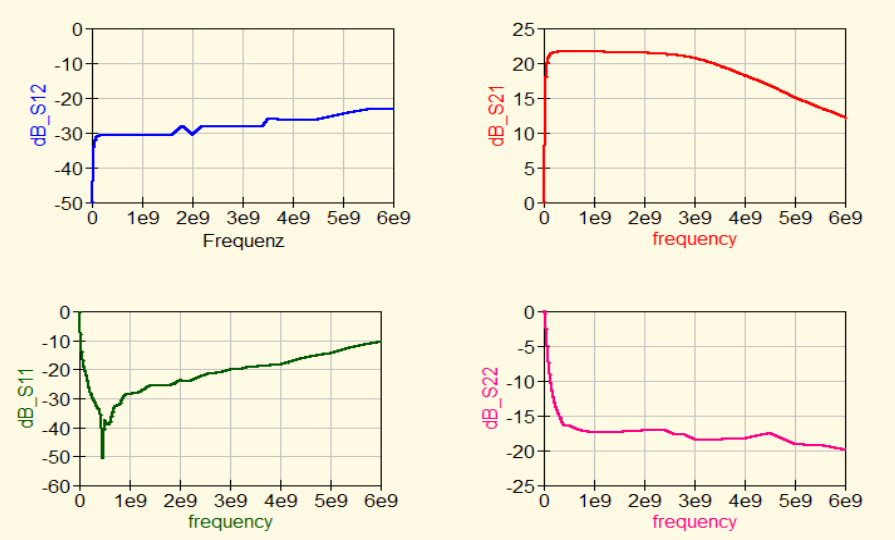
a) 5mm von der Eingangsbuchse bis zum 470 pF – Kondensator

b) 5mm vom Kondensator bis zum MMIC-Eingang

c) 5mm vom MMIC-Ausgang bis zum Ausgangs-Trennkondensator mit 470 pF

d) 35mm vom Ausgangs-Trennkondensator bis zur Ausgangsbuchse





Die Substrat- und Leitungs-Daten übernehmen wir aus den vorigen Kapitel. Um Platz zu sparen, sollte man nur bei einem Leitungsstück alle Eigenschaften im Schaltbild sichtbar machen. Bei allen anderen reicht die Information über ihre Länge. Dann wird von 1MHz bis 6 GHz simuliert.

Wer zurückblättert, der wird keinen merkbaren Unterschied zur Schaltung ohne die Leitungsstücke finden.

12.6.2. Ermittlung des Rollet Stabilitätsfaktors „k“

Man möchte schließlich wissen, ob sich die Schaltung anständig benehmen und nicht schwingen wird. Dazu gibt es diesen Stabilitätsfaktor, der aus den S-Parametern über eine recht komplizierte Beziehung ermittelt wird -- aber das nimmt uns der PC ab.
Die Auswertung ist sehr einfach:

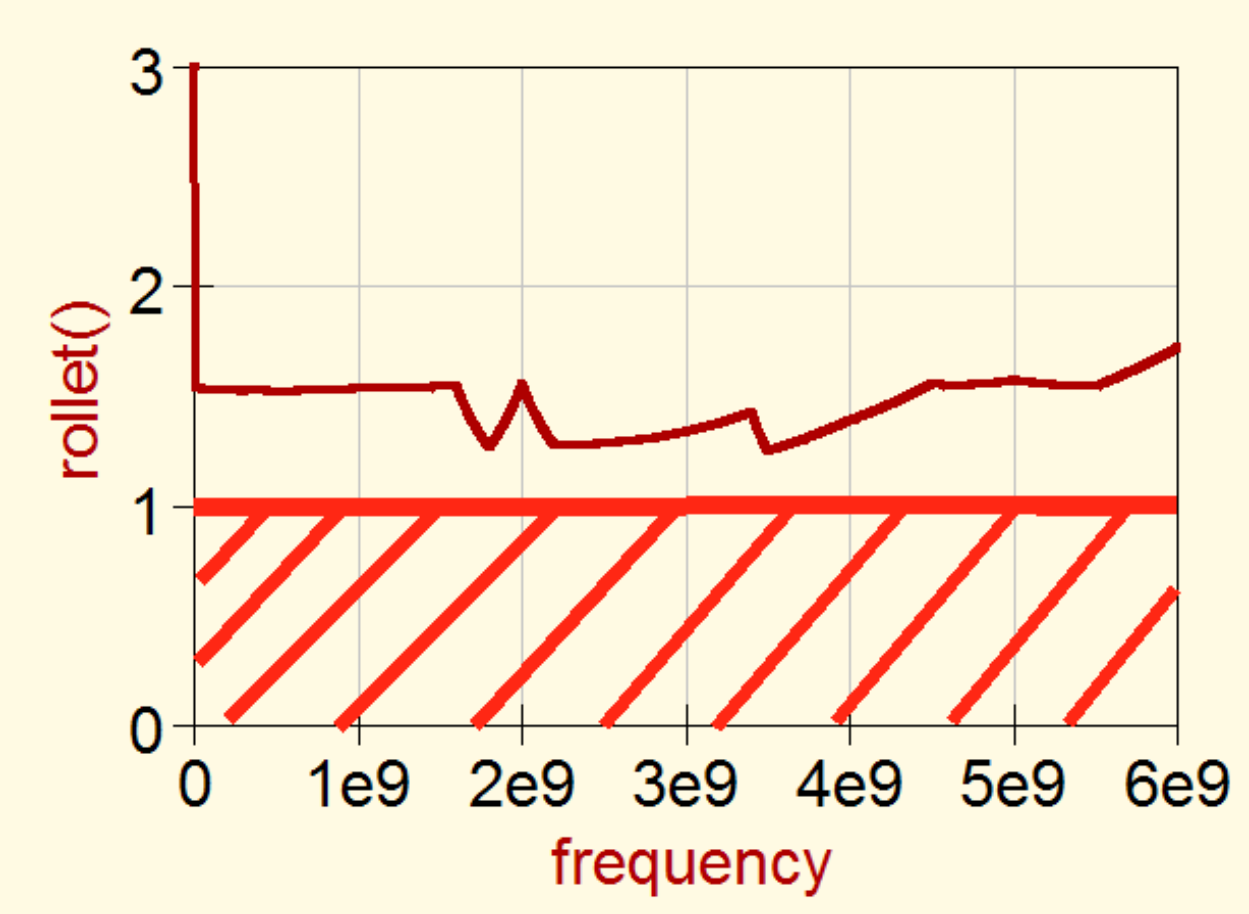
Der Stabilitätsfaktor „k“ muss bei allen Frequenzen größer als 1 sein, damit die Schaltung stabil bleibt und nirgends schwingt.

Nach der S-Parameter-Simulation holt man sich ein kartesisches Diagramm, editiert seine Eigenschaften und trägt unter „Eigenschaften des Graphen“ die Anweisung

„rollet()“

ein.

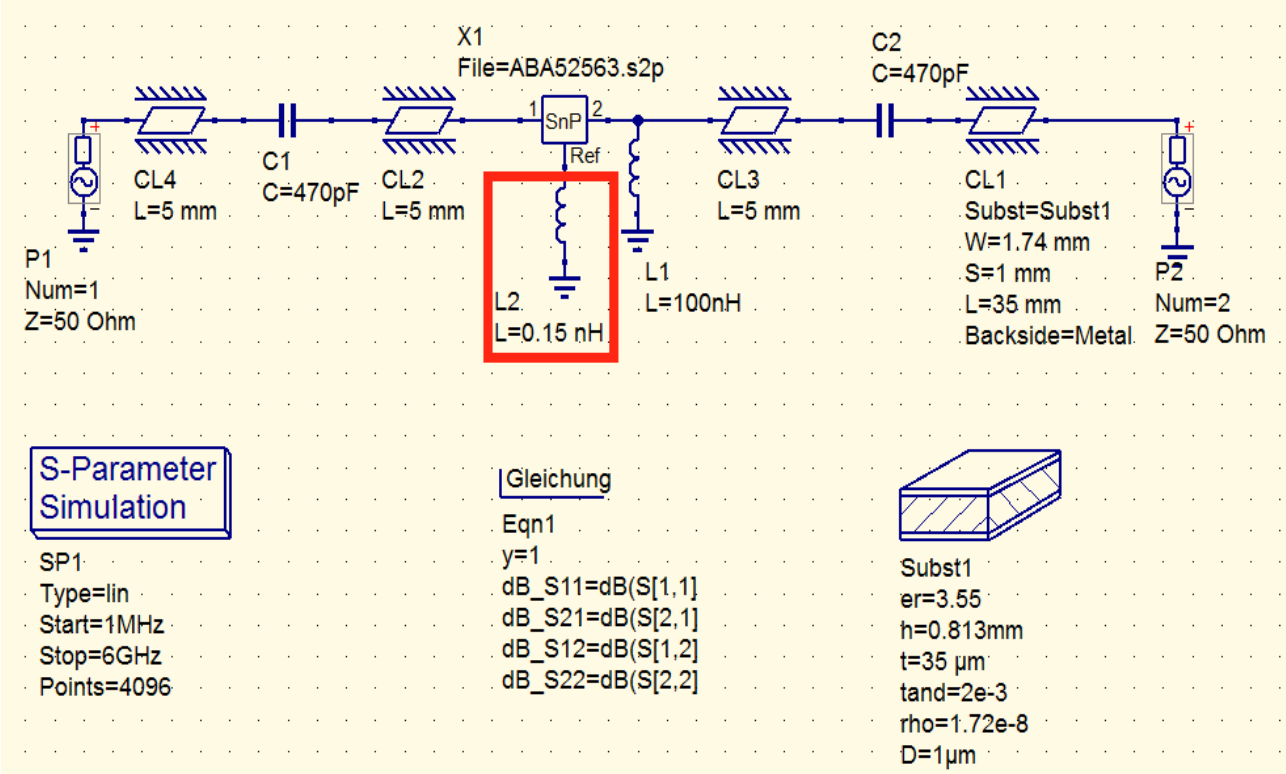
Mit „Anwenden“ erhält man dieses Bild und man weiß nun, dass alles in Ordnung ist.
Der Stabilitätsfaktor ist nämlich von Null bis 6 GHz größer als 1.

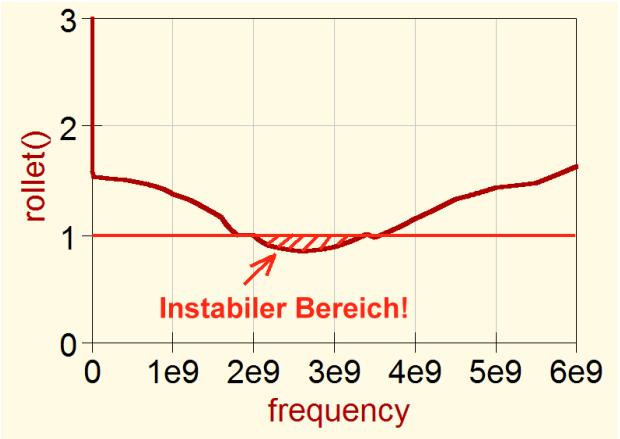


Praktisches Beispiel:
Bei unserem MMIC ist die **Verbindung vom Referenzpunkt nach Masse äußerst kritisch:**

Wenn diese Verbindung nicht sehr induktivitätsarm (= mit vielen Vias“) ausgeführt wird, sinkt „k“ unter 1 und die Schaltung schwingt.

Sehen wir uns das mal für **L = 0,15 nH** an (…das entspricht etwa einer Drahtlänge von knapp 2 mm):





So sieht jetzt „k“ aus…und die tatsächlich aufgebaute Schaltung schwang in der ersten Version dann auch bei .etwa 2,2 GHz. Da half nur eine Erhöhung der Vias auf jeder Seite des MMICs.

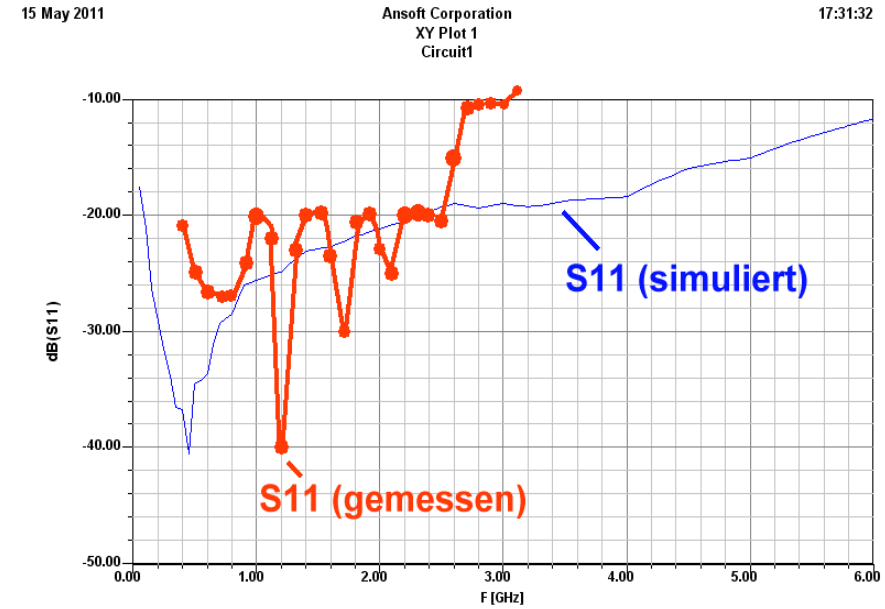
Hinweis:
Man weiß jetzt aber nicht, ob diese angezeigte Instabilität den Eingang oder den Ausgang betrifft (= welche Werte und Blindanteile beim Quell- oder Lastwiderstand gefährlich sind).

Dazu braucht man die Darstellung des Input- und des Output-Stabilitätskreises im Smithchart -- aber das ist ein eigenes Simulationsthema und das verschieben wir auf später….

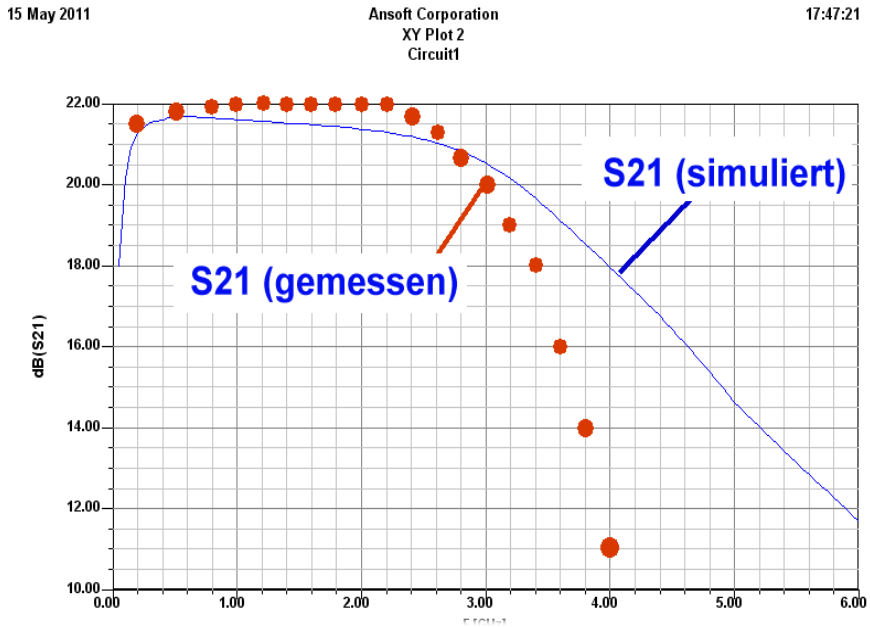
12.6.3. Die Stunde der Wahrheit: Messergebnisse des Vektoriellen Network-Analyzers

Dieser Teil stammt aus meiner **Veröffentlichung in der Zeitschrift „UKW-Berichte**“, in der ich die eben vorgestellte Entwicklungsprozedur unter Verwendung der Software „Ansoft Designer SV“ genau beschrieben hatte**. (Diese Software sowie alle meine Artikel aus den UKW-Berichten finden sich in meiner Homepage www.gunthard-kraus.de und können von dort heruntergeladen werden).**

Sehen wir uns mal an, ob unsere Simulation etwas getaugt hat und beginnen wir mit **S11:**



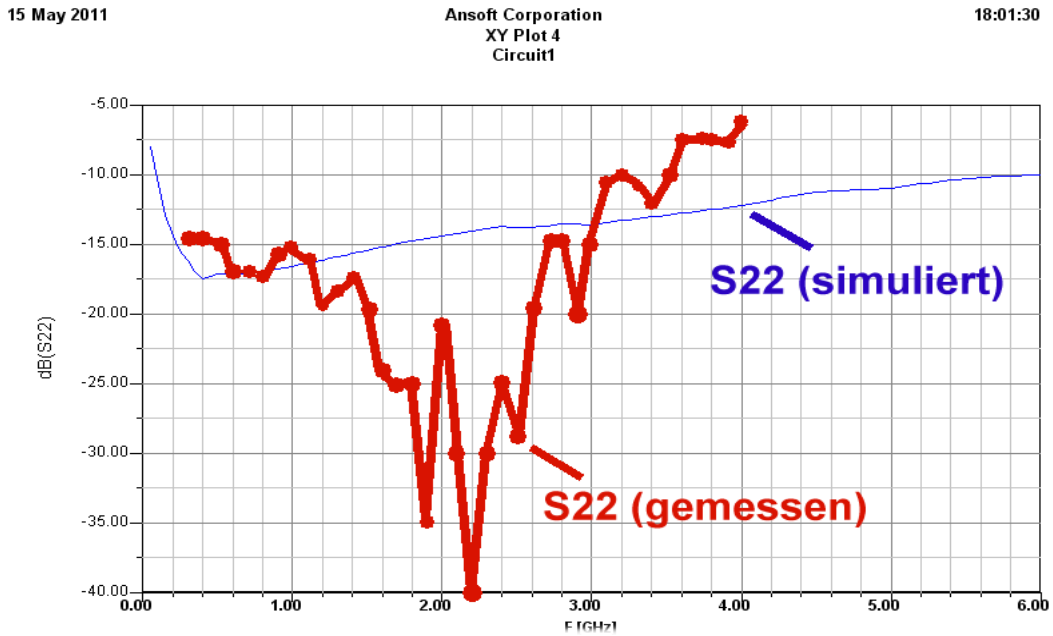
Bis 2,5 GHz ist S11 kleiner als -20 dB, aber dann wird es deutlich schlechter. Jedoch sind die Vorgaben (S11 kleiner als -10 dB) eingehalten. Bei den wilden Zickzack-Bewegungen der Kurve werden wohl die Koppel- und Abblock-Kondensatoren mit ihren Eigenresonanzen beteiligt sein…



Auch bei **S21** (also der Verstärkung) sieht man, dass die **100 nH-Induktivität oberhalb ihrer Eigenresonanz bei 2,2 GHz die Verstärkung brutal vermindert**, denn dann ist sie nur noch ein Kondensator. Sie müsste deshalb kleiner gewählt werden, aber dann steigt wiederum die untere Grenzfrequenz an.

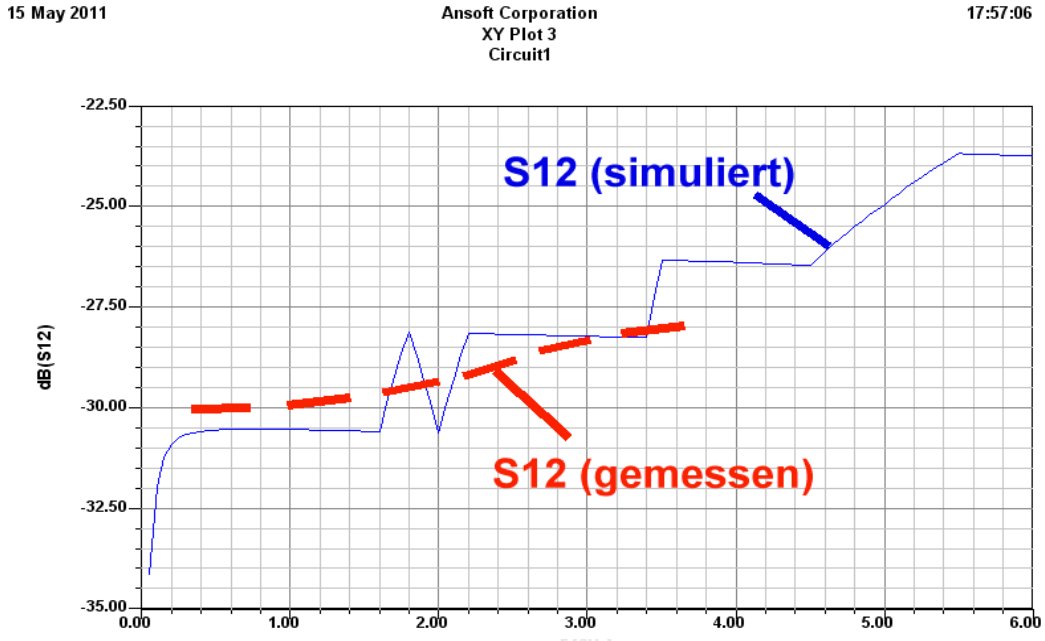
In einer getrennten Messung wurde die untere Grenzfrequenz (-3 dB-Frequenz) des Verstärkers zu 44 MHz für diesen Induktivitätswert (100 nH) ermittelt.

Dann interessiert natürlich **die Ausgangs-Reflektion**, ausgedrückt durch **S22**:



Sie ist bis 3 GHz sogar besser als das Simulationsergebnis. Und bei 2,2 GHz dürfte die Eigen-Parallelresonanz der 100 nH-Induktivität liegen, denn da haben wir den niedrigsten Wert der Ausgangs-Reflektion. Das passt auch zum leichten Maximum bei S21 für diese Frequenz (siehe voriges Diagramm).

Und auch über die Rückwirkungen (= ausgedrückt durch S12) kann man nicht klagen:



Grundsätzlich kann nun die Platine in die Fertigung gegeben werden, denn durch den Einbau in das gefräste Aluminiumgehäuse und das Aufschrauben des Abschirmdeckels ändert sich nicht mehr viel.

13. Entwicklung eines Low Noise Amplifiers für 1 GHz...1,7 GHz mit einer maximalen Rauschzahl von 0,4 dB

13.1. Überblick

Die Entwicklung hochwertiger integrierter Mikrowellen-Verstärkerbausteine (MMIC's) ist dauernd im Fluss und führt zu immer besseren Daten. Deshalb ist es reizvoll, damit einen LNA zu bauen, denn die Vorteile hören sich so an:

Eingang und Ausgang bereits für den 50 Ω-Betrieb angepasst Die Rauschzahlen sind nun auch bei tieferen Frequenzen (z. B. ab 500 MHz) und beim Einsatz im 50 Ω-System unter 0,5 dB gesunken Ein Einzelbaustein erzielt jetzt ca. 20 dB an Verstärkung bis 2 GHz Wenig äußere Beschaltung nötig

Da auch hier der Satz „Keine Rose ohne Dornen" gilt, sollte man um die Probleme wissen. Denn zur Erzielung dieser Werte gilt es, eine ganze Reihe von Aufgaben zu lösen:

Diese SMD-Bausteine sind nun winzig klein und ohne Anschlussfüße (…die hier verwendete Ausführung hat Abmessungen von 2 mm x 2mm, aber 8 Anschlusspads plus Masse) Jetzt ist der gemeinsame Masseanschluss des Chips in der Mitte der Gehäuse-Unterseite angeordnet und oft nicht mehr bis zum Gehäuserand geführt Der Layout-Entwurf erfordert sehr hohe Aufmerksamkeit (= Leiterbahnen und Anschlusspads am IC typisch 0,25 mm, maximal 0,5 mm Breite) Die zusätzlich erforderlichen SMD-Bauteile der äußeren Beschaltung dürfen die Größe 0603 (= 1,25 mm x 0,75 mm) möglichst nicht mehr überschreiten Die Grenzfrequenzen der Bausteine liegen in der Zwischenzeit so hoch, dass die Stabilitätskontrolle selbst beim Betrieb unter 1 GHz unbedingt bis 10 GHz vorgenommen und entsprechende Vorsorge getroffen werden muss Die Arbeitspunkte müssen wegen der z. T. hohen Ströme (= oft über 50 mA pro Stufe) sehr sorgfältig stabilisiert und die Betriebsspannungen noch sorgfältiger und extrem breitbandig gesiebt werden Die Dicke der Leiterplatten wurde wegen dieses erweiterten Frequenzbereichs bei allen bekannten Applikationen auf 0,25 mm reduziert, um die Entstehung unerwünschter Modi bei den Signalen auf den Streifenleitungen zu verhindern. Damit können die vielen erforderlichen Durchkontaktierungen nicht mehr mit versilberten Hohlnieten vorgenommen werden.

Dieses Projekt ist für den Einsatz im 23 cm-Band (= ca. 1300 MHz) und zum Meteosat-Empfang bei 1691 MHz vorgesehen. Deshalb wurden folgende Daten angestrebt und das Internet nach neuesten MMIC-Entwicklungen durchforstet:

Rauschzahl: maximal 0,4 dB

Verstärkung (S21) ca. 20 dB

Absolute Stabilität (k größer als 1 bis hinauf zu 10 GHz)

Die Wahl fiel schließlich auf den Baustein „MGA-635P8" der Firma Agilent-Avago, dessen Beschaffung sich als absolut unproblematisch erwies (…die Bauteile waren eine Woche nach der Online-Bestellung bei mouser.com im Haus) und auch der Preis (80 Euro für 25 Stück) erträglich.

13.2. Die Ausgangs-Schaltung

Hierzu brauchen wir das Datenblatt des MMIC's, die herunter geladenen S-Parameter und zwei vom Hersteller bereitgestellte Application Notes für 2500 MHz und 3500 MHz [1] [2] [3]. In allen Fällen werden dieselbe Verstärker-Grundschaltung und dasselbe Platinen-Layout eingesetzt. Lediglich die Bauteilwerte sind den unterschiedlichen Frequenzbereichen angepasst.

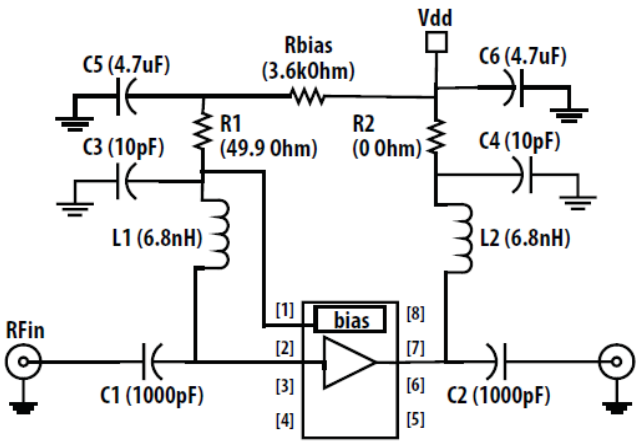
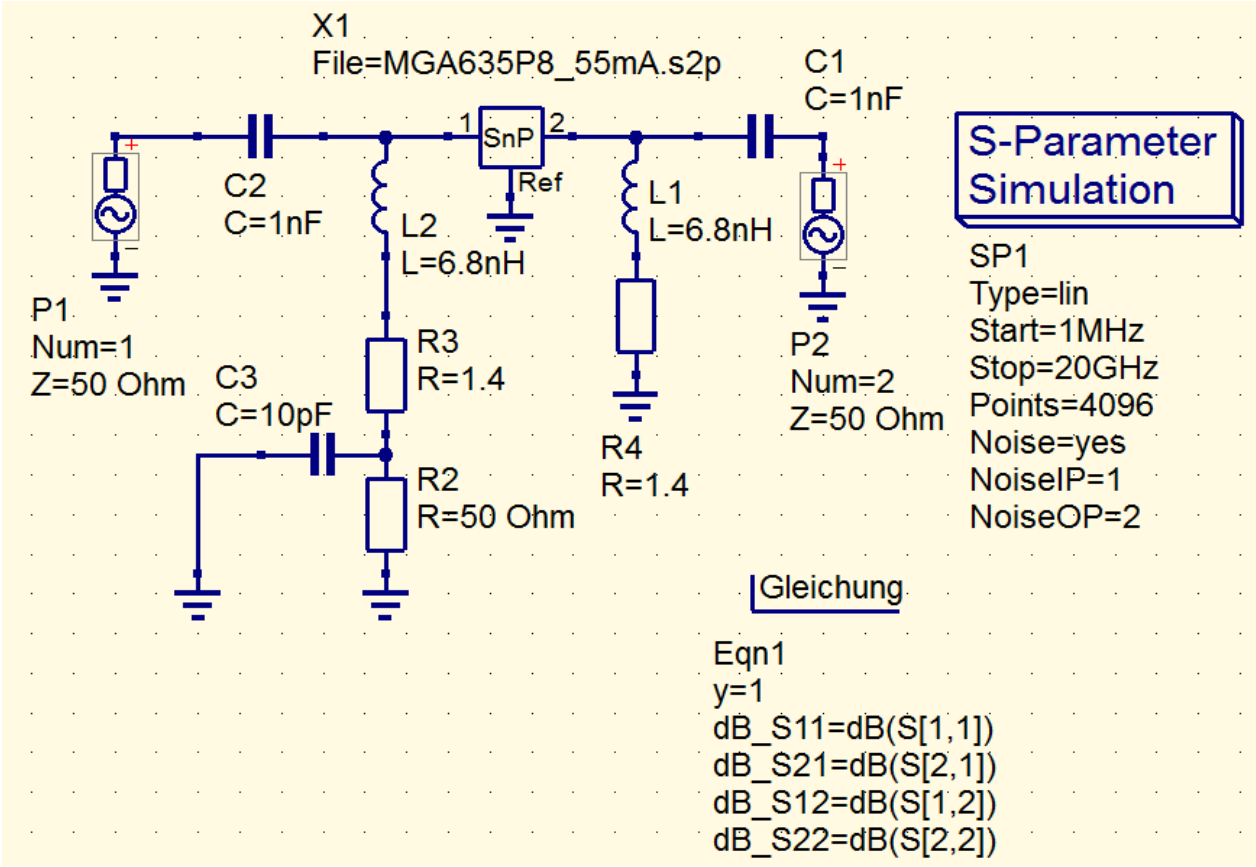


Figure 6. Demo Board Schematic Diagram

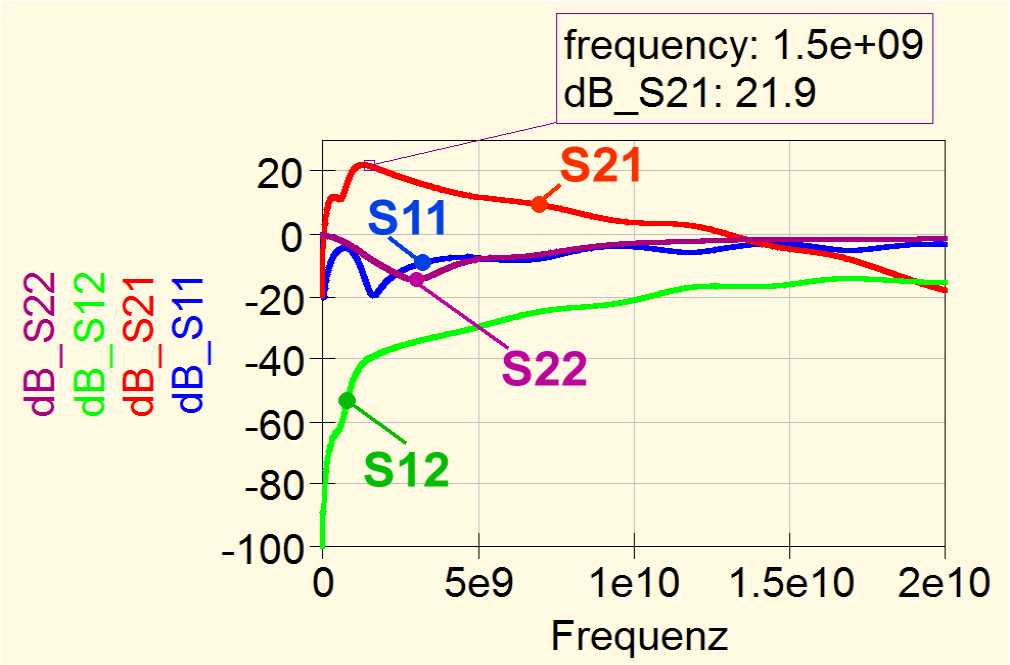
Doch nun zur Schaltung:
Sie arbeitet mit einer Versorgungsspannung von Vdd =+5 V und im Inneren des MMIC-Gehäuses. befindet sich ein GaAs-pHEMT-Kaskodenverstärker. An Pin 1 wird über einen Vorwiderstand (Rbias = 3,6 kΩ) der gewünschte Arbeitspunkt von 55 mA eingestellt.

Ein großes Problem der HEMT-Bauteile ist die Stabilität bei tiefen Frequenzen -- also ihre Schwingneigung. Da hilft man sich mit einem einfachen Trick: man sorgt dafür, dass mit abnehmender Frequenz immer mehr der Zusatzwiderstand R1 mit etwa 50Ω am Eingangspin 2 wirksam ist und so ein Schwingen verhindert wird.
(Erläuterung:
mit sinkender Frequenz sinkt auch der Blindwiderstand von L1 (6,8 nH), aber es steigt der Blindwiderstand von C3 (10 pF). Also ist irgendwann allein R1 am Eingangspin aktiv.
Zusätzlich reduziert die Beschaltung mit der sehr kleinen Induktivität L2 = 6.8 nH als Arbeitswiderstand der zweiten Stufe die Verstärkung bei sinkenden Frequenzen bis auf Null.
Der ebenfalls in der Schaltung eingetragene Widerstand R2 mit Null Ohm wurde ignoriert.
Vor der Weiterentwicklung wurde die obige Schaltung mit unserem"qucsstudio" untersucht. So sieht das dazu eingegebene Simulationsschaltbild aus:



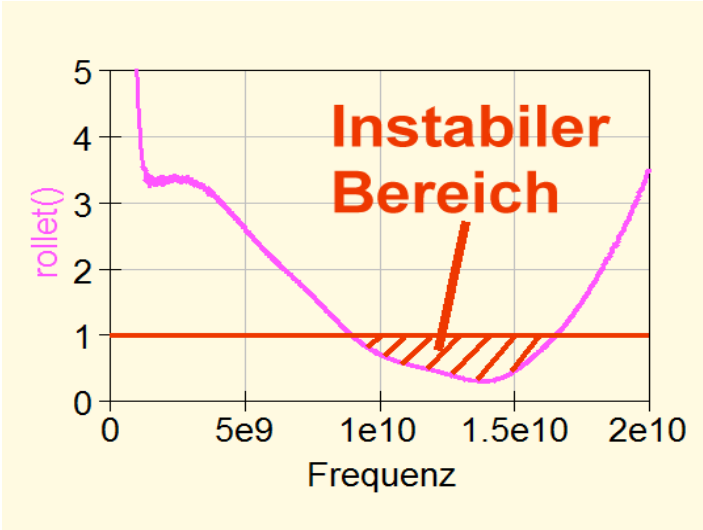
Die S-Parameter-Datei „**MGA635P8_55mA.S2P**“ holt man sich aus der **Agilent-Avago-Homepage** und kopiert sie in das Projekt, das für diese Schaltung angelegt wurde (Beispiel: „LNA“).
Die beiden Induktivitäten mit je 6,8 nH sind SMD-Ausführungen der Größe 0805, sie weisen etwa eine Güte Q = 30 bei f = 1 GHz auf. Die zugehörigen Verluste werden durch einen Reihenwiderstand mit R = 1,4 Ω berücksichtigt.

Die Umrechnung der S-Parameter in dB erfolgt wieder durch die passenden Gleichungen
dB_S11=dB(S[1,1])
dB_S21=dB(S[2,1])
dB_S12=dB(S[1,2])
dB_S22=dB(S[2,2])



Simuliert wird linear von 1 MHz bis 20 GHz (..dort hört das S-Parameter-File auf!) und es wird auch gleich die Rauschsimulation freigegeben.

Hier haben wir die S-Parameter der Schaltung.
Markiert ist **S21 = 21,9 dB bei 1,5 GHz.**



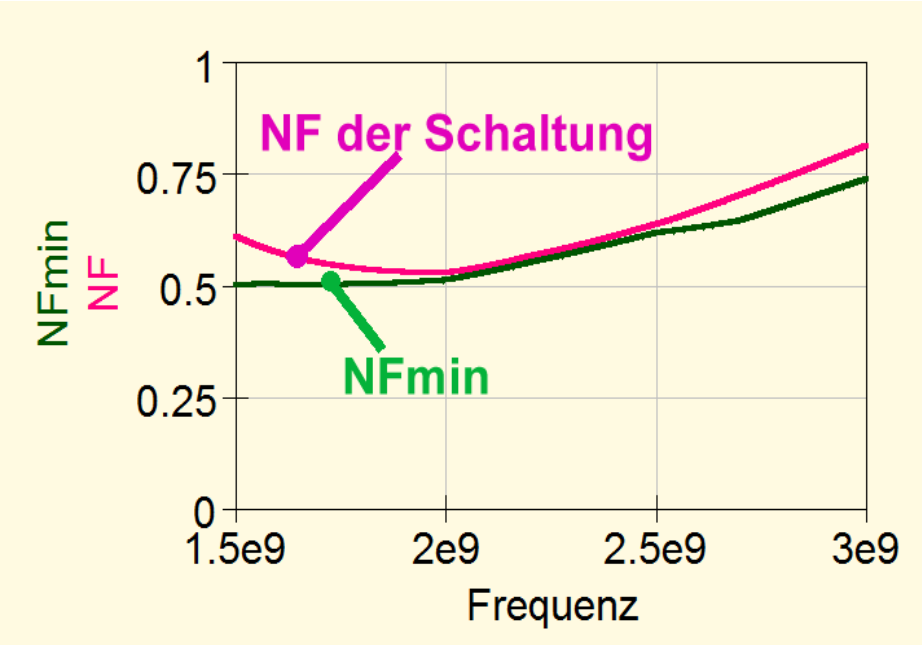
Und auf die Stabilität können wir auch gleich einen Blick werfen (= Darstellung des Rollet Stabilitätsfaktors „k“ bis 20 GHz).

Allerdings hat der Avago-Entwickler offenbar nicht den ganzen Frequenzbereich bis 20 GHz ausgetestet, denn zwischen 8 GHz und 17 GHz gibt es eine instabile Region.

```
! Noise Parameters
! Freq     FMIN     GAMMA OPT     Rn @
! (Hz)    (dB)    Mag      Ang   50 Ohm
1.900      0.40      0.200       95.5       0.0482
2.000      0.42      0.206       96.4       0.0552
2.200      0.49      0.205      113.2       0.0456
2.500      0.58      0.216      128.8       0.0472
2.700      0.62      0.214      163.5       0.038
3.300      0.82      0.292      172.7       0.038
3.500      0.77      0.289      174.7       0.0426
```

Der Hersteller liefert im Anhang des S-Parameter-Files auch Rauschinformationen für den Bereich von 1,9 GHz.....3,5 GHz und das sollte man ausnützen.

Sehen wir uns doch den Bereich von 1,5 GHz bis 3 GHz an.



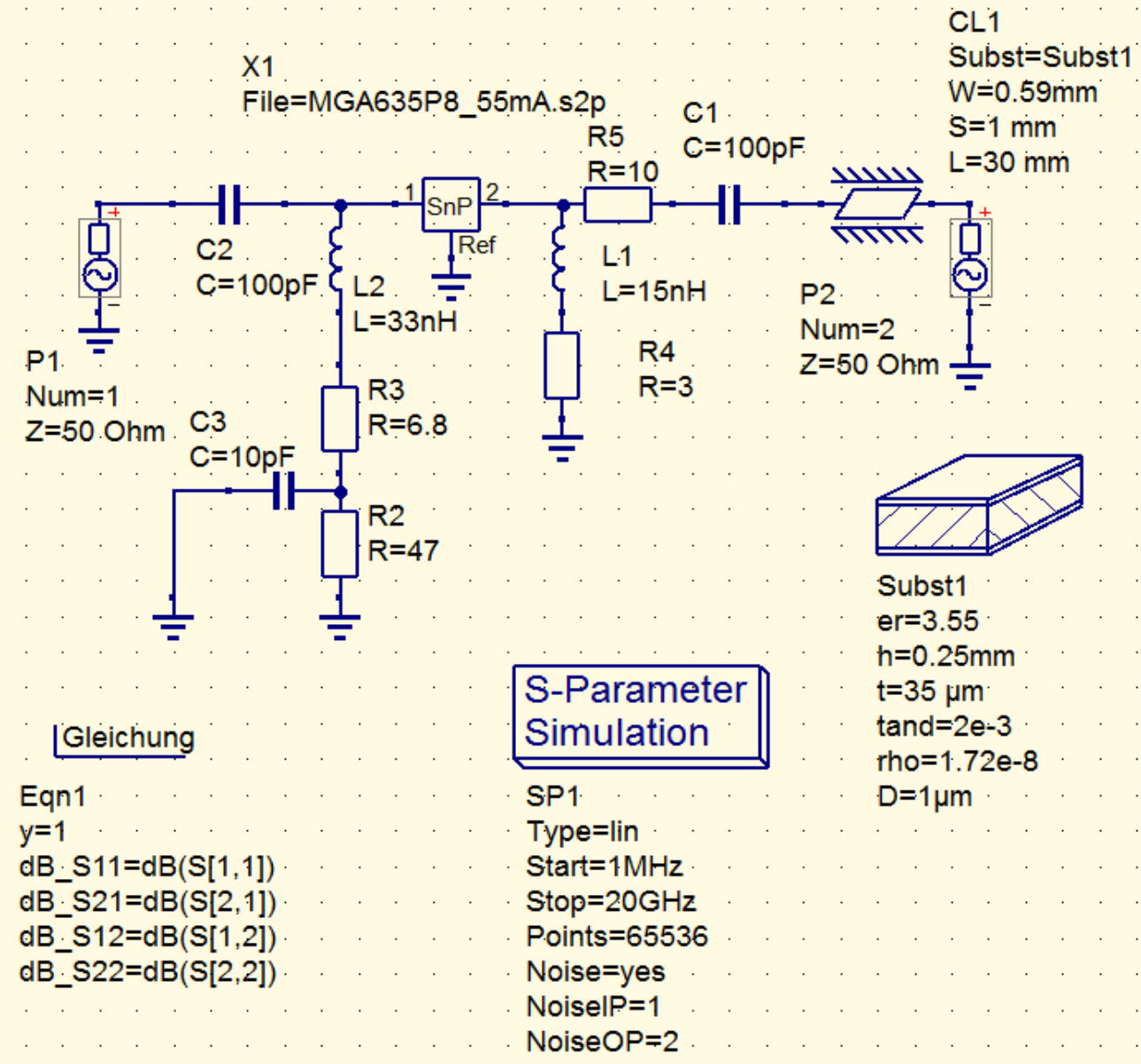
Es ist gut zu erkennen, dass die Schaltung für den WLAN-Bereich (2,4 GHz) ausgelegt ist. Da gibt es noch einige Arbeit, wenn wir bis zu 1 GHz herunterkommen wollen...

13.3. Weiterentwicklung der Schaltung

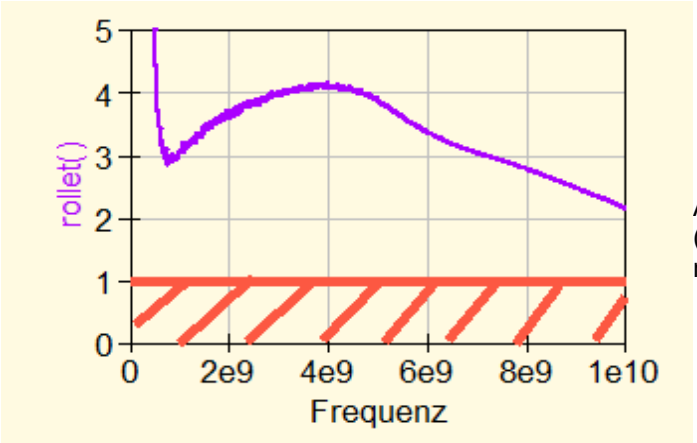
Nun wurde die Schaltung im **Frequenzbereich von 1 bis 1,7 GHz überarbeitet**, um dort das Eigenrauschen zu minimieren **Das erfolgte durch schrittweise Vergrößerung der Werte für L1 und L2 unter Beachtung der Simulationsergebnisse für die Rauschzahl „NF in dB“.**

Der angestrebte Wert von NF = 0,4 dB wurde mit **L1 = 33 nH / L2 = 15 nH** erzielt und die ausreichende Stabilität bis 10 GHz ergab in bekannter Weise ein kleiner Zusatzwiderstand mit 10 Ω in der Ausgangsleitung (…aber nahe beim Output-Pin des MMICs angebracht).
Die Verstärkung (= S21) sank dadurch etwas, aber die Forderung von 20 dB bei 1,7 GHz wurde noch nicht unterschritten.
Auch die beiden Koppelkondensatoren C1 und C2 (am Eingang und Ausgang) sind auf100 pF verkleinert, um die untere Grenzfrequenz anzuheben.
Die Ausgangs- Microstrip-Leitung (korrekt: „Grounded Coplanar Waveguide“ mit Z = 50 Ω durfte in der Simulation nicht fehlen. Ihre Daten wurden mit den „qucs Transmission Line Calculator“ bestimmt, der unter „Werkzeuge“ und „Leitungsberechnung“ zu finden ist. Sie besitzt eine Leiterbreite von 0,59 mm, einen Gap von 1 mm auf jeder Seite und eine Länge von 30mm, wobei die Platinendicke nun 0,25 mm beträgt.

Das ergab das endgültige Simulationsschaltbild.

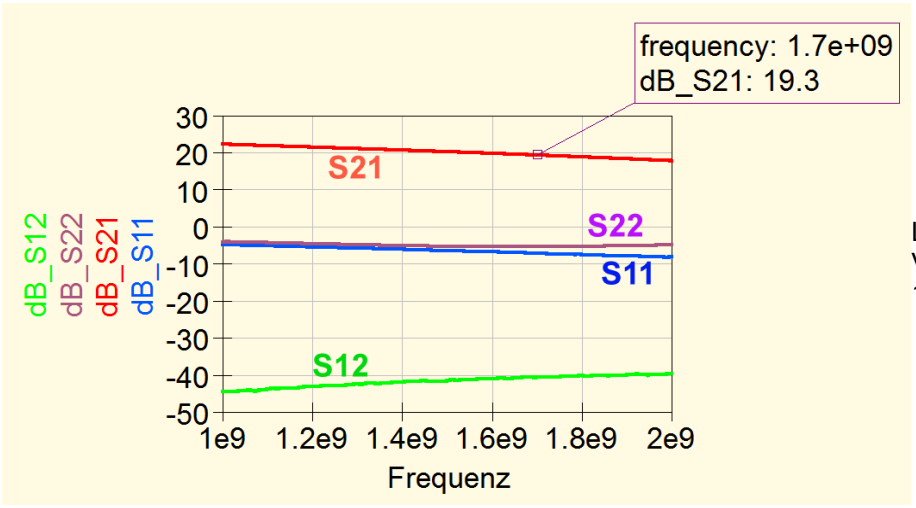


Damit wurden diese Rauschdaten als letztes Simulationsergebnis im Bereich von 1...2 GHz erreicht.
Wie man sieht, bleibt die Noise Figure NF bis 1,7 GHz unter 0,4 dB.

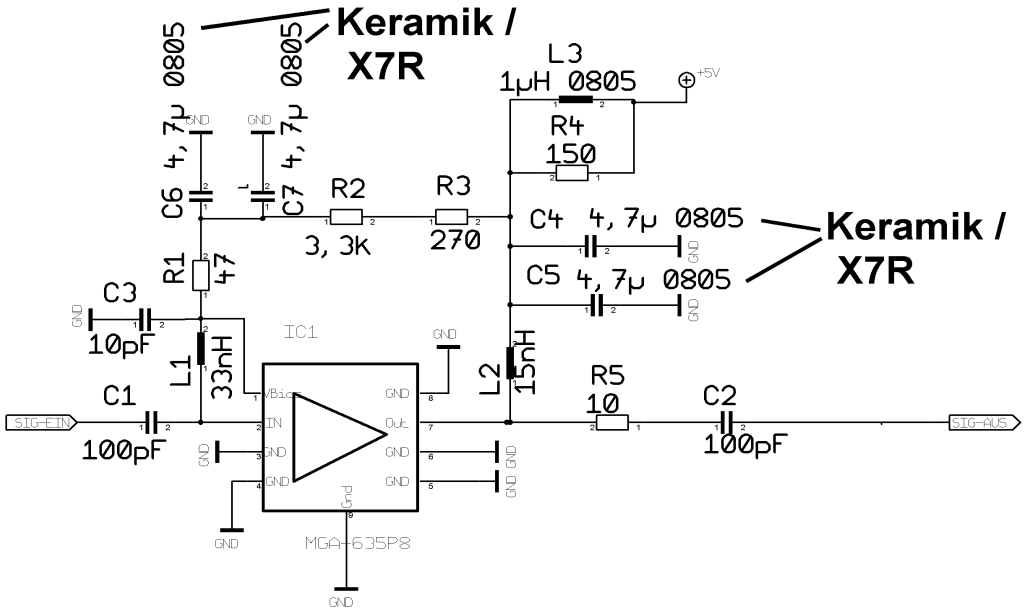


Auch die geforderte Stabilität
(= k ist größer als 1 bis 10 GHz) ist nun kein Thema mehr.

Dann wollen wir uns mal um die S-Parameter im gewünschten Frequenzbereich von 1....2 GHz kümmern.

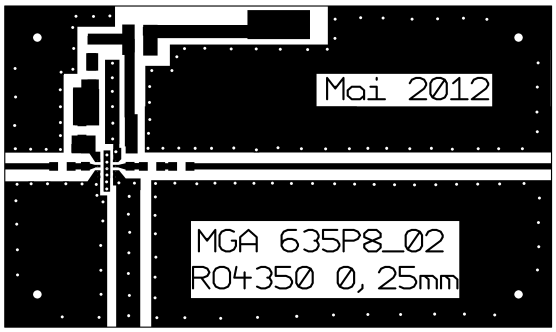


Leider wird da die gewünschte Verstärkung von 20 dB doch mit 19.3dB knapp unterschritten:



Die **praktische Schaltung** wurde gegenüber dem Agilent-Vorschlag modifiziert -- es kamen noch aufwendige Breitband - Siebungen bei der Versorgung dazu. Die endgültige Version sieht so aus.

Damit ging es an den Platinenentwurf (Werkstoff: Rogers RO4350B, beidseitig mit 35 µm Cu-beschichtet. Platinendicke = 10 mil = 0,254 mm). Die Außenmaße betragen 30 mm x 50 mm.



Gut zu erkennen ist links die kurze Eingangs-Streifenleitung bis zum DC-Trennkondensator vor dem Input-Pin des MMIC.
Die Ausgangs-Streifenleitung ist 30mm lang und beide Leitungen sind als „Grounded Coplanar Waveguide“ mit einer Leiterbreite von 0,59 mm und einem beidseitigen „Gap“ von je 1 mm ausgeführt. Diese Werte lieferte der kostenlos in qucsstudio integrierte Leitungcalculator.

Der zentrale Masse-Anschluss an der Unterseite des MMICs erforderte eine eigene streifenförmige, nur 0,6 mm breite, Masse-Insel mit 6 Durchkontaktierungen, die im Bild gut zu erkennen ist. Alle übrigen Masse-Inseln sind sorgfältig voneinander getrennt und jeweils mit genügend Durchkontaktierungen versehen (….wer diese Spielregel mit den getrennten Inseln samt vielen Durchkontaktierungen nicht beherzigt, der bringt sehr schnell die Schaltung zum Schwingen. Aber eigentlich ist das ein alter Hut: es handelt sich dabei schlicht und einfach um die auch bei tiefen Frequenzen empfohlene „Sternpunkt-Erdung“).
Übrigens: alle Durchkontaktierungen weisen einen Durchmesser von 0,3mm auf.

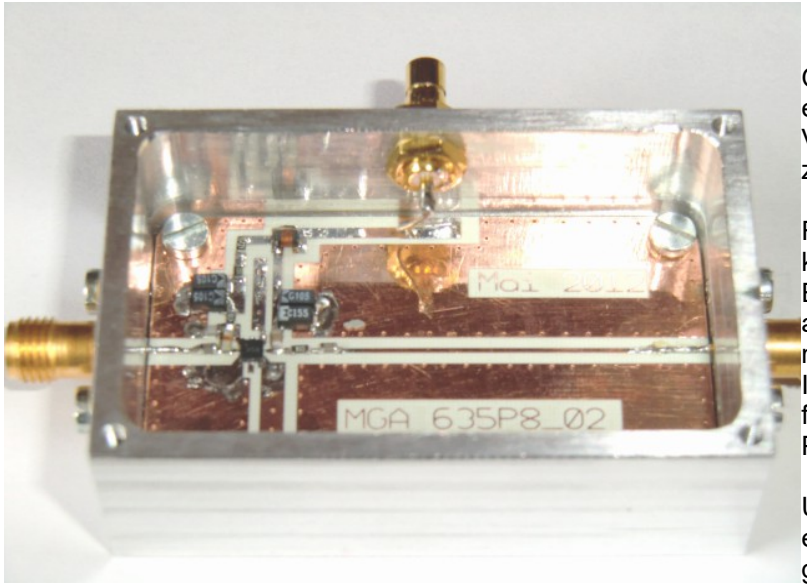
13.4. Der Prototyp

Da musste man sich erst mal den Kopf zerbrechen, denn bei der geringen Platinendicke von 0,25mm fällt die bisher immer daheim vorgenommene Durchkontaktierung mit versilberten Hohlnieten weg. Also blieb nur der Weg über eine professionell gefertigte und durchkontaktierte Leiterplatte und das ist teuer. Erst muss ein Gerber-Plot erstellt (= 1 Mausklick im Programm „Target“) und an den PCB-Hersteller gemailt werden, dann läuft das von selbst seinen Gang. Leider hat man als „Grundlast“ gleich mal die so genannten „Einrichtungskosten“ und eine Mindestbestellmenge von 4 Stück wegen der Mindest-Platinengröße für die Produktion. Sehr gut und kooperativ lief das aber beim „Ätzwerk München“ ab (Homepage: **www.aetzwerk.de**), denn die haben sogar das passende Leiterplattenmaterial (RO4350B mit 0,254 mm Dicke) an Lager. Aber: selbst ein „Minimalauftrag“ zum Niederhalten der Kosten ergab (incl. PCB-Material) eine Rechnung von 235 Euro für 4 Platinen….

Noch eine Bemerkung zum Leiterplattenmaterial: normalerweise ist für diesen Zweck „RO4003“ von Rogers der preisgünstigste und verlustarme Standardwerkstoff. Aber: er ist nicht „flammhemmend“, wenn es brennt und das wird immer mehr zur Vorschrift. Deshalb erhielt er entsprechende Zusatzstoffe und heißt dann „RO4350B“. Dadurch änderten sich auch seine elektrischen Daten etwas (ε_r steigt leicht, aber auch die Verluste werden höher. Bei 10 GHz gilt: loss tangent lt = 0,0027 für RO4003 und 0,0037 für RO4350. Bei 2,5 GHz findet man dagegen: lt = 0,0021 bzw. 0,0031).

Die Bestückung mit den Bauteilen der Größe 0603 funktioniert natürlich nur noch mit ruhiger Hand unter einer Stereo-Lupe oder einem passenden Mikroskop (Siehe [4]), und ohne SMD-Lötpaste in einer Injektionsspritze geht es nicht mehr ab. Sie wird in winzigen Mengen mit einem spitzigen Skalpell an den Lötstellen angebracht.
Außerdem ist eine schmale temperaturgeregelte Lötkolbenspitze mit 0,8 mm …1,5 mm Breite nötig, denn bei einer Kantenlänge von 2 mm und 4 Anschlüssen auf jeder Seite des MMIC’s hat man es mit Leiterbahn- und Anschlusspad-Abmessungen sowie einem gegenseitigem Pin-Abstand von 0,25 mm zu tun.

Und so sieht das Ergebnis nach dem Einbau in ein gefrästes Alugehäuse (Außenmaße = 35 mm x 55 mm) aus.



Oben im Bild ist die SMB-Buchse zu erkennen, über die eine Vorsorgungsspannung von +5V zugeführt wird.

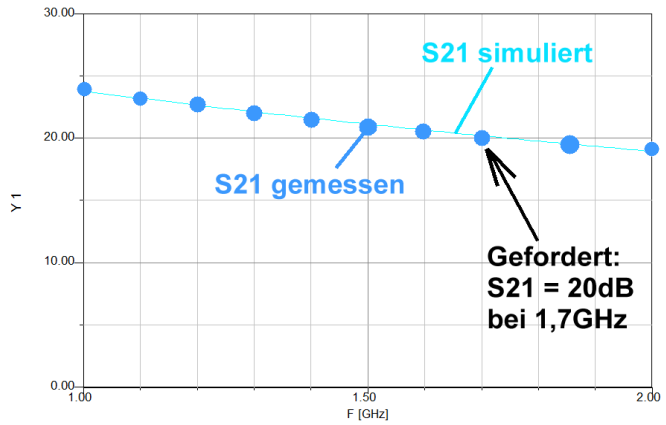
Für das Eingangs- und Ausgangssignal kommen spezielle SMA-Buchsen zum Einsatz, bei denen der Mittelleiter bereits abgeflacht ist (= ergibt den gewünschten reflektionsarmen Übergang vom runden Innenleiter innerhalb der Buchse zur flachen Microstrip-Leitung auf der Platine.

Unterhalb der Leiterplatte wurde noch ein selbst gefertigtes Aluminiumblech mit genau passender Dicke unterlegt, um die Microstrip-Leitung bis zum abgeflachten Mittelleiter der Buchse anzuheben und so eine korrekte und spannungsfreie Lötverbindung zu ermöglichen (…ein Tipp: das ist unbedingt nötig, denn es ist nicht möglich, am Pin der Buchse etwas zu biegen. Dieser dünne Pin und die flache Fahne brechen sofort und blitzschnell….).

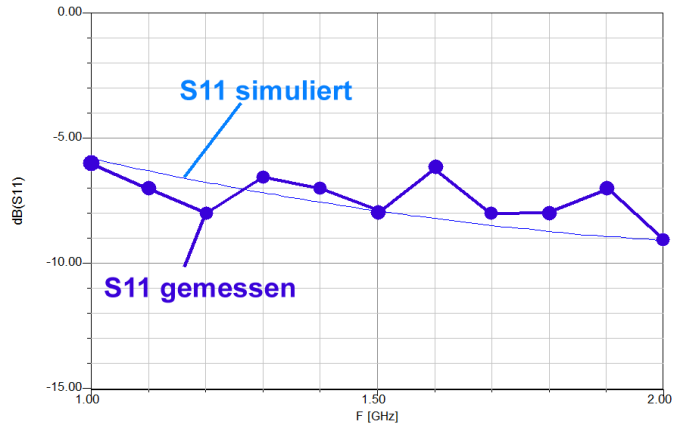
13.5. S-Parameter-Messungen am Prototyp

Hinweis:
Diese Ergebnisse wurden direkt aus dem Vortragsmanuskript übernommen und deshalb findet man in jedem Diagramm noch den zugehörigen, aber damals mit dem Ansoft Designer SV simulierten Parameter. Die Ergebnisse stimmen trotzdem...

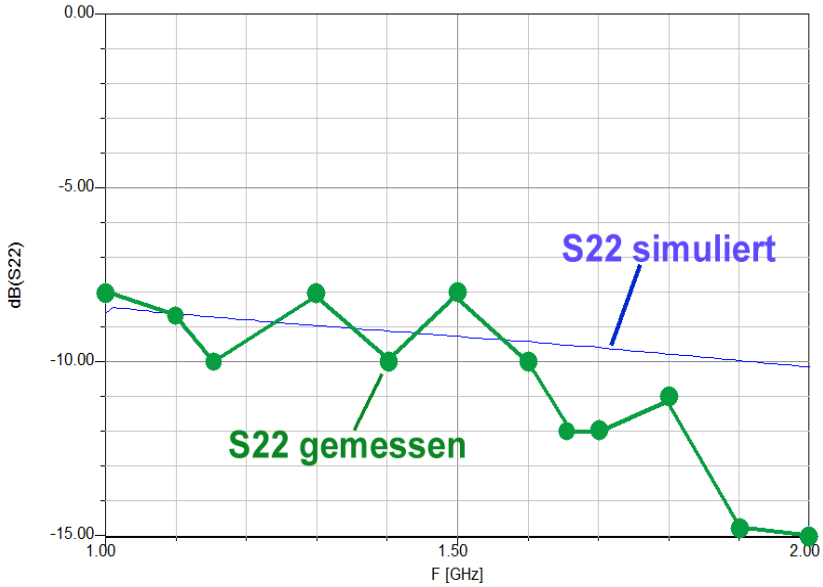
Sie erfolgten mit dem altbewährten Vektor-Analyzer (hp8410) und dem zugehörigen S-Parameter-Test-Set (hp8745A), wobei jedoch bei der S21-Messung ein 20 dB-Dämpfungsglied vor den Eingang des Prüflings geschaltet wurde, um Übersteuerung zu vermeiden.



S21 folgt genau der Simulation.



S11 schlängelt sich um den Simulationsverlauf herum



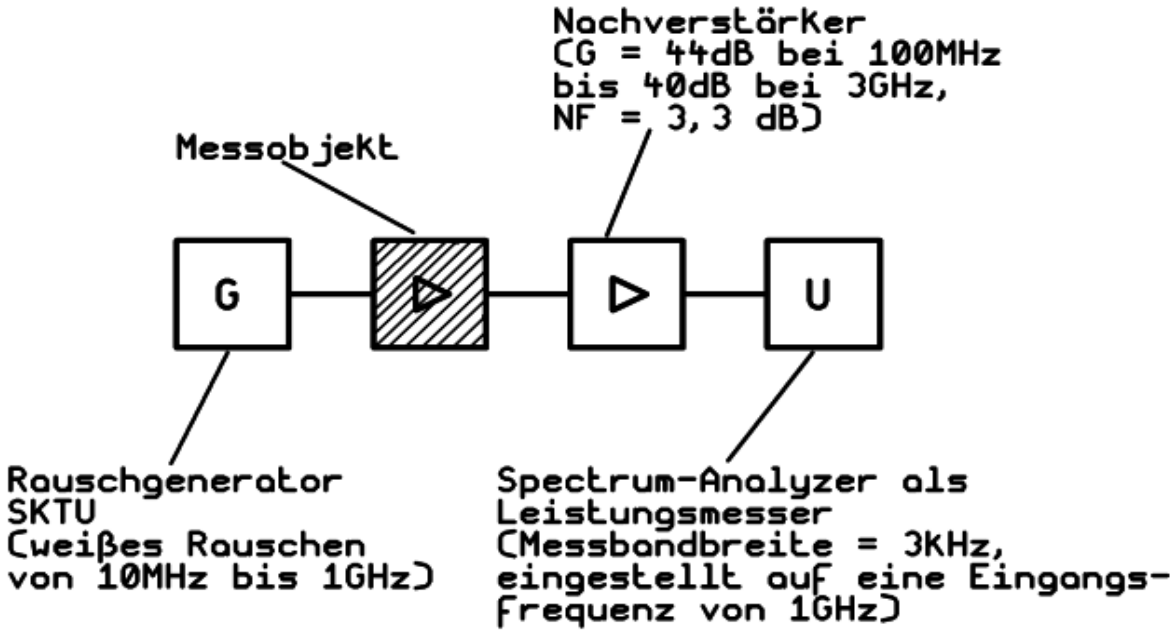
S22 ist ab 1,5GHz besser als erwartet.

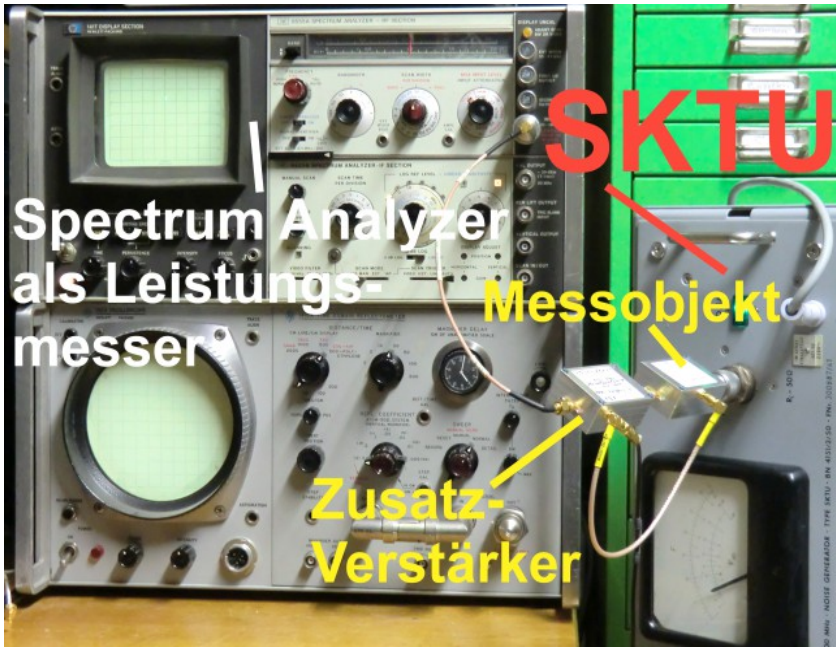
S12 wurde schließlich zu ca. -44dB im Bereich von 1…..2GHz bestimmt und ist damit etwas kleiner als der simulierte Wert zwischen -43 dB und -40 dB.

Dafür gibt es kein Bildchen....das lohnt nicht....

13.6. Das Rauschen

Bei der erwarteten und sehr kleinen Rauschzahl von maximal 0,4 dB war es für die vorhandene „Heimwerker-Ausrüstung“ schon sehr schwierig, überhaupt etwas zu messen. Trotzdem wurde ein Versuch gewagt und dazu folgender Aufbau für die Frequenz f = 1 GHz zusammengestellt:





So sieht diese Maschine in der Praxis (= in der Kellerwerkstatt) aus. **Dabei „sweept“ der Analyser nicht, sondern ist als reiner Messempfänger mit einer Bandbreite von 3 kHz genau auf die Frequenz f = 1 GHz eingestellt.**

Jetzt wird der Ausgangspegel des Rauschsenders solange vergrößert, bis sich **die am Analyser angezeigte Rauschleistung verdoppelt hat (= Leistungsanstieg um 3 dB bzw. Spannungsanstieg um den Faktor 1,41).** Dann ist das Eigenrauschen des Verstärkers gleich groß wie die zusätzlich vom Rauschsender gelieferte Leistung und die Rauschzahl NF in dB (oder der Rauschfaktor als lineares Verhältnis) kann am Instrument des Senders SKTU abgelesen werden.

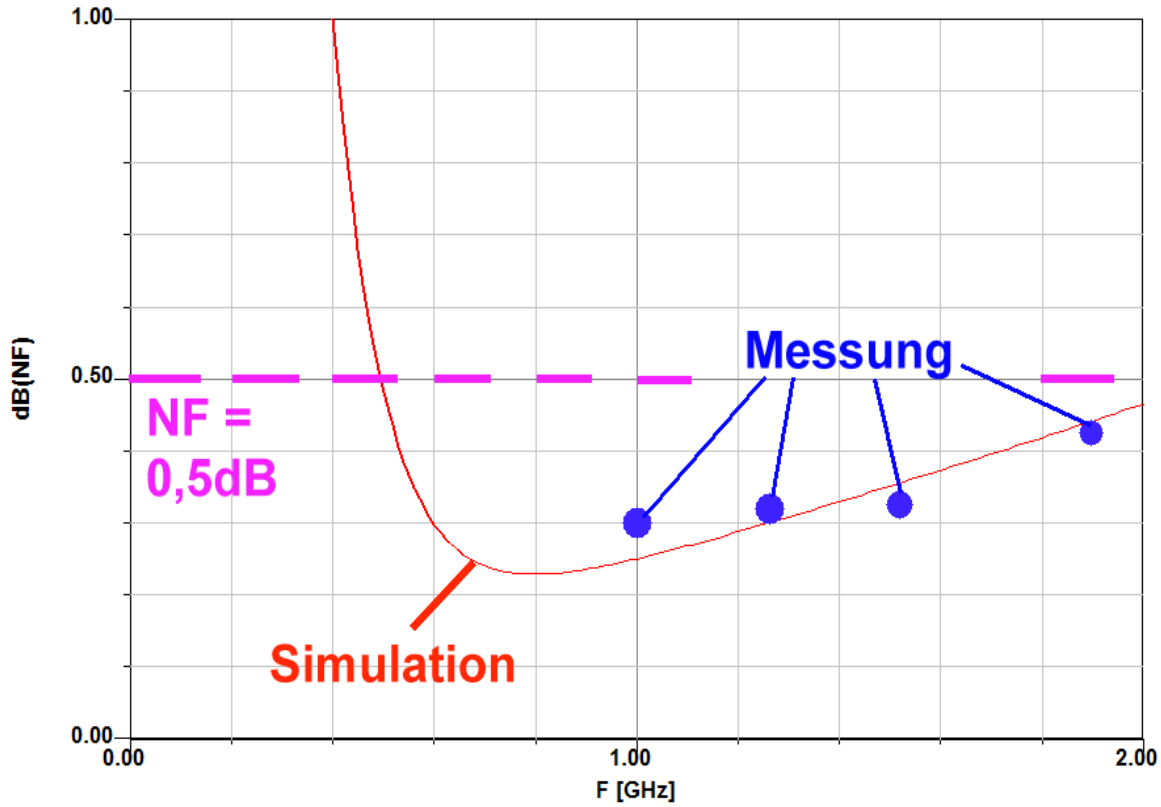
Mehrere Messungen ergaben Werte des Rauschfaktors irgendwo zwischen 1,1 und 1,2. Das entspricht einer „Noise Figure“ (= NF) des Verstärkers zwischen 0,4 und 0,8 dB -- aber wegen des sehr geringen Zeiger-Ausschlags beim SKTU kann dieses Ergebnis nur als unverbindlicher Hinweis betrachtet werden. Doch es stimmt wenigstens die Größenordnung.

Der Einfluss des nachgeschalteten Zusatzverstärkers kann wegen der hohen Verstärkung des Messobjektes (= 24 dB bei 1 GHz) vernachlässigt werden. Eine Kontrollrechnung (= „Cascaded Noise Figure“) ergab deshalb einen Anstieg bei NF um nur 0,02 dB durch diese folgende Stufe…
Ein Zuhörer gab aber nach dem Vortrag noch einen Tipp aus eigener Messerfahrung heraus:

Schaltet man zwischen Rauschgenerator und Messobjekt noch ein hochgenaues Dämpfungsglied, so steigt die Rauschzahl zusätzlich genau um dessen Dämpfungswert an. Diesen Dämpfungswert braucht man dann nur noch von der Rauschgenerator-Anzeige subtrahieren. Geht wesentlich genauer und bequemer..

Das wurde mit einem 6dB-Pad daheim getestet, das ja gut zum Maximal-Ausschlag des Rauschgenerators mit 8 dB passt. Ergebnis: nun zeigt der Rauschgenerator ca. 6,4 ….6,5 dB und damit haben wir einen Wert von NF = ca. 0,45 dB.

Für noch genauere Messungen ist man -- wie immer -- auf die Hilfe von Freunden angewiesen, denen moderne Rauschmessplätze mit hoher Auflösung und Genauigkeit zur Verfügung stehen.

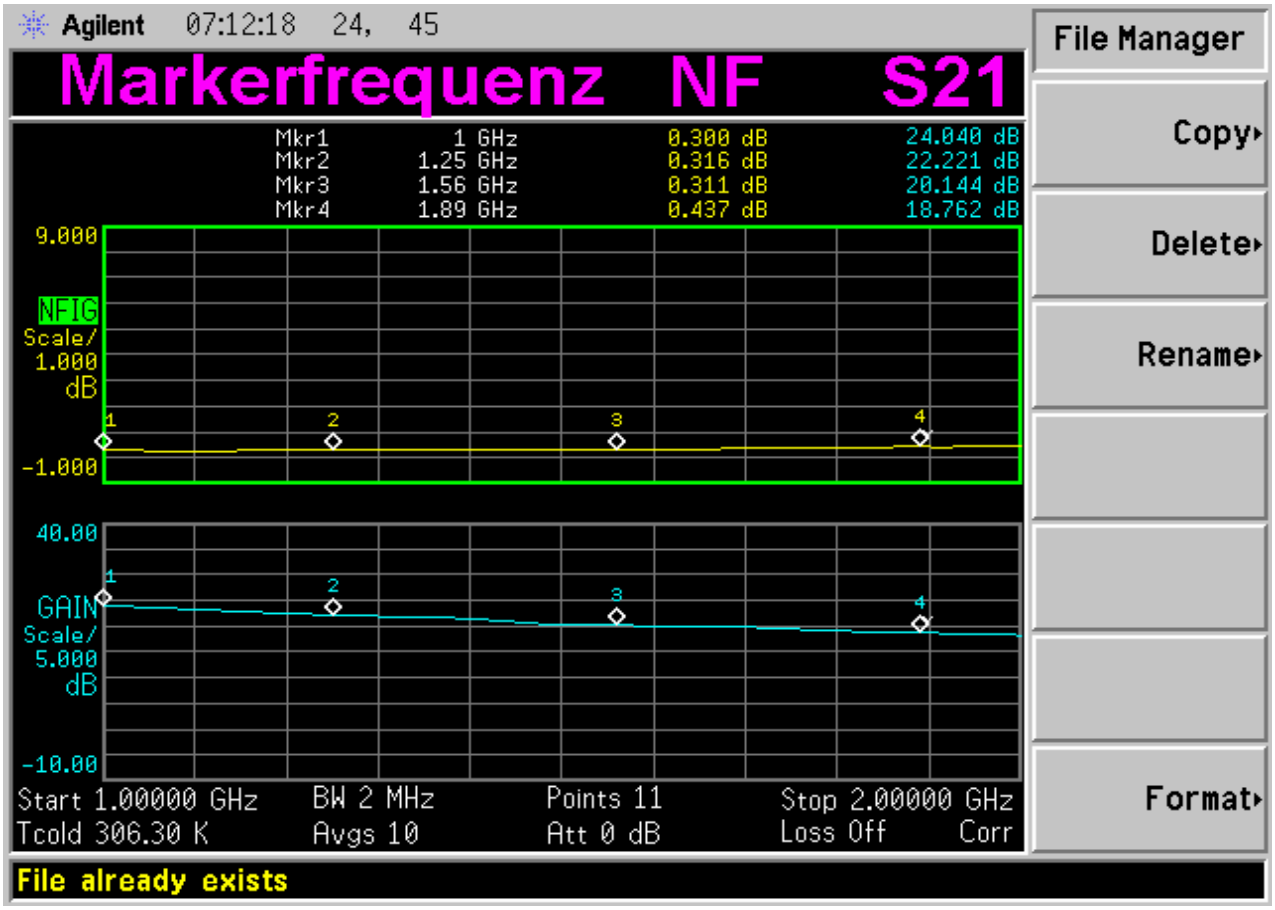


Und einer dieser Freunde (= Ulli Kafka mit seiner Firma „Eisch-Kafka“ in Ulm) bereitete dann mit diesem Ergebnis sowie der Einstufung als „hervorragender Verstärker“ dem Autor eine große Freude.

Die gemessene Rauschzahl bleibt nämlich tatsächlich bis 1,7 GHz noch unter 0,4 dB.

Da bleibt nur übrig, „Gott sei Dank“ zu sagen und allen Mitwirkenden zu danken.

Zum Abschluss folgt noch die Email mit dem Originalausdruck der Rausch- und S21-Messung, der für gehobene Stimmung sorgte -- besonders, wenn man die darin angegebene „Kalt-Temperatur“ (Tcold) mit 306,3 K, also mit etwa +33 Grad Celsius bei der Messung bedenkt….



Für die Spezialisten ein Nachtrag aus dem Datenblatt des **MGA-635P8 für I = 55 mA bei 2,5 GHz:**

Output IP3	**=**	**+35,9 dBm**
P1dB_out	**=**	**+22 dBm**

Wenn man da anfängt, über die Möglichkeiten und Eigenschaften einer zweistufigen Version nachzudenken…..Platz auf der Platine wäre ja genug…..

Literatur:

[1]: Datenblatt und S-Parameter-Files des MGA-635P8 aus der Homepage von Avago Technologies

[2]: Application Note von Avago: “MGA-635P8 GaAs ePHEMT MMIC 2.5 GHz Low Noise Amplifier with Superior Noise and Linearity Performance”

[3]: Application Note von Avago: “MGA-635P8 GaAs ePHEMT MMIC 3.5 GHz Low Noise Amplifier with Superior Noise and Linearity Performance“

[4]: Bernd Kaa: „Anleitung zum Löten von SMD-ICs mit 0,5 mm Pin-Abstand“. UKW-Berichte 2 / 2011, Seite 99 bis 104.

14. RF-Mixer (= HF-Mischer)

14.1. Einführung

Zunächst muss eine grundsätzliche Sache geklärt werden:

Der RF-Mixer hat überhaupt nichts mit einem „Audio-Mixer“ gemeinsam, bei dem nur verschiedene Signale zusammenaddiert werden!

Ein RF-Mixer (= HF – Mischer) ist grundsätzlich ein „Threeport“ –Baustein (= Dreitor) mit drei Anschlüssen, bei dem zwei Eingangssignale miteinander multipliziert werden.

Dadurch kann man z. B. die Frequenz von Signalen verändern, ohne dass die darin enthaltenen Informationen darunter leiden (Fachausdruck: „Frequenz-Umsetzung“).

Er besitzt also **zwei Eingänge** und **einen Ausgang**:

a) Den Eingang für die Information, deren Frequenz umgesetzt werden soll (Name: **RF = radio frequency signal**). Es soll eine **kleine Amplitude** aufweisen.

b) Den Eingang für das Signal, das die Frequenzumsetzung bewirkt (Name: **LO = local oscillator signal**). Es muss stets eine **große Amplitude** aufweisen

c) Den Ausgang, an dem die umgesetzten Signale zur Verfügung stehen (Name: **IF = intermediate frequency**)

Die Multiplikation der beiden Eingangssignale hat ein erstaunliches Ergebnis, denn dafür gibt es (bei Sinusform der beiden Eingangsspannungen) den „Cosinussatz“ in der Mathematik:

$$\cos(\alpha) \bullet \cos(\beta) = \frac{1}{2}[\cos(\alpha+\beta)+\cos(\alpha-\beta)]$$

Wenn man die rechte Seite der Formel betrachtet, erkennt man:

Nach dem Durchlaufen dieser Schaltung sind die beiden zugeführten Eingangssignale am Ausgang plötzlich verschwunden. Dafür beobachtet man dort ihre Summenfrequenz und ihre Differenzfrequenz!

Die praktischen Mixerschaltungen unterscheiden sich je nach Aufwand und vorgesehenem Einsatzgebiet und es existieren viele Varianten. Man kann sie jedoch in **drei Gruppen** einteilen:

a) Echte Multiplizierer

b) Additve Mixerschaltungen

c) **Geschaltete Mixer** (= Switched Mixers = Single Balanced Mixer und Double Balanced Mixer usw. Sie arbeiten beim LO-Signal mit einem rechteckförmigen Signalverlauf).

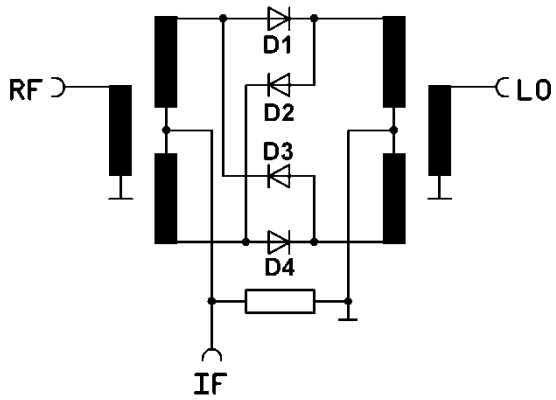
Wichtig:
Der „**echte Multiplizier**er“ ist auf allerhöchste Linearität gezüchtet und versucht, den obigen Cosinussatz möglichst perfekt -- auch bei großen Signalamplituden -- umzusetzen. Das schafft er, aber oft nur bis zu einigen MHz. Er wird z. B. für Aufgaben verwendet, bei denen „amplitudenmoduliert“ wird oder wo es auf minimalste Verzerrungen ankommt. Bei dieser Schaltung dürfen beide Eingangssignale gleich groß sein.

Der „**Additive Mixer**“ ist sehr einfach aufgebaut und schiebt den Arbeitspunkt bei einem Bauteil mit krummer Kennlinie (= Diode, Transistor, FET…) im Rhythmus des großen LO-Signals zwischen großer und kleiner Kennlinien-Steilheit hin und her. Dadurch ändert sich die Amplitude des gleichzeitig vorhandenen kleinen RF-Signals ebenfalls im Rhythmus des LO-Signals und das bedeutet wieder die gewünschte Multiplikation beider Signale.

Vorteil: einfach und billig. Wird oft bei sehr hohen Frequenzen (z. B. 50 … 100 GHz) eingesetzt, bei denen andere Mixerschaltungen nicht mehr so recht mitkommen..

Nachteil: steigt die Amplitude des kleinen RF-Signals an, so entstehen Oberwellen, die sich wieder zusätzlich mit den Oberwellen des LO-Signals mischen. Das führt zu einem kompletten „Urwald“ an zusätzlichen neuen und meist unerwünschten Mischprodukten und Signalen.

Die „**Switched Mixers**“ (insbesondere der Double Balanced Mixer = DBM) bilden die Arbeitspferde der Nachrichtentechnik. Dieses in der gesamten Kommunikationstechnik unentbehrliche Teil gibt es in **passiver wie auch in aktiver Version.**



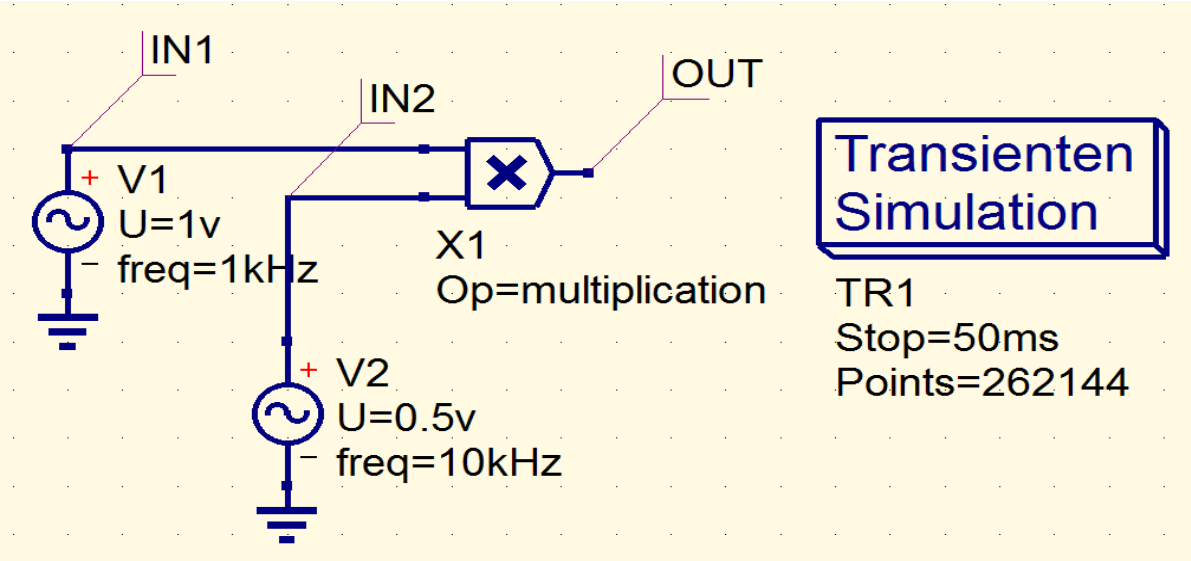
Am weitesten verbreitet ist die **passive Version des DBMs** als „Ringmodulator“ und wir wollen uns mal ihre Prinzipschaltung ansehen. Sie besteht aus einem Ring von 4 schnellen Schottky-Dioden (bitte aufpassen: sie sind **NICHT** so gepolt wie beim Brückengleichrichter!) sowie zwei Übertragern. Jeder Übertrager ist mit drei gleichen Wicklungen ausgestattet. Die beiden Sekundärwicklungen werden dabei in Reihe geschaltet und die „Mittenanzapfung“ herausgeführt.

Und so funktioniert das mit der Multiplikation: durch das (große) LO-Signal werden bei dessen positiver Halbwelle -- also bei der Amplitude „+1“ -- die beiden in Reihe liegenden Dioden D2 und D4 eingeschaltet. Bei der negativen Halbwelle dagegen -- sie entspricht der Amplitude „-1“ -- werden die beiden Dioden D1 und D3 als elektronische Schalter betrieben. So wird das von links kommende RF-Signal entweder über die untere bzw. die obere Trafowicklung zum IF-Ausgang geleitet. Da die Spannungen dieser beiden Sekundärwicklungen jedoch gegenphasig sind, erreicht man auf diese Weise ein **Umpolen des RF-Signals im Rhythmus des LO-Signals. Und genau das entspricht der Multiplikation einer Sinusspannung mit einem Rechtecksignal, das die Amplitude „+1“ bzw. „-1“ besitzt.**

Vorteile: sehr verzerrungsarm. Bei richtiger Konstruktion sehr breitbandig (z. B. von Null bis 5 GHz, in Spezial-Microstrip-Versionen bis weit über 50GHz) und auch von der Anpassung her ohne Probleme. Braucht keine Betriebsspannung, verlangt aber an allen Ports möglichst exakt den Systemwiderstand von 50 Ω als Innen- oder Abschlusswiderstand. Wird von vielen Firmen in großen Stückzahlen hergestellt und vertrieben, kostet also nicht viel. Ist im Einsatz völlig laiensicher und kann wie ein Bauklötzchen benützt werden, da er meist schon mit drei passenden HF-Buchsen (BNC- oder N- oder SMA-Steckverbindern etc.) ausgeliefert wird.

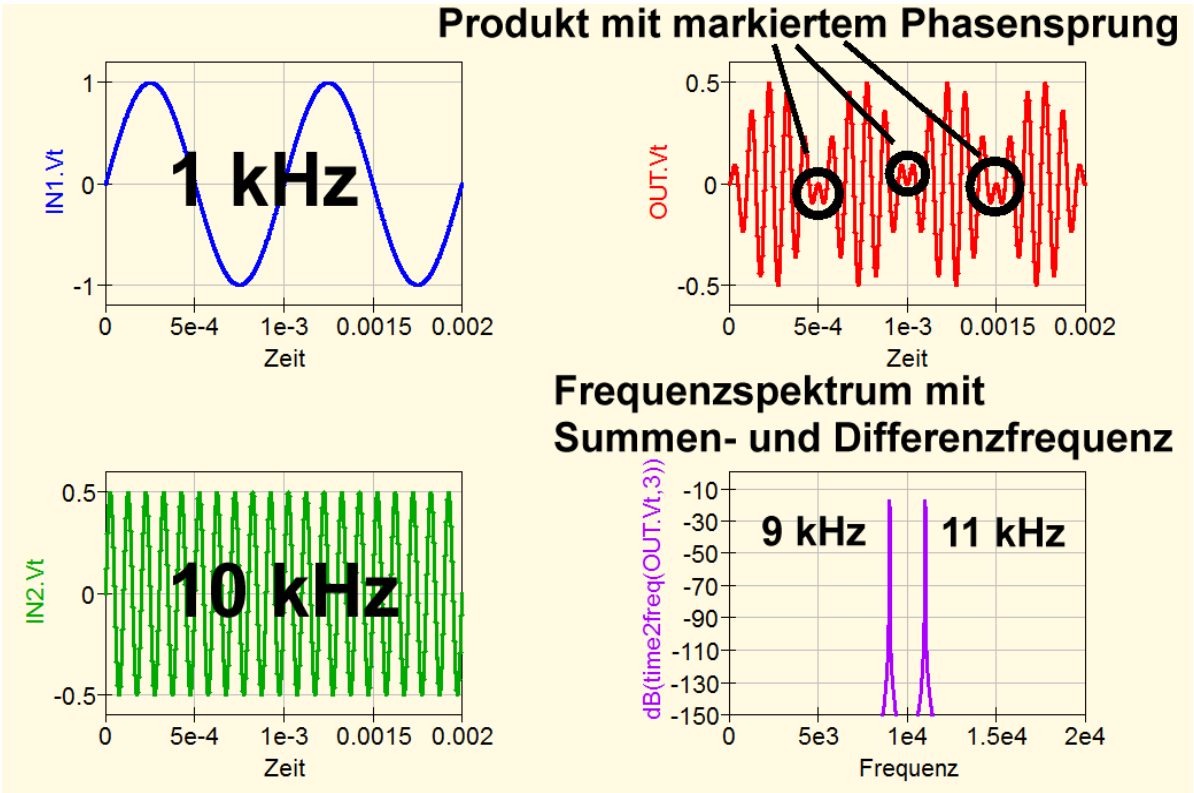
Nachteile: wegen der fehlenden Betriebsspannung **muss die „Schaltleistung“ für die Schottky-Dioden komplett vom LO-Signal aufgebracht werden** -- es sind also recht große LO-Pegel nötig, um die Dioden schnell ein- und auszuschalten. Es ist keine Verstärkung möglich, deshalb besitzt der Baustein grundsätzlich eine **Dämpfung** (typischer Praxiswert: zwischen 5 und 8dB) und verschlechtert deshalb das Rauschverhalten eines Systems. Da kann als Konsequenz: mehr Vorverstärkung bei einem Empfänger nötig sein.

14.2. Simulation eines echten Multiplizieres



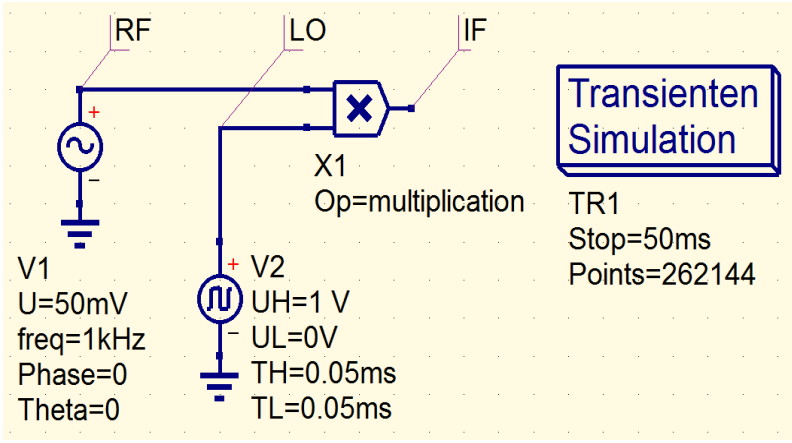
Das ist keine schwierige Übung, da uns in „**Komponenten / Systemkomponenten“** der Baustein „**Operation**“ zur Verfügung steht. Allerdings muss man nach dem Absetzen wieder „mit rechts“ auf das Schaltzeichen klicken und „Eigenschaften editieren“ aufrufen. Dann kann man sehr einfach auf „**multiplication“** umstellen. Und die bereits voreingestellte Zahl von 2 Eingängen ist genau richtig.

Am Ausgang können wir uns die beiden Eingangssignale, das Ergebnis der Multiplikation und schließlich auch das Frequenzspektrum mit Summen-und Differenzfrequenz ansehen.
Bitte genau beim Ausgang hinschauen: durch die Multiplikation wird nicht nur die Amplitude des 10 kHz-Signals im Rhythmus der 1 kHz-Spannung verändert. Sobald beim 1 kHz die negative Halbwelle beginnt, wird sich auch die Phase der 10 kHz-Spannung umkehren („Phasensprung“).



14.2.1. Der Multiplizierer als Switched Single Balanced Mixer

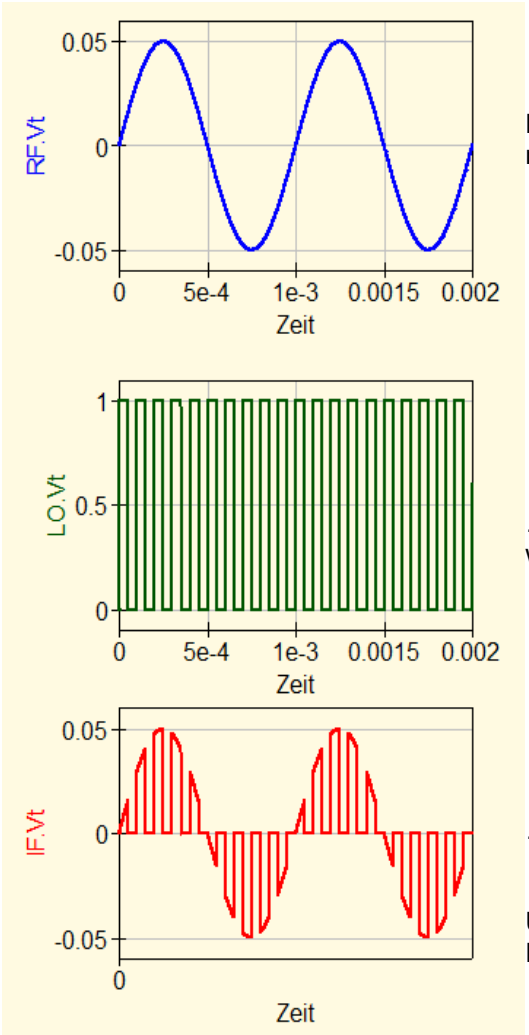
Das ist eine einfache Übung, denn wir müssen nur auf folgende Betriebsart umsteigen:



a) Das Signal „**RF**“ stellt das sogenannte „**RF-*Signal“ (= Radio Frequency Signal)** dar. Es besitzt (im Normalfall) nur eine kleine Amplitude von weniger als 50 mV, da es in der Frequenz umgesetzt werden soll und dabei nur möglichst wenig unerwünschte Effekte und Verzerrungen entstehen dürfen.

b) Das Signal „**LO**“ weist Rechteckform auf, heißt **„LO-Signal“ (= Local Oscillator Signal)** und dient als „**EIN / AUS -Schaltspannung“**. Deshalb findet man dort Amplituden bis zu einigen Volt (...denn damit müssen z. B. schnelle Schottkydioden in Picosekunden aus- und eingeschaltet werden). Hier ist es ein Puls-Signal mit 1 V Spitzenwert und einem Minimalwert von Null Volt.

c) Das Ausgangssignal hat die Bezeichnung „**IF“ (= Intermediate Frequency).**

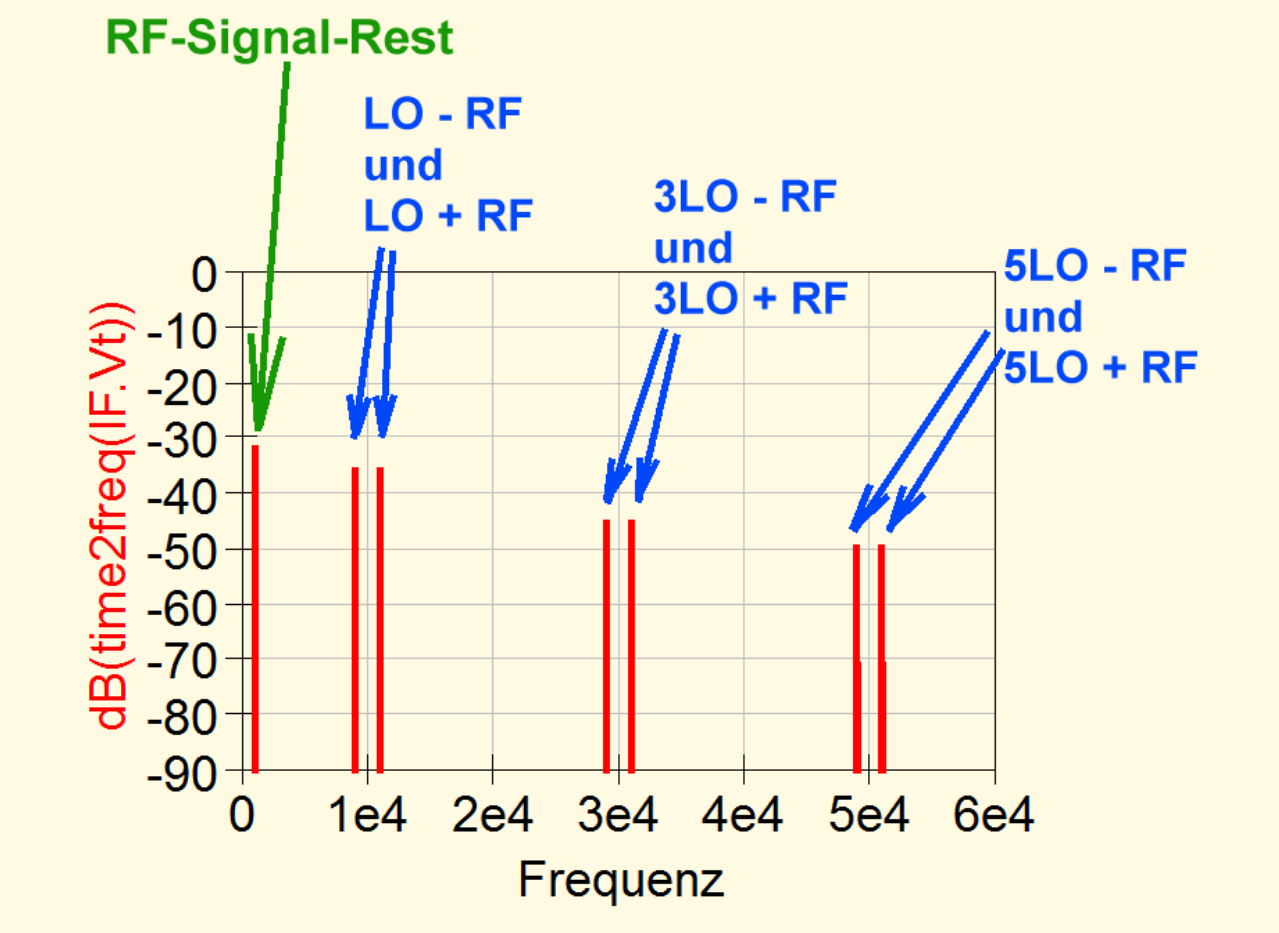


Das RF-Signal mit einem Spitzenwert der Urspannung von 50 mV und der Frequenz f = 1 kHz....

….wird durch das LO-Signal (das wie ein EIN / AUS-Schalter wirkt)....

….sozusagen „**zerhackt**“.

Und es wird interessant sein, das zugehörige neue Frequenzspektrum über die FFT zu untersuchen.



Zuerst finden wir bei 1 kHz noch die RF-Spannung, wobei ihre Amplitude durch die „Schalterei“ vermindert wurde.

Dann folgen Linien-Pärchen, die sich um die Frequenzen **10 kHz / 30 kHz / 50 kHz** usw. herum gruppieren. Ihre Ursache ist leicht zu finden:

Das LO-Signal als symmetrisches Pulssignal enthält nach Fourier die Grundwelle, die dreifache Grundwellenfrequenz, die fünffache Grundwellenfrequenz usw. Aber nicht nur Grundfrequenz, sondern auch jede dieser „Oberwellen“ wird mit dem RF-Signal multipliziert und liefert als Produkt die zugehörige Summen - und Differenzfrequenz.

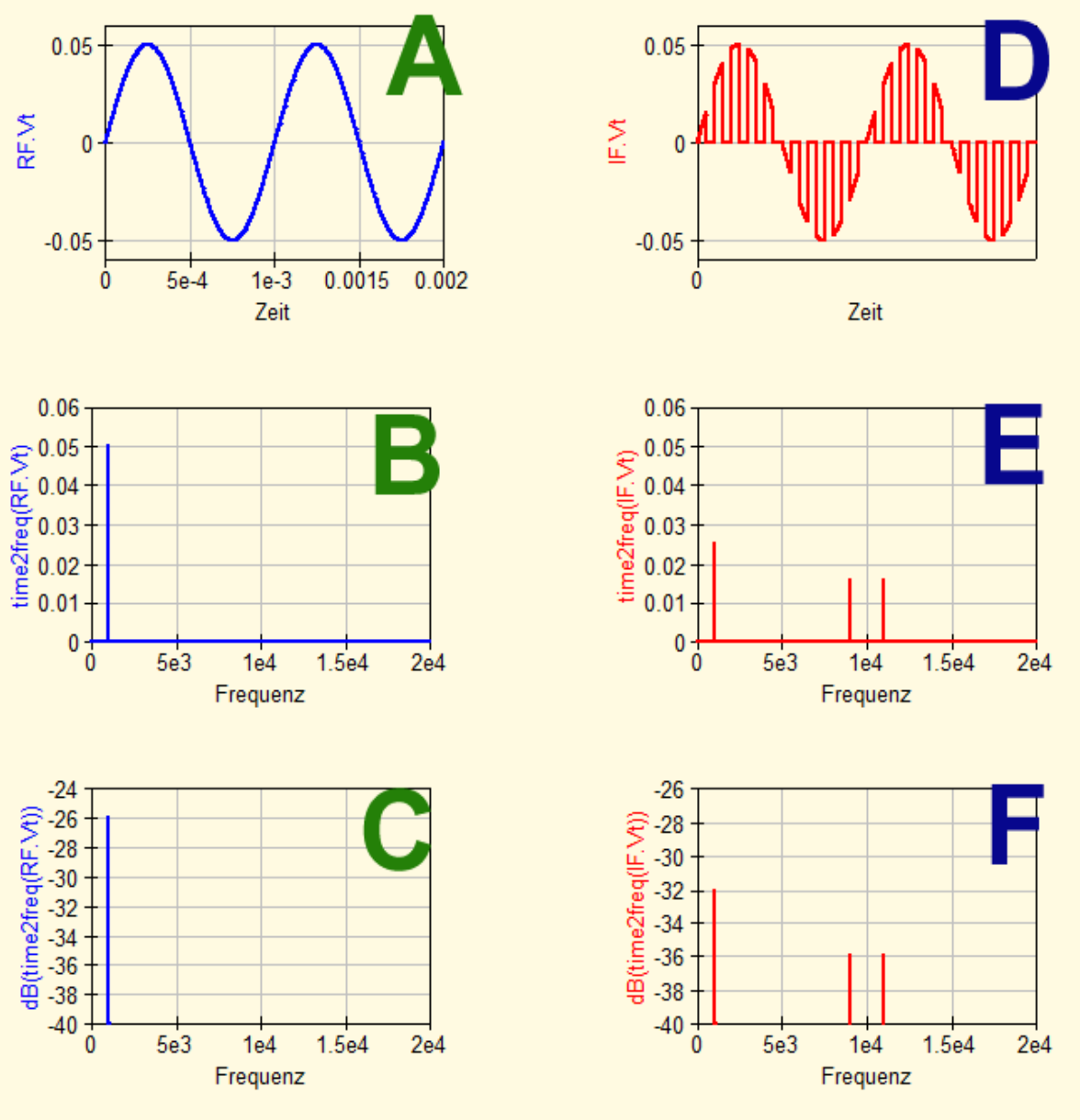
Das LO-Signal selbst taucht am IF-Ausgang nicht auf.

Auf der nächsten Seite wollen wir die Verhältnisse noch etwas genauer untersuchen und folgende Fragen beantworten:

a) Wie groß ist der RF-Rest im Vergleich zur Signalamplitude am RF-Eingang?

b) Wie groß ist die „Umsetzdämpfung in dB“ für die (meist verwendeten) beiden Seitenfrequenzen „LO – RF“ und „LO + RF“?

Lösung:



In der linken Spalte sehen wir in **Bild A** das **RF-Signal** mit einem Spitzenwert von 50 mV bei f = 1 kHz.

Bild B zeigt das **Ergebnis der FFT** in linearer Darstellung. Auch hier wird ein Spitzenwert von 50mV angezeigt.

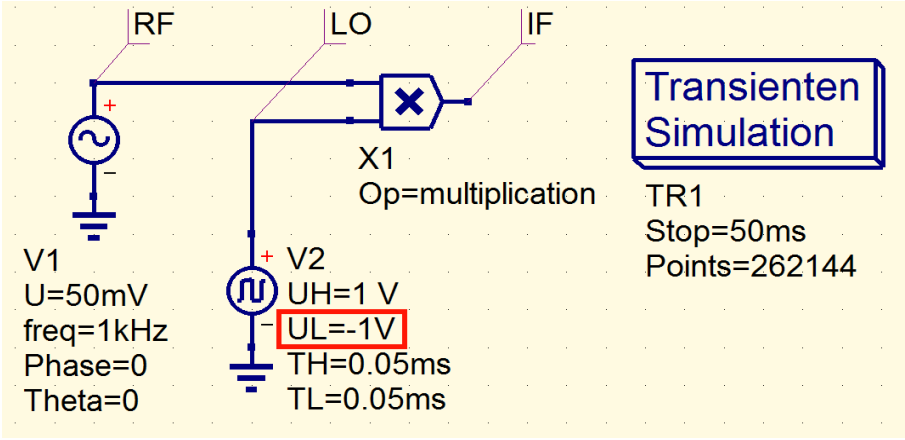
Bild C ist dagegen eine „**dB-Darstellung**“, wobei als Referenz ein Wert mit **„1 Volt“** verwendet wird. Sie ergibt -26 dBV (...sprich: **-26 dB Volt**).

Am IF-Ausgang des Mischers (**Bild D**) ist der RF-Anteil durch das Ein- und Ausschalten auf die Hälfte des Eingangswertes gesunken und beträgt nun laut **Bild „E**“ nur noch 25 mV bei f = 1 kHz.

Bei der **dB-Darstellung** in **Bild „F“** muss deshalb die Amplitude dieser **Spektrallinie um 6 dB gegenüber Bild „C“ abgenommen** haben.

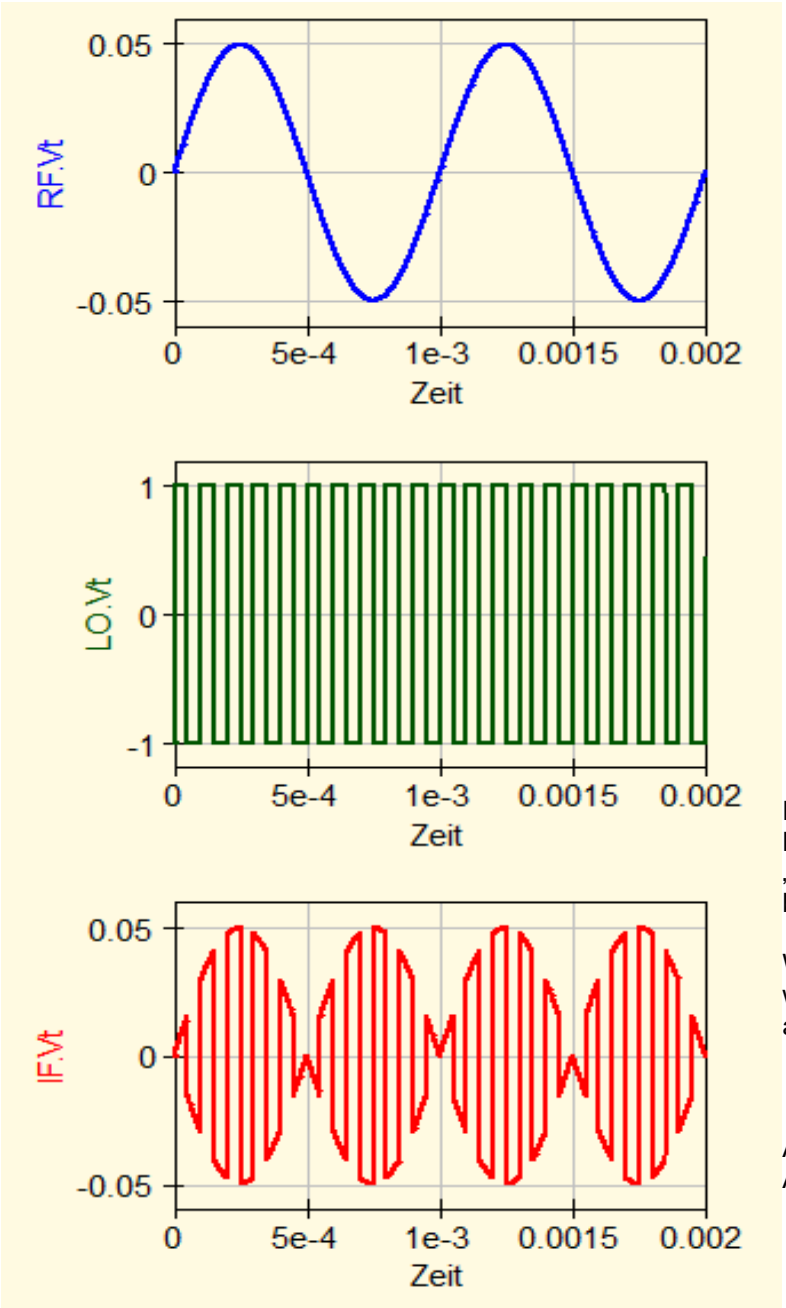
Vergleicht man endlich die Amplitude des RF-Signals in dB am Eingang (**Bild „C“)** mit den beiden erzeugten Seitenfrequenzen von 9 kHz und 11 kHz in **Bild „F“**, so findet man einen **Unterschied von 10 dB. Das ist die Umsetzdämpfung eines solchen „Eintakt-Mixers“.**

14.2.2. Der Multiplizierer als Switched Double Balanced Mixer



Das LO-Signal ist nun ein **symmetrisches Rechtecksignal** mit den Amplituden „+1V“ und „-1V“.

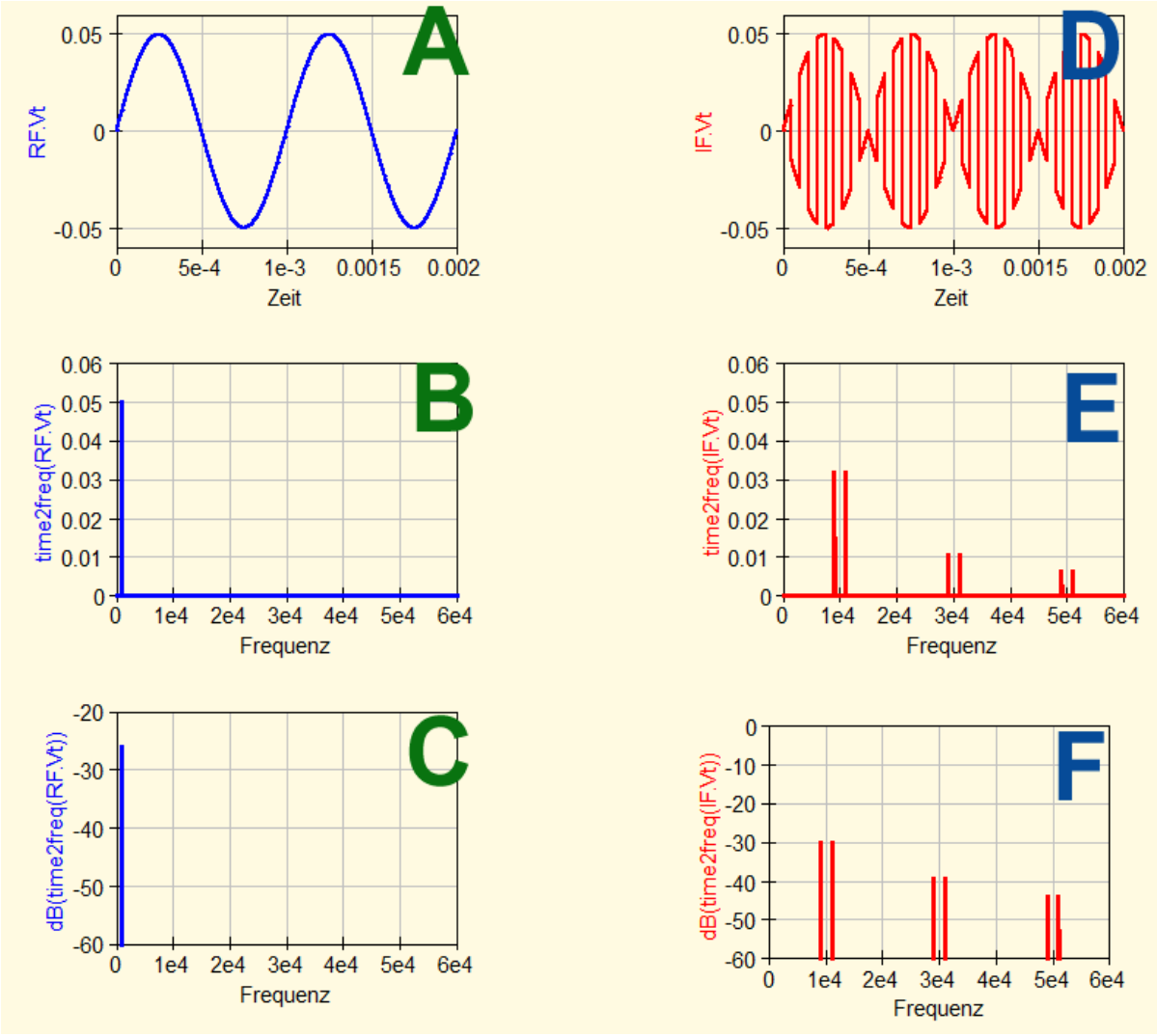
Dadurch wird der Multiplizierer vom EIN / AUS-Schalter zum „Umpoler.



Hier ist (im Vergleich mit dem Single Balanced Mixer“) schön zu sehen, was mit „**Umpolen**“ gemeint ist und was es bewirkt.

Wer genau hinschaut, erkennt auch hier wieder den „**Phasensprung**“ aus dem ersten Kapitel.

Auf dem nächsten Blatt folgt wieder die Analyse der Signale.



Zu A): Das **RF-Signal** weist wieder den bekannten **Spitzenwert von 50 mV bei f = 1 kHz** auf.

Zu B): Diesen Spitzenwert von 50 mV finden wir natürlich auch im Spektrum nach der FFT. Allerdings ist die Frequenz von f = 1 kHz etwas schwierig abzulesen, da wir einen größeren Frequenzbereich von 0...60 kHz eingestellt haben.

Zu C): Bei der **dB – Darstellung** gilt erneut ein **Bezugswert von 1V**. Ein Spitzenwert von 50 mV ist deshalb um den **Faktor 20 kleiner** – also muss das Diagramm die zugehörige Amplitude von **-26 dBV** zeigen

Zu D) Die Sache mit dem im Rhythmus des LO-Signals umgepolten RF-Signals kennen wir nun schon.

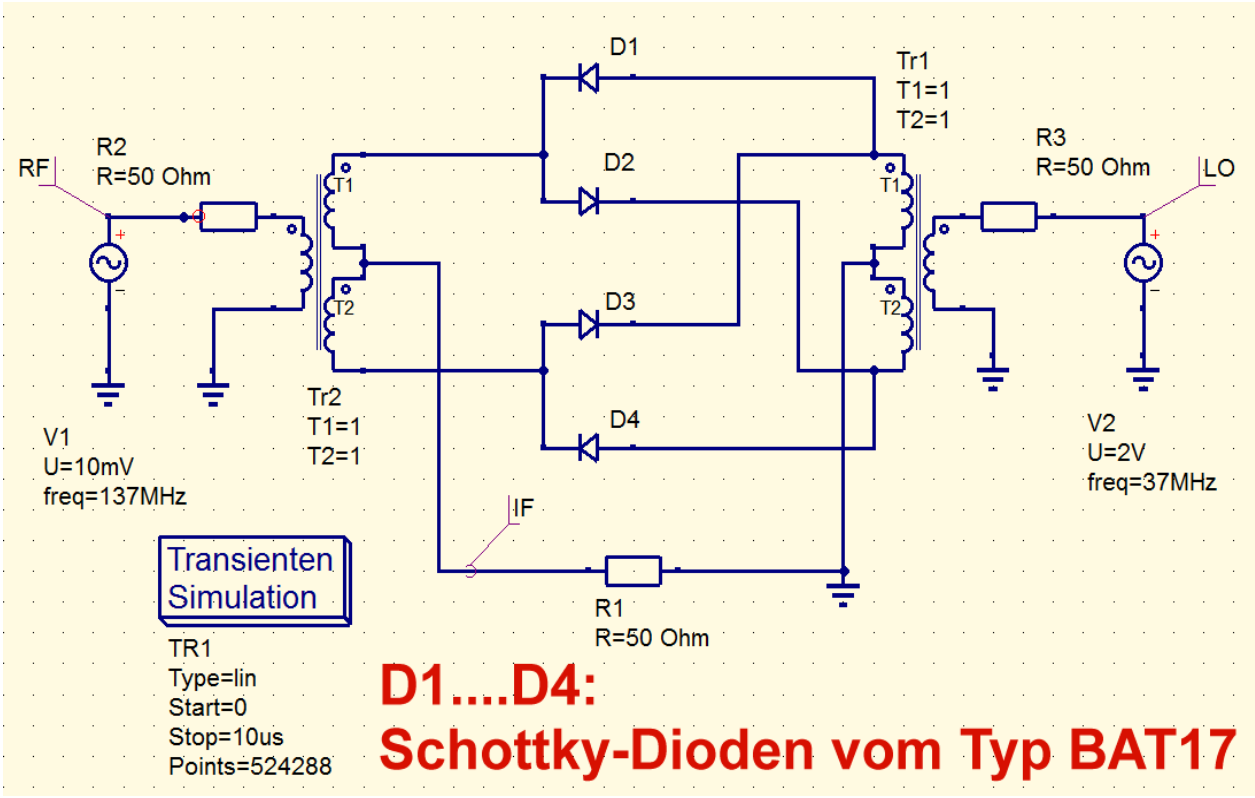
Zu E): Der Spitzenwert der beiden „**Seitenfrequenzen** neben der Grundwellen-LO-Frequenz“ (= 9 kHz und 11 kHz) beträgt **32 mV**. **Sowohl RF-Signal wie auch LO-Signal werden unterdrückt und erscheinen nicht am IF-Ausgang!**
Und bei jeder ungeraden Oberwelle finden wir wieder ein Seitenfrequenz-Pärchen..

Zu F) Der **Pegelunterschied** zwischen dem RF-Signal am Eingang in Bild C und den beiden ersten Seitenfrequenzen in Bild F beträgt damit **-26dB - (-30dB) = 4 dB**

und das ist die gesuchte **Umsetzdämpfung.**

14.3. Simulation eines echten Double Balanced Mixers (= Ringmodulator) aus einem 137MHz-Wettersatelliten-Empfangskonverter

NOAA-Wettersatellitenbilder können auf der Frequenz f_{RF} **= 137 MHz** als FM-Signal empfangen werden. Dabei wird in einem solchen Konverter erst das von der Antenne kommende Signal verstärkt und gefiltert, anschließend erfolgt eine Frequenzumsetzung auf f_{IF} **= 100 MHz** in einem DBM. Er ist als „Ringmodulator“ aufgebaut und ihm wird ein LO-Signal mit f_{LO} **= 37 MHz** zugeführt. Bei diesem LO-Signal wird die Rechteckform durch einen sinusförmigen Verlauf mit großer Amplitude ersetzt und deshalb ist die Wirkung auf die als Schalter dienenden Schottky-Dioden mit ihrer niedrigen Schwellspannung von ca. 0,3…0,4 Volt praktisch dieselbe.
Auf diesen DBM folgt ein schmalbandiger und sehr steilflankiger Bandpass mit f_0 **= 100 MHz**, an den z. B. ein UKW-Autoradio oder ein SDR zur Auswertung angeschlossen werden können.
So sieht die qucsstudio-Schaltung aus:



Am **RF-Eingang** wird der Mixer mit dem empfangenen und verstärkten Antennensignal (Der Spitzenwert von **U = 10 mV / f = 137 MHz ergibt Uhin = Uo /2 = 5 mV**) gespeist.

Die beiden **Übertrager** finden sich unter „**Komponenten / konzentrierte Komponenten / 3-Spulen-Trafo“**. **Die Übersetzungsverhältnisse sind auf „1“ eingestellt.**

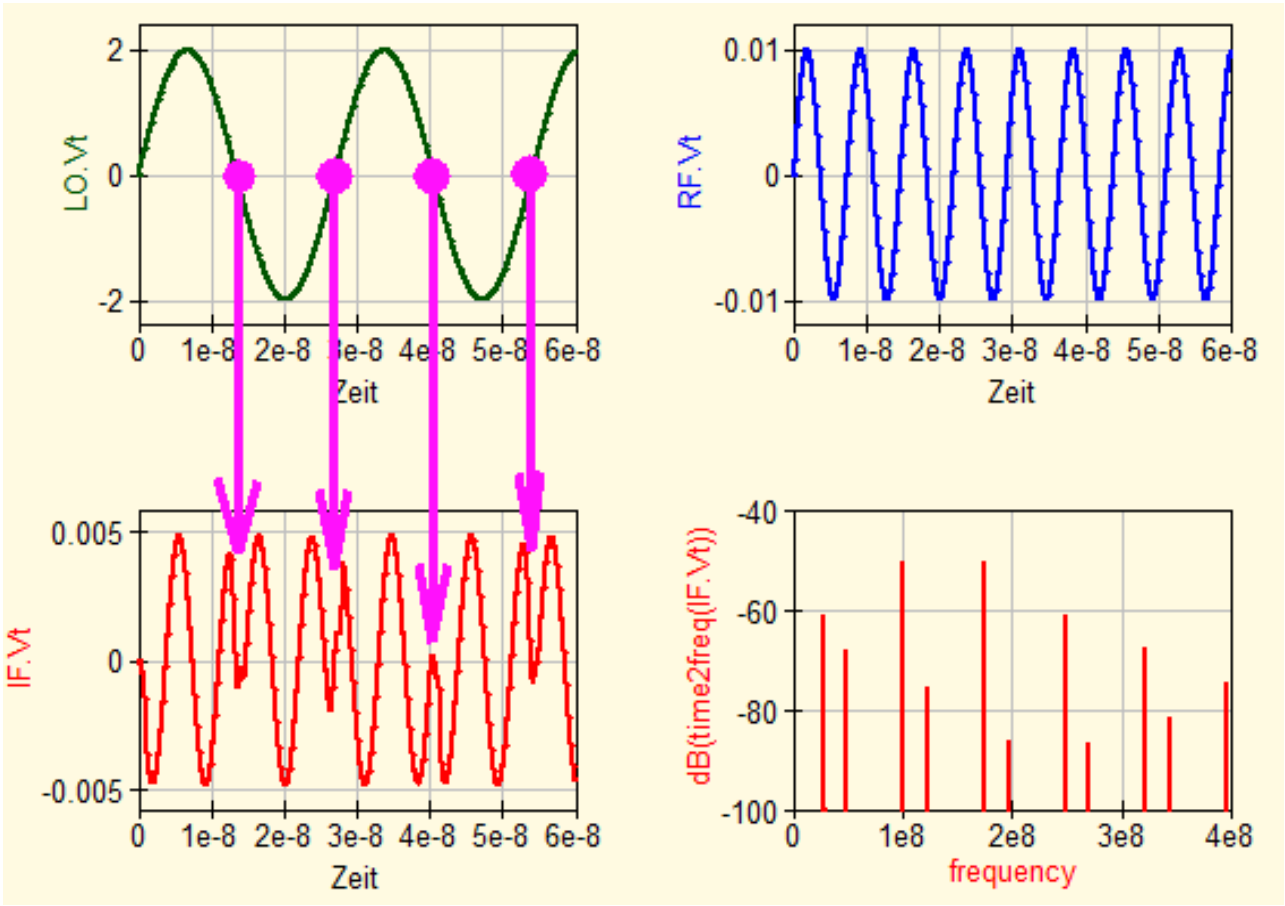
Als Schaltdioden dienen Schottkydioden vom Typ **BAT17**, dessen Spice-Modell wir bereits in Kapitel 4.3. aus dem Internet importiert und anschließend in die qucs-Form (**BAT17.cir**) konvertiert haben.

Das **LO-Signal** ist eine Sinusspannung mit einem **Spitzenwert von 2V und einer Frequenz von 37 MHz.**

Simuliert wird von **Null bis 10 Mikrosekunden mit 524 288 Punkten**. Damit erreichen wir eine ausreichende Anzahl an Samples für die FFT zur Darstellung des Frequenzspektrums. Die Auflösung bei der Frequenzachse beträgt damit

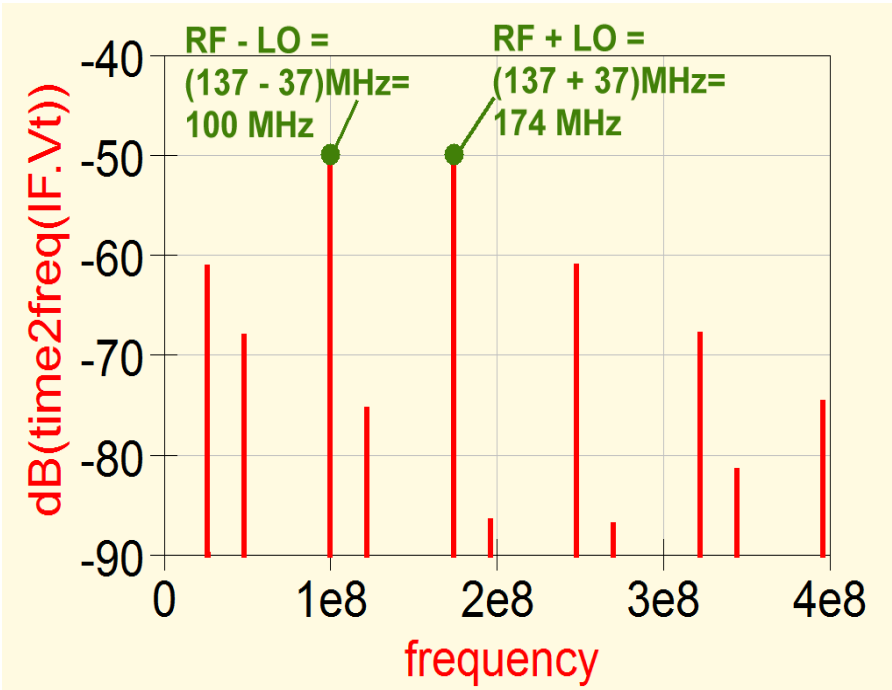
1 / 10µs = 100 kHz

Das ist das Simulationsergebnis:



Am IF-Ausgang erkennt man sehr schön die „Umpol-Zeitpunkte durch das LO-Signal“, bei denen das zugeführte RF-Signal seine Phasenlage umkehrt (...sie sind violett in beiden Diagrammen markiert).

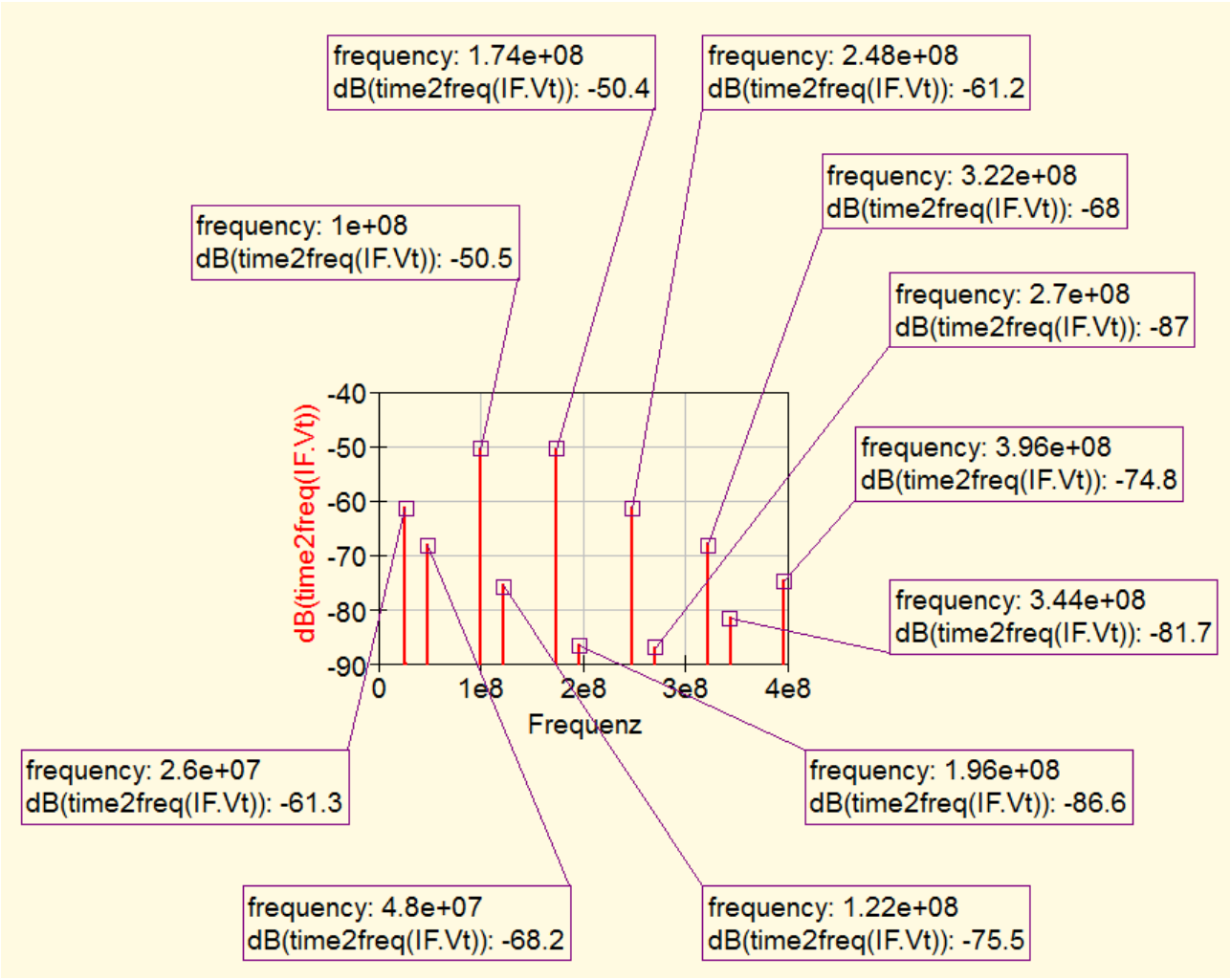
Das Frequenzspektrum des IF-Signals wirkt etwas überraschend und wir werden es nun genauer analysieren. Dargestellt wird der Frequenzbereich von 0....400 MHz.



Zuerst holen wir uns die gewünschten **beiden Seitenfrequenzen, also die Summen- und Differenzfrequenz von RF- und LO- Signal heraus.**

Zu den restlichen Spektrallinien gibt es eine hübsche Übung:

Wenn man an jede Spektrallinie einen eigenen Marker setzt, dann zeigt uns das Programm darin nicht nur die Frequenz, sondern auch den Pegel in dBV (= bezogen auf 1 Volt).



Jetzt kommt die Übungsaufgabe:

Es gilt bei den anderen Spektralllinien des Diagramms:

Die Grundwelle oder die Oberwellen des RF-Signals wurden mit der Grundwelle oder den Oberwellen des LO-Signals multipliziert.

Das Ergebnis ist immer eine zugehörige Summen- und Differenzfrequenz als „Pärchen“.

Beispiel:

RF+3xLO = 137 MHz +3x(37 MHz) = 137 MHz + 111 MHz = 248 MHz
RF - 3xLO= 137 MHz - 3x(37 MHz) = 137 MHz – 111 MHz = 26 MHz

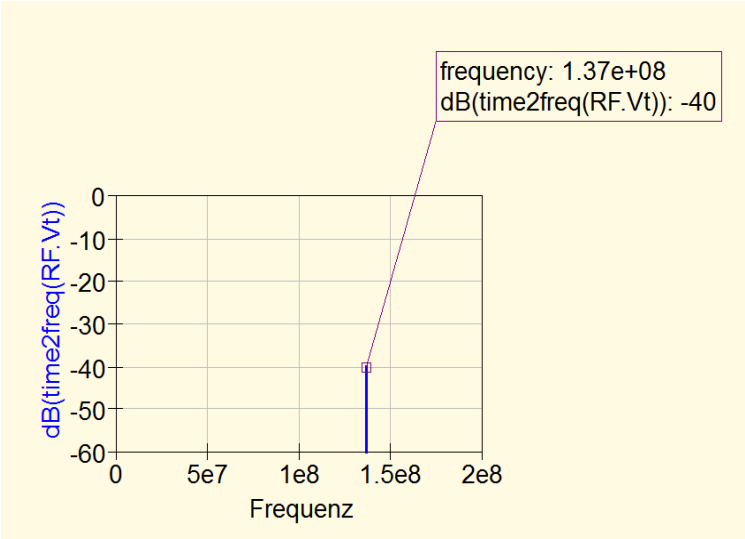
Dazu ein Tipp: **Zwei zusammen gehörende Seitenfrequenzen weisen im Spektrum immer dieselbe Amplitude auf**. Das zeigt dieses Beispiel. Bitte prüfen!

Aufgabe:
Ermitteln Sie nun selbst die Zusammenhänge zwischen den übrigen Spektrallinien!

Noch eine Aufgabe:
Bestimmen Sie die genaue Umsetzdämpfung, die das RF-Signal im Mixer erleidet.

Lösung:
Die genauen Pegel von Summen- und Differenzfrequenz am IF-Ausgang können wir aus dem vorigen Diagramm zu **-50,5 dBV** ablesen.

Aber zur **Messung des zugeführten RF-Signals** benützen wir die FFT und bestimmen auf diese Weise seinen exakten Pegel in dBV.



Vorsicht:
Der Pegel von -40 dBV bezieht sich auf die „Urspannung“. Die „Hinlaufende Welle“, die in den Mischer eintritt, ist um 6 dB kleiner (denn Uhin = Uo /2).

Damit erhält man die Umsetzungsdämpfung zu

-40 dBV – 6 dB - (-50,5 dB)

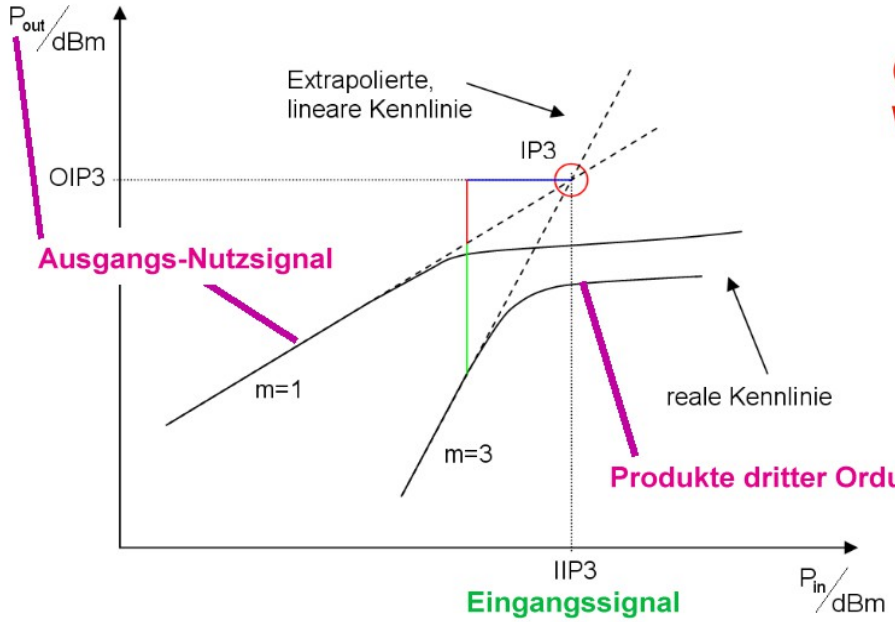
= **4,5 dB**

14.4. Der IP3-Punkt - sein Zweck und seine Ermittlung

Die Simulation des „***Third Order Intercept Points***“ liefert die erforderlichen Informationen über das **Verzerrungsverhalten** irgend einer Stufe bei steigendem Eingangspegel.

Beim IP3 wären (…wenn das ginge, aber vorher laufen wir schon in die Begrenzung…) die Verzerrungen dritter Ordnung genau so groß wie das zugehörige Nutzsignal. Und da sie echte Nachbarn des umgesetzten Nutzsignals sind, stören sie sehr…..

Sehr gut ist das an diesem Diagramm zu sehen:



Quelle: Wikipedia

Kennt man nun diesen Punkt genau, dann lässt sich für jeden Nutz-Eingangs- oder Ausgangspegel der Abstand zu solchen störenden Intermodulations - Signale dritter Ordnung leicht ausrechnen, denn das sind (wie man im Bild sieht!) lediglich Geradengleichungen, die man ohne Mühe bewältigt…

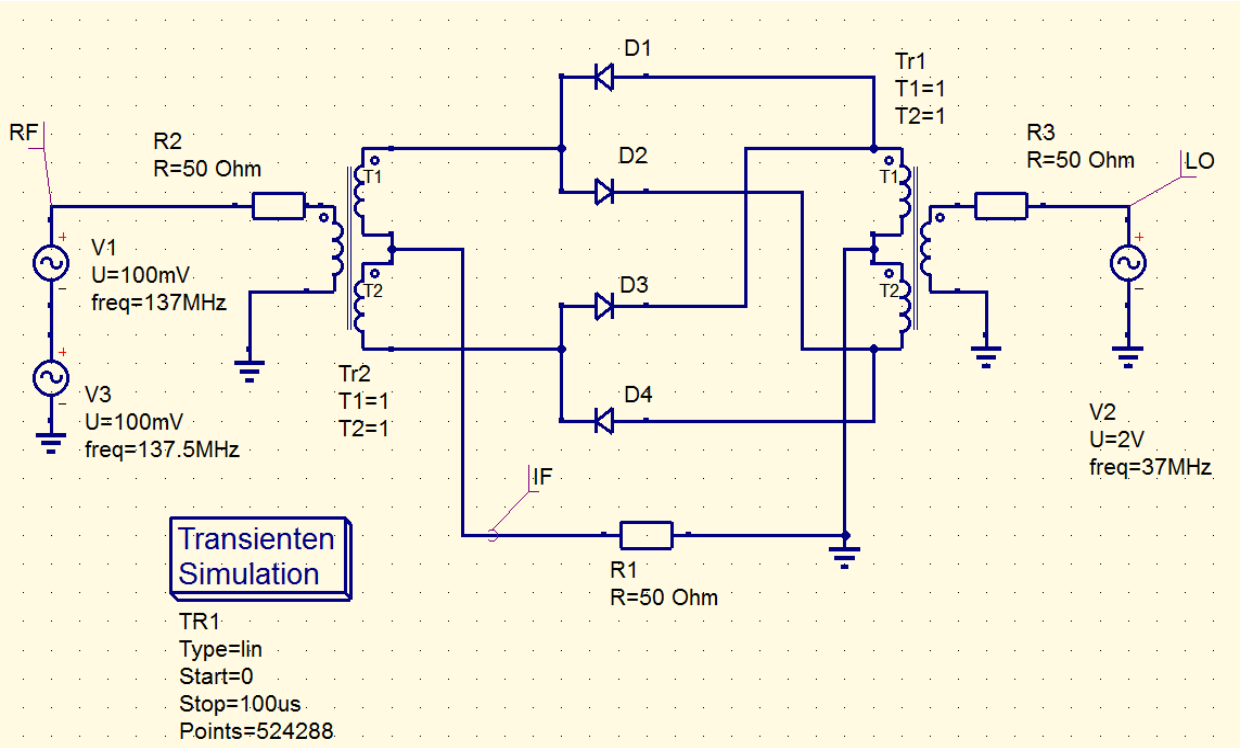
Außerdem erhält man (wenn man das Eigenrauschen der Stufe als maximale Obergrenze dieser Verzerrungen definiert) mit dieser Geradengleichung den „Intermodulationsfreien Dynamikbereich“

Zur Bestimmung von IP3 steuern wir den Baustein mit zwei „In Band-Signalen“ an, die nur einen geringen Frequenzunterschied aufweisen und in der Mitte des vorgesehenen IF-Durchlassbereiches liegen.

Der zugeführte RF-Pegel wird dabei so hoch gewählt , dass damit bereits ein deutlicher Intermodulationseffekt zu erwarten ist.

Noch ein Hinweis: Die Intermodulationsprodukte **zweiter** Ordnung sind wesentlich stärker, aber sie stören meist nicht, da sie weit weg von der gewünschten Seitenfrequenz liegen und deshalb leicht durch Filterung am Ausgang beseitigt werden können.
Die oben erwähnten Produkte dritter Ordnung liegen dagegen ganz dicht bei den gewünschten Seitenfrequenzen. Und wenn man nun zwei nahe beieinander liegende Eingangs-Signale an den Mischer anlegt, dann tauchen plötzlich solche Störlinien genau im Frequenzabstand der beiden Eingangs-Spannungen NEBEN diesen gewünschten und umgesetzten Nutzsignalen auf. Und da hilft dann kein Filter mehr....!
Wir werden das gleich auf der nächsten Seite sehen.

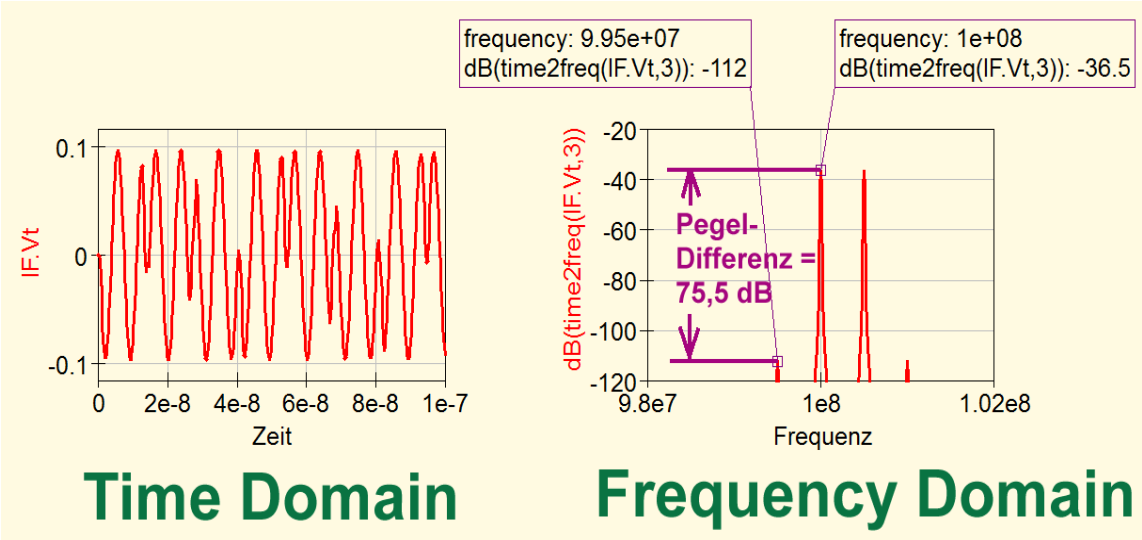
Deshalb wollen wir uns noch das Intermodulationsverhalten unserer Mixer-Schaltung in Form der IP3-Simulation ansehen. Dazu steuern wir den Mixer am RF-Eingang mit zwei „Inband-Signalen“, **nämlich 137 MHz und 137,5 MHz** an, deren „Hinlaufende Wellen“ (Uhin = Uo / 2 mit je 50 mV Spitzenwert) gleich groß sind. Die Urspannungen betragen natürlich 100 mV.



Wir simulieren 100 Mikrosekunden lang (….das ergibt eine Frequenzauflösung von 1 / 100µs = 10 kHz) und wählen die maximal mögliche Punktezahl (524 288).

Dann sehen wir uns das IF-Signal an. Im linken Diagramm haben wir den zeitlichen Verlauf, im rechten Diagramm ist das Spektrum von 98 bis 102 MHz abgebildet. **Dabei wurde mit einem Hanning-Fenster (Index = 3) gearbeitet, um die erforderliche hohe Amplituden-Auflösung zu ermöglichen.** Sehr schön sind darin die beiden Seitenfrequenzen mit 100 MHz und 100,5 MHz zu erkennen – aber auch die entstandenen Intermodulationsprodukte dritter Ordnung tauchen links und rechts davon auf. Für die IF mit 100 MHz beträgt die **Pegeldifferenz** zum ersten „Störprodukt“ (bei f = 99,5 MHz):

-36,5 dB - (-112)dB = 75,5 dB



Zusammen mit dem zugeführten Eingangspegel kann man nun den IP3-Punkt nach folgender Beziehung bestimmen:

IP3 = Eingangs-Nutzpegel + 0,5 x (Pegeldifferenz)

Wir müssen da aber auf mehrere Dinge beim Eingangs-Nutzpegel achten und haben etwas Arbeit von uns:

a) Es muss die „**Hinlaufende Welle**“ (= Uo / 2) verwendet werden. Sie beträgt **50 mV** bei einem **Spitzenwert der Urspannung** von je 100 mV für beide verwendete RF-Signale.

b) Pegel werden üblicherweise mit **Effektivwerten** berechnet, da sie **Leistungsangaben** darstellen, aber unser Programm arbeitet mit Spitzenwerten und da müssen wir umrechnen.
Der Effektivwert ist beim Sinus um den Faktor 0,7071 niedriger als der Spitzenwert und das entspricht genau einer nötigen Reduktion von 3 dB. Also hat die „Hinlaufende Welle“ nur einen Effektivwert von
0,7071 x 50 mV = **35,35 mV.**

c) IP3- Werte werden immer mit der Einheit „dBm“ angegeben.
Hier wird stets mit einer Bezugsleistung von 1 Milliwatt am Systemwiderstand gearbeitet!

Nun rechnen wir erst mal die zu 35,35 mV (an Z = 50Ω) gehörende Leistung der Hinlaufenden Welle aus:
P = U x U / 50Ω = (35,35mV) x (35,35mV) / 50Ω = 0,025 mW

Dann bestimmen wir ihren Pegel in dBm:
dBm-Pegel = 10 x log (0,025mW / 1mW) = -16 dBm

Jetzt haben wir es geschafft:

IP3 = -16dBm + 0,5 x (75,5dB) = -3,47dBm + 29,85dB = +21,7dBm

15. Entwicklung eines schmalen HF-Bandpasses für 10,7 MHz (= Narrow Bandpass = Coupled Resonator Type)

15.1. Vorbemerkungen

Benötigt man im HF-Bereich Bandpass-Filter mit sehr geringer Bandbreite (Richtwert: 1...3 % der Mittenfrequenz), so erreicht man dieses Ziel nur durch besondere Schaltungen, die mit „Transformationen“ arbeiten. Die zugehörige Theorie ist etwas anspruchsvoll und in den teuren Mirowave-CAD-Programmen sind die entsprechenden Filtertools natürlich enthalten.
Aber bereits seit DOS-Zeiten gibt es kostenlose Filtercalculatoren, die alle Rechenarbeit übernehmen und nur mit den gewünschten Daten gefüttert werden müssen. Allerdings laufen solche DOS-Programme unter den modernen Windows-Versionen nicht mehr und das ist schade. Denn da gab (und gibt!) es für die HF-Technik unzählige sehr leistungsfähige Werkzeuge, z. B. zur Berechnung von allen möglichen Antennen, Parabolspiegeln, Hohlleitern, Filtern, Anpass-Netzwerken usw. usw. Aber für unser Thema (Narrow Bandpass Design) findet man unter den heutigen Windows-Versionen nichts Kostenloses. dafür existiert aber noch ein hervorragendes DOS-Programm mit dem Namen „fds.exe“ (= filter design software...“).

Achtung:
Der Filter-Calculator in qucsstudio beherrscht erst ab Programmversion 2.4.0. auch die Berechnung des „Coupled Resonator Bandpasses“. Deshalb erfolgt im letzten Kapitel 14.9. eine Vergleichssimulation zum jetzt folgenden Entwurf!

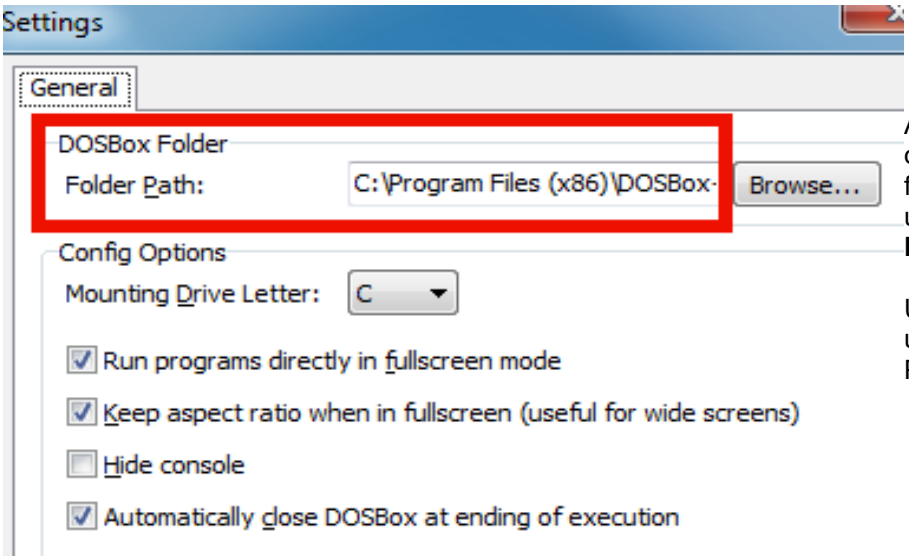
Zuerst wird jedoch eine Lösung vorgestellt, die sich beim Autor sehr bewährt har und die Nutzung aller interessanten DOS-Fundsachen ermöglicht:

Man installiert unter Windows eine „DOS-Box“, die ein komplettes DOS-Betriebssystem simuliert. Kombiniert mit der passenden guten Bediener-Oberfläche (= „DOS Shell“) reicht plötzlich ein Mausklick, um mit einem solchen DOS-Produkt zu arbeiten und hinterher wieder mit einem Mausklick zu Windows zurückzukehren.

15.2. Installation von DOS-Box, DOS-Shell und fds.exe

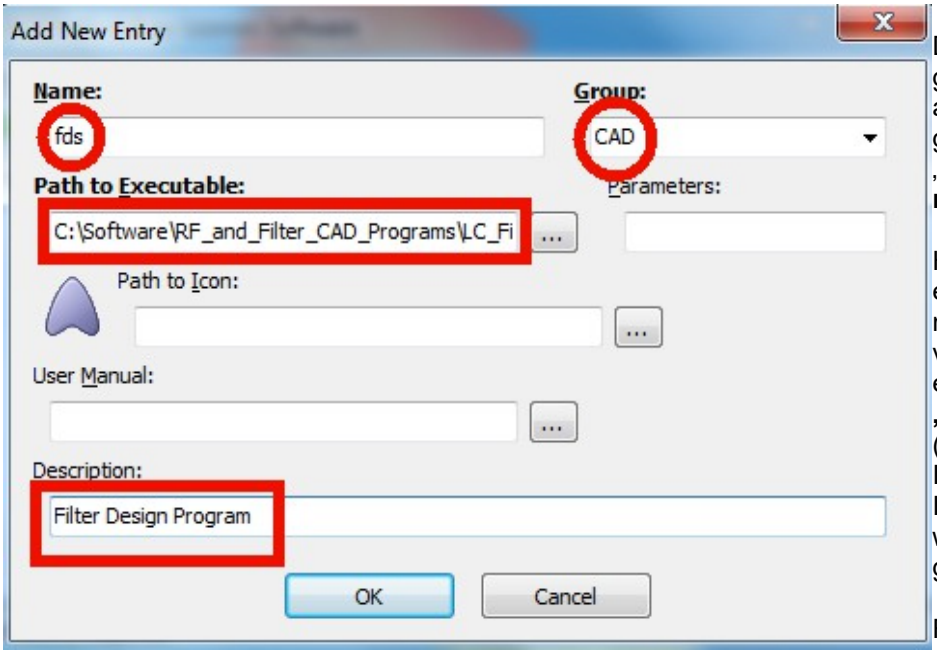
Alle drei Programme warten in meiner Homepage beim Thema „qucsstudio-Tutorial“ im Anschluss an das Tutorial auf Ihren Download.

Begonnen wird mit der Installation der DOS-Box, gefolgt von der DOS-Shell. Das Filterprogramm wird in einem eigenen Ordner (z. B. unter „Programme \ Filter“) untergebracht.



Auf den Bildschirm wurden bei der Installation automatisch Icons für die Box und die Shell angelegt und da starten wir erst **nur die DOS-Shell.**

Unter „**Edit**“ finden wir „**Settings**“ und da müssen wir genau den Pfad zur DOS-Box einstellen.



Die DOS-Shell wird nun geschlossen und anschließend nochmals geöffnet. Wieder brauchen wir „**Edit**“, aber diesmal „**Add new entry**“

Hier sehen wir die erforderlichen Einträge sowie meinenpersönlichen verwendeten Pfad zum erfolgreichen Zugriff auf **„fds.exe“**.
(Wir brauchen ab jetzt nur den Icon für die DOS-Shell, der Icon für die DOS-Box kann wieder vom Bildschirm gelöscht werden).

Fertig!

15.3. Pflichtenheft für den 10,7 MHz – Bandpass

Die Frequenz f = 10,7 MHz wird meist bei FM-Empfängern als Zwischenfrequenz benützt. Es gibt dafür viele fertige Bauteile zu kaufen (Keramikfilter, Quarzfilter usw.) und für den Einsatz moderner Techniken wie die Digitale Signalverarbeitung ist dieser Wert längst nicht mehr zu hoch. Bei eigenen Entwicklungen und Experimenten in dieser Richtung bietet sich folglich dieser Bereich an, aber man sollte neben den nötigen Signalquellen auch passende Filter bereithalten. Sehr oft wird es nämlich nötig sein, unerwünschte Signale auch weit außerhalb des vorgesehenen Arbeitsbereiches um 10,7 MHz massiv abzuschwächen.

Den Anfang bildet immer eine Reihe von Überlegungen. Sie münden in einer Liste der nötigen Anforderungen, die dann zum Konzept und zum Pflichtenheft führen. Und so hört sich das dann an:

Systemwiderstand:	**Z = 50 Ω.**
Filtertyp:	**Tschebyschef mit einem Ripple von 0,3 dB**
Mittenfrequenz:	**10,7 MHz**
3 dB - Bandbreite:	**500 kHz**
Durchgangsdämpfung:	**nicht wesentlich über 6 dB**
Flankensteilheit:	**Sperrdämpfung bei 9 und 12 MHz wenigstens 70 dB**
Weitab-Selektion:	**angestrebt werden 70 dB Sperrdämpfung bis 500 MHz**

Der Baustein soll in ein gefrästes Alugehäuse mit aufgeschraubtem Abschirmdeckel eingebaut werden. Die zugehörige Platine weist die Abmessungen von 30 mm x 50 mm auf, zur Befestigung der Platine im Gehäuse dienen vier Schrauben (M2,5) im Abstand von 3 mm vom Platinenrand.
SMA-Buchsen bilden den Eingang und den Ausgang, wobei ihr Innenleiter direkt auf passende Microstrip-Leitungen der Platine gelötet wird. Als Platinenwerkstoff dient das bekannte Rogers-Material „RO4003“ mit einer Dicke von 32 mil = 0,813 mm und beidseitiger Kupferkaschierung (Dicke = 35 µm). Die Cu-Unterseite bildet die durchgehende Massefläche, nötige Durchkontaktierungen von der Oberseite zur Massefläche (= „Vias“) nehmen versilberte Hohlnieten mit einem Durchmesser von 0,8 mm vor.

15.4. Entwicklungs-Strategie

Folgende Fragen bedürfen zunächst einer Klärung:

a) Welcher Bandfiltertyp erfüllt diese Ansprüche?

b) Welcher Filtergrad ist erforderlich?

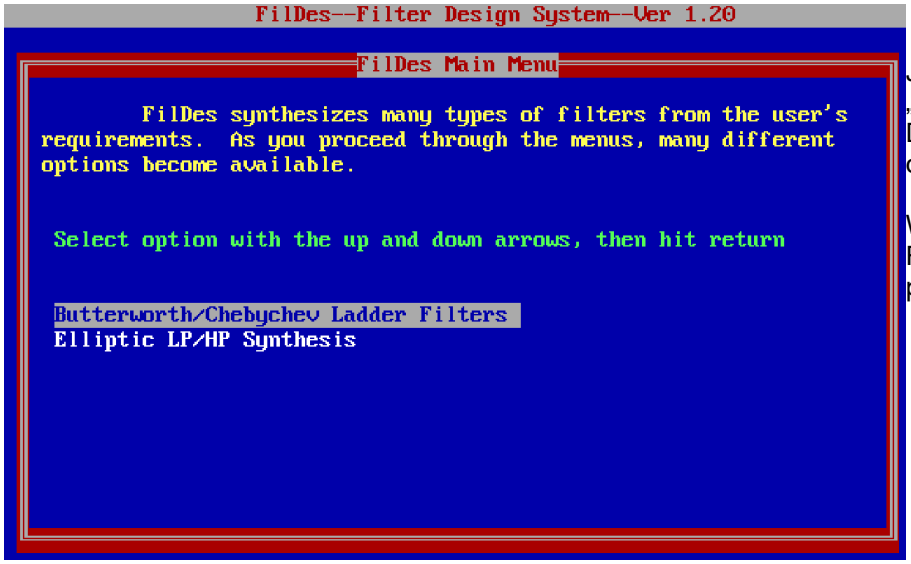
c) Welche Spulengüte ist bei der geforderten maximalen Durchgangsdämpfung mindestens nötig?

d) Welche Spulenkonstruktion erfüllt diese Anforderung UND passt gleichzeitig noch in das Gehäuse?

e) Welche Maßnahmen sind eventuell noch vorzusehen, um die hohe und breitbandige Sperrdämpfung zu erzielen?

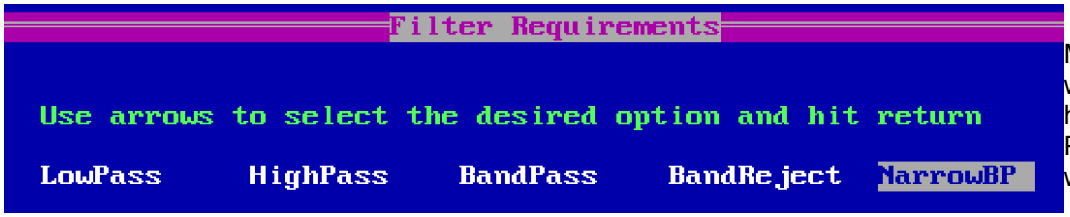
Von früheren Projekten her war bekannt, dass sich der „Coupled Resonator Bandpass Filter Typ“ = „Narrow Bandpass“ hervorragend für diese Aufgabe eignet. In ihm weisen alle Induktivitäten denselben Wert auf und die Anpassung an den Quell- und Lastwiderstand erfolgt durch eine kapazitive Transformation. Deshalb wird in einer ersten Simulation untersucht, welcher Filtergrad zur Erzielung der geforderten Bandbreite und Sperrdämpfung gewählt werden muss. Gleichzeitig wird geprüft, welche Mindest-Spulengüten für die geforderte maximale Durchgangsdämpfung erforderlich sind. Die Umsetzung dieser Güte-Anforderung in eine passende Spulenkonstruktion wird im nächsten Kapitel behandelt. Und schließlich geht es an die Prototyp-Erstellung, wobei zunächst ein niedriger Filtergrad (beispielsweise erfordert ein Filtergrad von n = 3 nur drei Spulen -- das gibt weniger Arbeit beim Tüfteln und Bestücken…) zum Herantasten an die optimale und nachbausichere Ausführung gewählt wird. Erst wenn alle Probleme zur Zufriedenheit gelöst sind, werden die Vollversion und die Optimierung der Sperrdämpfung angepackt.

15.5. Erforderliche Filterschaltung und Spulengüte



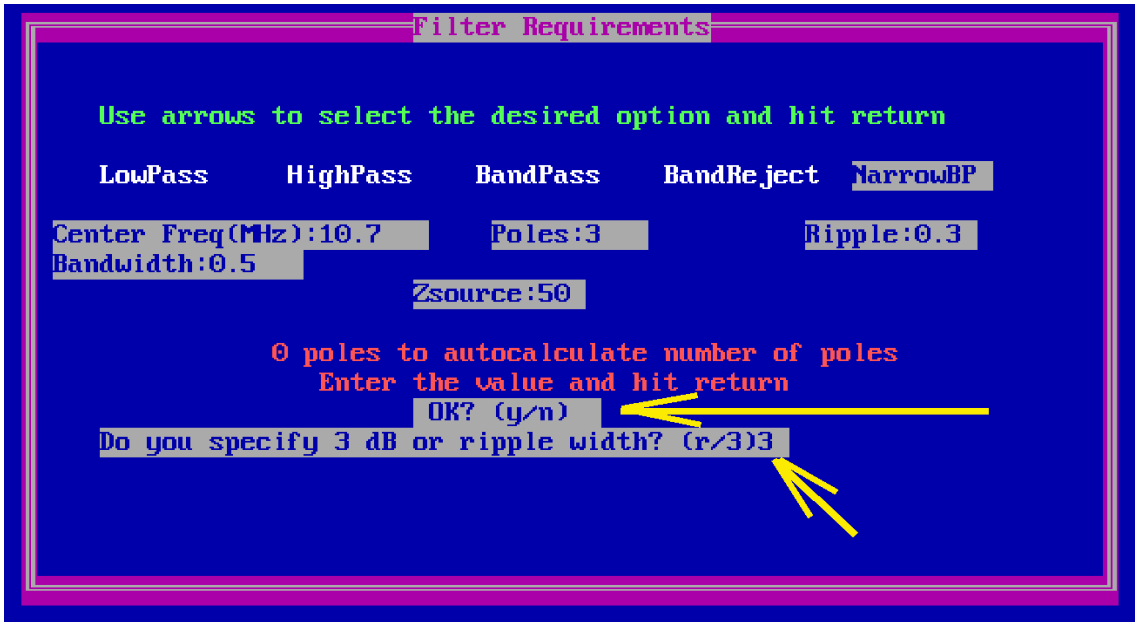
Jetzt folgt der Praxistest für „fds.exe“:
Die DOS-Shell wird gestartet und darin „fds“ aufgerufen.

Wir brauchen das „Chebychev“-Filter und markieren die passende Zeile.



Mit OK geht es weiter und mit den horizontalen Pfeiltasten wählen wir „NarrowBP“.

Dann wird es interessanter:

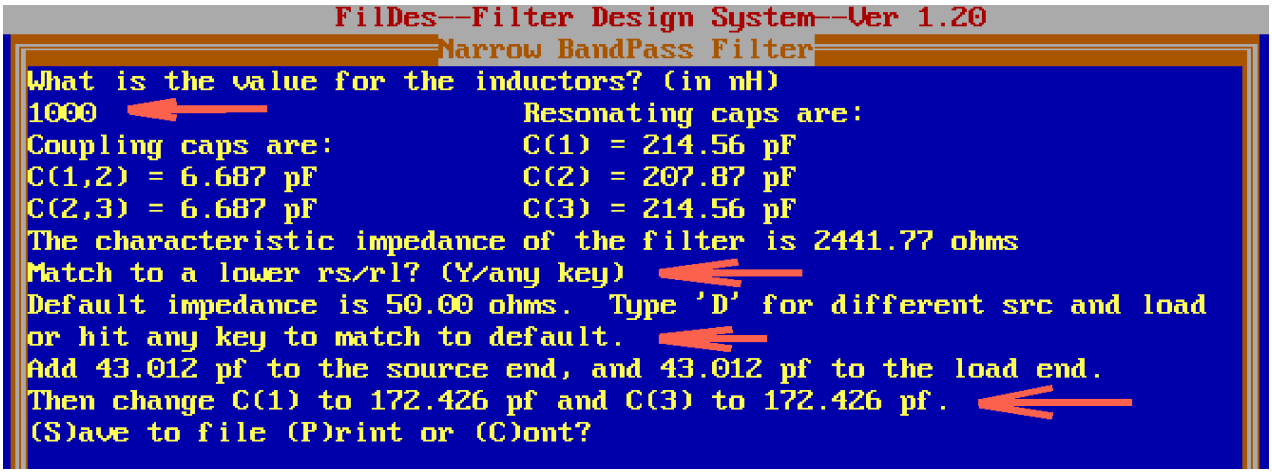


Der Reihe nach geben wir ein und bestätigen jeweils mit der „ENTER“-Taste:

Center Frequency	**= 10.7**
Poles	**= 3**
Ripple	**= 0.3**
3 dB or ripple width	**= 3**
Bandwidth	**= 0.5**
Zsource	**= 50**
OK?	**= y**

Dann folgen im nächsten Schirmbild nacheinander folgende Schritte:

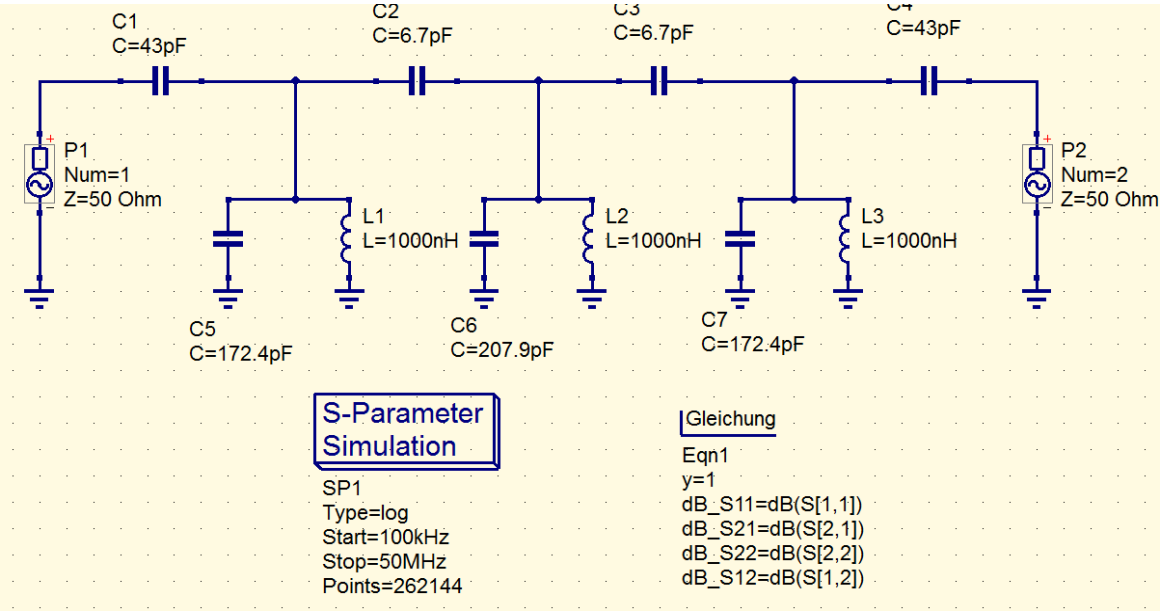
L		**= 1000 nH**
Match to a lower rs/rl		**= y**
Hit any key for default	**of 50 Ω**	**= beliebige Taste drücken für Z = 50Ω**

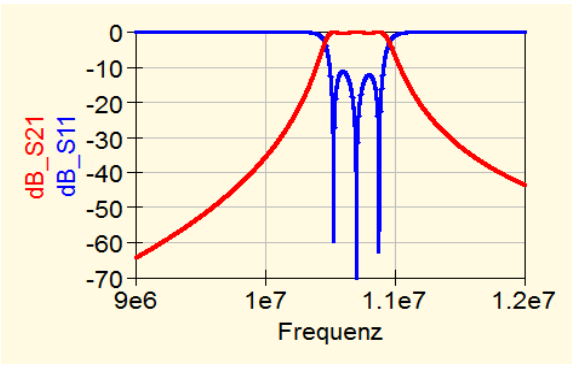


Zur Induktivität „L“ noch einige Worte, denn die muss der Entwickler angeben! Sie hat in allen Kreisen denselben Wert.
Da wählt man einen praktischen Erfahrungswert, bei dem die Umsetzung in die Wirklichkeit auch funktioniert.

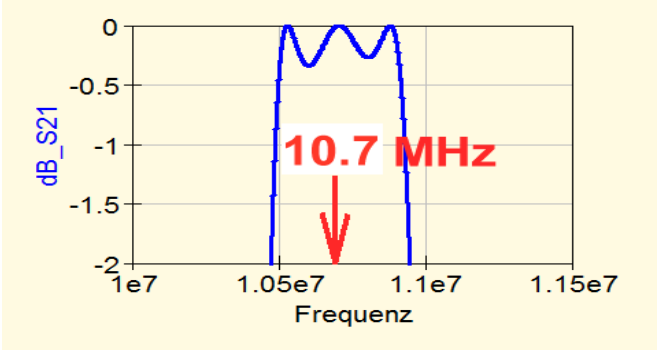
Erfahrungsgemäß ist das eine Induktivität, deren Blindwiderstand bei der Mittenfrequenz irgendwo zwischen 50 und 100 Ω liegt UND die sich problemlos als Normwert beschaffen lässt.

Gewählt wurde 1 µH = 1000 nH mit einem Blindwiderstand von ca. 63 Ω bei 10,7 MHz. Sie erfüllt diese gestellten Forderungen und ist vom Wert her leicht beschaffbar.
Weiterhin sagt uns das Programm:
Auf jeder Seite der Schaltung müssen wir einen Serienkondensator mit 43,012 pF anschließen und den folgenden Schwingkreis-Kondensator auf 172,46 pF reduzieren.
Da haben wir nun alles, was wir brauchen und machen eine hübsche qucsstudio-Simulation daraus. Allerdings sollte man vorher die vielen Nachkommastellen der Bauteilwerte praxisgerecht auf eine Stelle nach dem Komma runden.





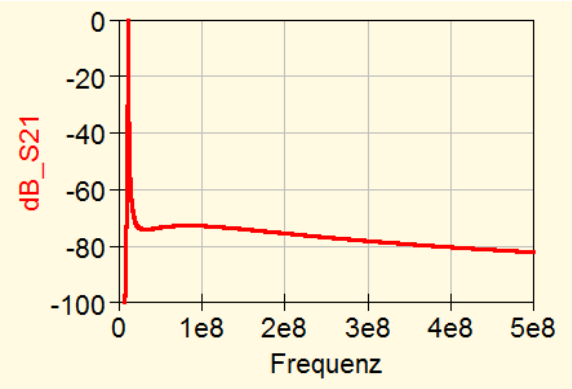
Simuliert wird von 100 kHz bis 50 MHz, aber beim Ergebnis schauen wir uns erst mal S11 und S21 im Bereich von 9....12 MHz an:



Das sieht ja schon ganz gut aus und wir werfen mal einen Blick auf die „Tschebyschef-Wellen“ von S21 im Durchlass-Bereich:

Abgesehen von einer kleinen „Schieflage“ der Tschebyschef-Wellen durch das großzügige Auf- und Abrunden der Bauteilwerte können wir mit dem fds-Programm und seinem Ergebnis sehr zufrieden sein.

Man sollte sich aber stets bei Filterschaltungen die Mühe machen, auch nach der „**Weitab-Selektion**“ zu schauen, denn da kann es manchmal böse Überraschungen geben.



So verläuft S21 bis 500 MHz und man sieht, dass in diesem Bereich die „Sperrdämpfung“ nur wenig abnimmt (von S21 = -70 bis auf S21 =-80 dB).

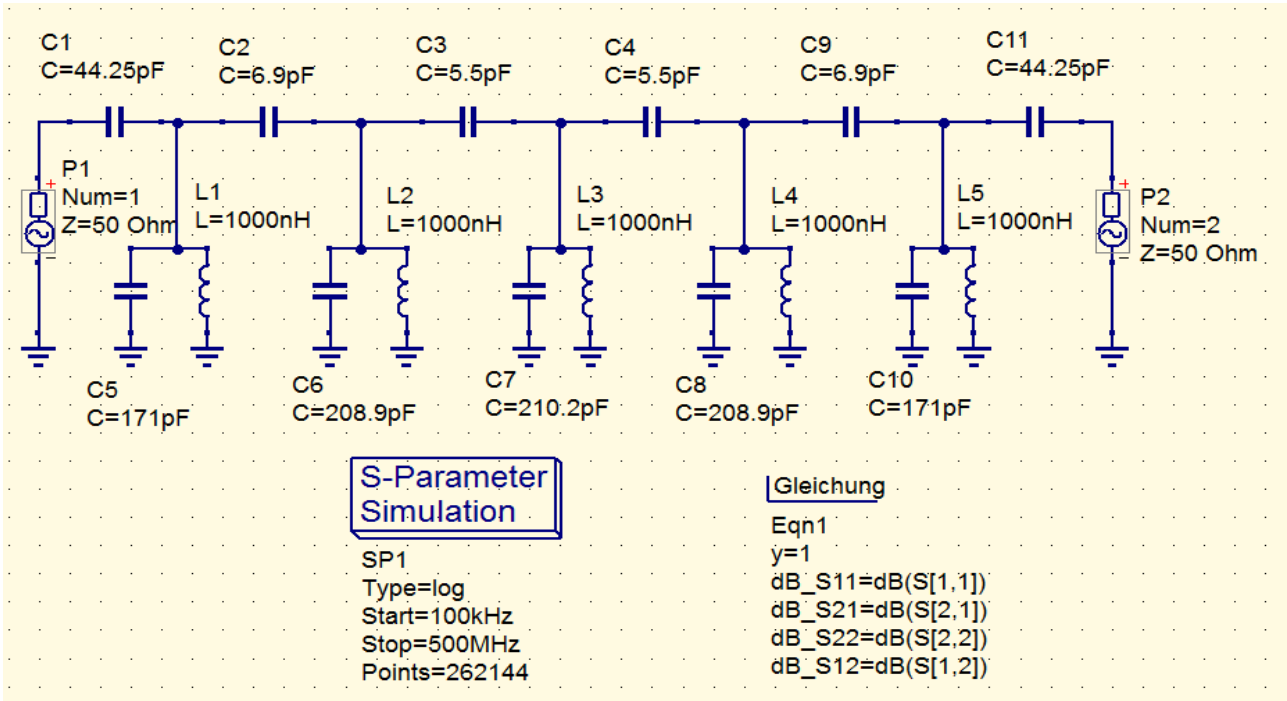
Der Grund ist recht einfach:
Der Bandpass wurde durch Transformation aus einem Tiefpass erzeugt. Dadurch wurde der „Nullpunkt verschoben und er liegt nun bei 10,7 MHz.
Deshalb findet man den Frequenzbereich des Tiefpasses „von Null bis Unendlich“ nun zwei Mal:

a) An 10,7 MHz „gespiegelt“ erstreckt er sich von 10,7 MHz herunter bis Null.
b) Vom Startpunkt 10,7 MHz aus sinkt S21 schnell bis auf ca. -70 dB. Der weitere Verlauf bis Unendlich ist aber nun natürlich stark gedehnt und deshalb sinkt die Dämpfung nur sehr zögernd mit weiter steigender Frequenz.

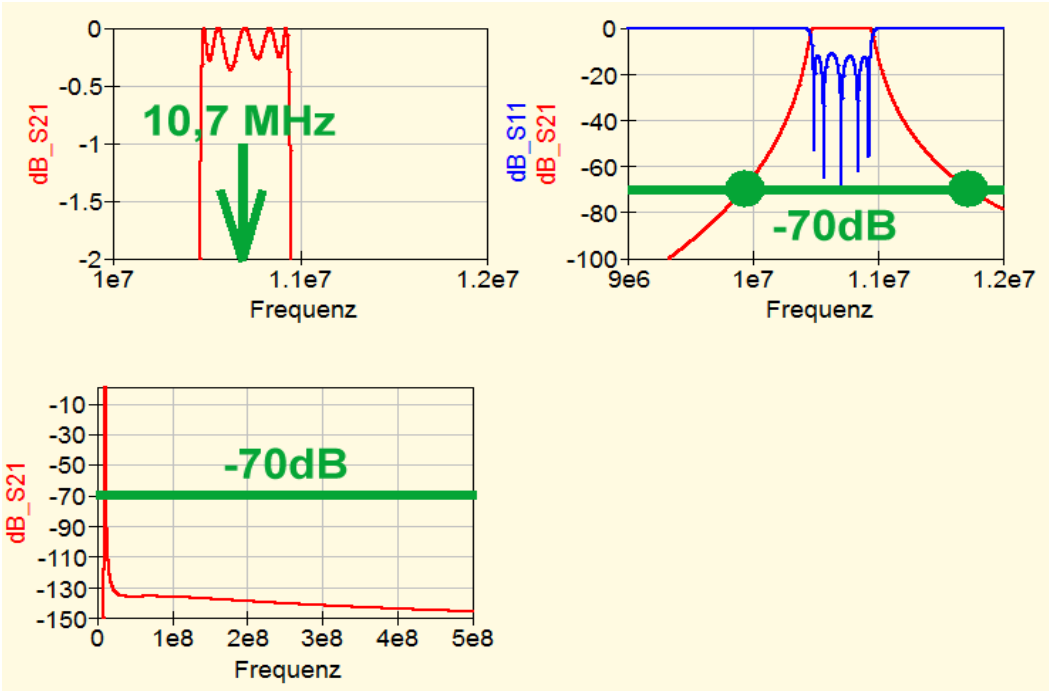
Zurück zur Schaltung:
Es ist gut zu sehen, dass mit dem Filtergrad N = 3 die **geforderten Sperrdämpfungen von 70dB** bei 9 MHz und 12 MHz (es sind nur 64 dB bzw. 45 dB) **nicht zu schaffen** sind. Also wiederholen wir den Entwurf und wählen nun 5 Pole (Filtergrad N = 5). Da muss „fds“ wieder ran und es sagt:

```
What is the value for the inductors? (in nH)
1000                          Resonating caps are:
Coupling caps are:            C(1) = 214.37 pF
C(1,2) = 6.877 pF             C(2) = 208.86 pF
C(2,3) = 5.508 pF             C(3) = 210.23 pF
C(3,4) = 5.508 pF             C(4) = 208.86 pF
C(4,5) = 6.877 pF             C(5) = 214.37 pF
The characteristic impedance of the filter is 2309.75 ohms
Match to a lower rs/rl? (Y/any key)
Default impedance is 50.00 ohms.  Type 'D' for different src and load
or hit any key to match to default.
Add 44.251 pf to the source end, and 44.251 pf to the load end.
Then change C(1) to 171.075 pf and C(5) to 171.075 pf.
```

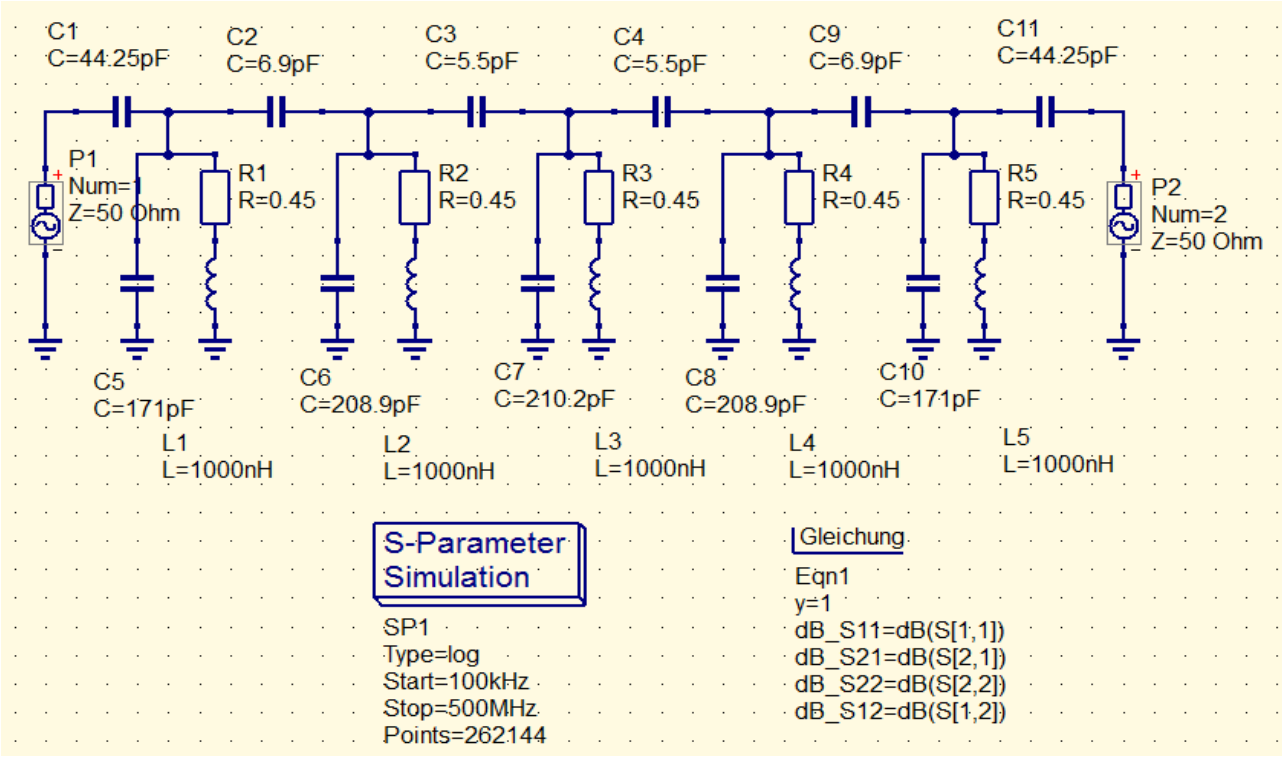
Dazu gehört folgender Schaltplan mit gerundeten Bauteilwerten:



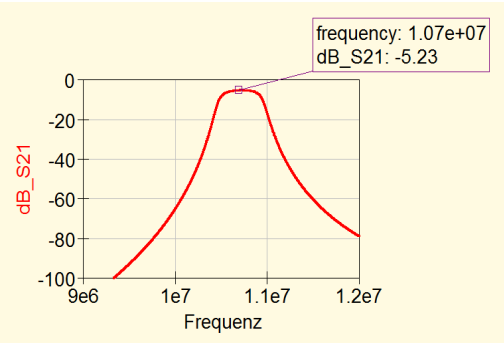
Und hier kommen die interessierenden Eigenschaften und Details:



Oowohl die **Nah-Selektion wie auch die Weitab-Selektion sind nun sehr zufriedenstellend** und erfüllen die Forderungen des Pflichtenheftes.
Das Pflichtenheft spricht aber auch von einer „**maximalen Einfügungsdämpfung des Filters von 6 dB im Durchlass-Bereich**“. Die hängt direkt mit den Verlusten der Bauteile - speziell mit der Güte der Spulen – zusammen und man sollte das ebenfalls austesten.
Realistisch ist im Kurzwellenbereich eine **Spulengüte von Q = 150**, die man nur mit Amidon-Ringkernen erreichen kann.
Für diese Güte Q = 150, einer Induktivität von 1000 nH und einer Betriebsfrequenz on 10,7 MHz ergibt sich ein Reihen-Verlustwiderstand von **$Rr = X_L / Q = 2 * \pi * 10{,}7MHz * 1\mu H / 150 = 0{,}45\Omega$**



Nach der Simulation ist die Bestimmung der gesuchten Einfügungsdämpfung über den Frequenzmarker eine Freude:



Sie beträgt **5,23 dB** und bleibt damit deutlich unter der geforderten 7 dB – Grenze.

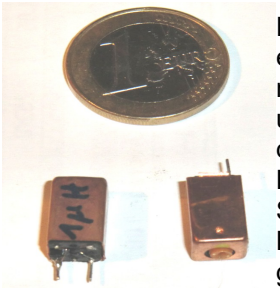
Ein kleiner Tipp zum Frequenzmarker:
Markiert man ihn und ruft seine Eigenschaften auf, dann kann man die Anzahl der angezeigten Nachkommastellen verändern.

Das ist äußerst praktisch, wenn man eine sehr feine Frequenzauflösung benötigt.

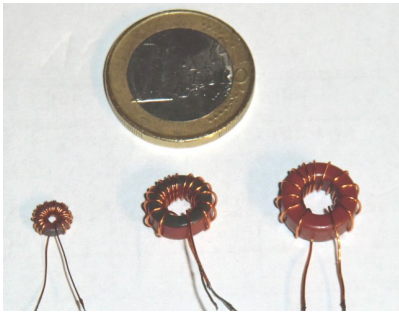
15.6. Der Kampf mit den Spulen

Es galt ja schon immer in der Technik der Satz: „Ohne Messen kein Wissen“. Aber die Ermittlung von hohen Spulengüten bei hohen Frequenzen (hier: bei 10 MHz, aber bis 100 MHz wäre auch noch interessant) erfordert teure und als Gebrauchtgeräte nur schwer zu findende Messeinrichtungen. Da ist es gut, wenn man schon viele Jahre auf den Elektronikflohmärkten manchem Kauf nicht widerstehen konnte und das zahlte sich nun aus. Im Fundus versteckte sich schon ewig ein „Boonton RX Meter“ von 1960 und da handelt es

sich um eine Impedanzmessbrücke für den Frequenzbereich von 10 MHz bis 300 MHz (Prinzip: modifizierte Scheringbrücke mit Testoszillator und Einfach-Superhet zur Auswertung der Diagonalspannung -- alles noch in Röhrentechnik). Dies ist das gute Stück nach Instandsetzung und Abgleich -- und es zeigte sich mal wieder, dass eine Messbrücke eine feine und genaue Sache ist, trotz teilweise abenteuerlicher Bedienung und den erforderlichen Knobeleien und Auswertungen mit dem Taschenrechner. Die Messbrücke ermittelt nämlich eine Induktivität als „negative Kapazität“ und der einstellbare Bereich ist auf 100 pF begrenzt. Deshalb musste bei L = 1 µH die Messfrequenz auf 16 MHz erhöht werden, um mit der „negativen Ersatzkapazität“ unter 100 pF zu bleiben. Und der gelbe Aufkleber auf der Geräte-Oberseite (= Achtung: 110 V Betriebsspannung) erfordert peinliche Aufmerksamkeit sowie einen zusätzlichen externen Versorgungstrafo. Sonst ist es vorbei mit den Röhren und der Freude…



Dann wurde es ernst und zunächst wurden damit die bekannten und schon öfters eingesetzten Neosid-Spulenbausätze mit Abgleichkern (Grundfläche: 7,5 mm x 7,5 mm) untersucht. Hier sehen wir ein solches Exemplar, aber Versuchsreihen mit unterschiedlichen Kernwerkstoffen, Bauarten und Drahtstärken lieferten immer dasselbe Ergebnis: mehr als eine Güte Q = 100 war für Filter bei 10 MHz mit dieser Bauart einfach nicht zu schaffen.
Schalenkern-Bausätze als Alternative sind jedoch deutlich größer und relativ teuer. Bei Luftspulen mit Abgleichkern schafft man die kleinen Abmessungen bei der geforderten hohen Güte von 150 bei 10 MHz für L = 1 µH nicht und deshalb blieben als Alternative nur noch die bekannten **Amidon Ringkerne**.



Sie sind leicht erhältlich und billig (Lieferant: z. B. Reichelt Elektronik) und die Güte-Angaben im Datenblatt aus dem Internet machen Hoffnung. Außerdem versprechen sie auf Grund des geschlossenen Feldlinienweges sehr geringe Streuung und dadurch einen Verzicht auf zusätzliche Abschirmung der Einzelkreise. Es bleibt lediglich die Wahl zwischen Ferrit und feinstem Eisenpulver als Magnetmaterial im Kern. Ferrit ergibt sehr hohe Permeabilität (das bedeutet hohen AL-Wert und damit geringe Windungszahlen!) und sehr erfreuliche Gütewerte. Seine Kehrseite als geschlossener Ringkern ist die frühe Sättigung und die deshalb bald zu spürende Nichtlinearität mit zunehmender Aussteuerung, die zu Signalverzerrungen führt. Reineisenpulver dagegen mit seinen mikroskopisch feinen Eisen-Partikeln enthält genügend Kunststoff-Bindemittel zwischen den einzelnen

Körnchen und ist wegen dessen Luftspalt-Wirkung viel höher aussteuerbar -- und das bei hohen Güten, da durch die winzigen Korngrößen die Wege für Wirbelströme minimal sind. Deshalb wurde dem Eisenpulverkern der Vorzug gegeben und ein kleines Sortiment zum Experimentieren bestellt, nämlich die Versionen T20-2, T37-2 und T44-2. Das Bezeichnungssystem ist einfach, denn die erste Zahl gibt den Außendurchmesser in 0,01 Inch an, während die Zahl „2“ nach dem Bindestrich besagt: „optimaler Frequenzbereich = 1…30 MHz“. Außerdem ist der Ring noch in Farbe getaucht und für diesen Frequenzbereich leuchtet er in Rot-Braun.

Diese drei Kerne wurden soweit mit Kupferlackdraht bewickelt, dass bei allen die Induktivität in der Größenordnung von 1 µH lag, wobei die Windungen über den gesamten Kreisumfang verteilt wurden. Die einzelnen Induktivitätswerte wurden mit dem guten alten „LARU“ von Rohde und Schwarz bestimmt und dann ging es an die Messungen mit dem Boonton RX – Meter. Wie schon erwähnt, musste 16 MHz als Messfrequenz verwendet werden, um im Einstellbereich der Brücke von maximal „-100 pF“ zu bleiben und das ergab die folgende interessante Tabelle:

Kerntyp	Kern-Außen-Durch-messer	Windungen	Draht-Durchmes-ser	Induktivität	Parallel-Verlustwider-stand	**Güte Q bei16MHz**
T20-2	5,08mm	19	CuL 0,2	0,97µH	12,9k	**132**
T37-2	9,53mm	15	CuL 0,3	1,02µH	18,8k	**183**
T44-2	11,2mm	13	CuL 0,3	0.99µH	15,9k	**159**

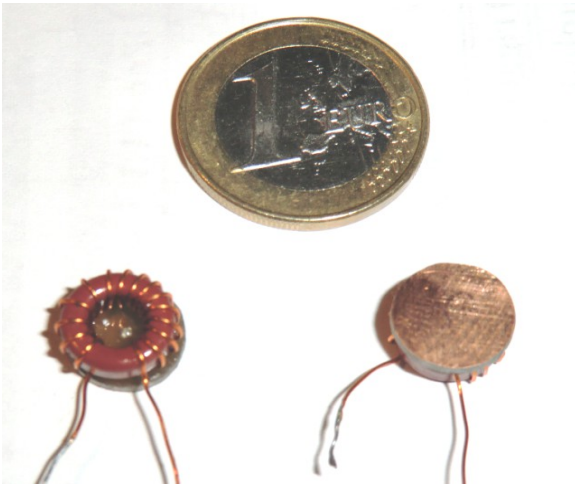
Daraus erkennt man, dass der Kern T37-2 in dieser Ausführung die höchste Güte von Q = 183 liefert und er wurde für die Realisierung des Filters ausgewählt.

So geht es nun weiter zum Problem der korrekten Befestigung der Kerne auf der Platine, damit

a) absolute mechanische Stabilität und Reproduzierbarkeit gewährleistet ist und

b) die schöne hohe Spulengüte nicht allzu sehr vermindert wird.

Nach etlichen Experimenten und manchem Grübeln lief es auf folgende Lösung hinaus:



Aus einer einseitig kaschierten Platine, die aus 1,52 mm starkem FR4-Material (= Glasfasergewebe + Epoxidharz) besteht, werden kleine Scheiben mit der Größe des bewickelten Ringkerns mit der Laubsäge ausgesägt. Auf die nicht mit Kupfer kaschierte Seite dieser Scheibe wird der auf 1 µH abgeglichene Ringkern mit UHU-Plus (= Epoxidharz) aufgeklebt. Zusätzlich sind auf der Oberseite des Kernes die Windungen mit einem Hauch von Kleber zusätzlich fixiert. („Auf 1 µH abgeglichen“ geht so vor sich: man schließt die Spule an das Induktivitätsmessgerät an und verschiebt die Anfangs- oder Endwindungen solange, bis genau der geforderte Indukivitätswert erreicht ist. Erst dann wird geklebt). So erhält man Bauteile, die auch bei herzhaftem Zufassen ihren Wert nicht mehr ändern und bei denen die Windungen stets einen Abstand von 1,5 mm zur Kupferfläche auf der Unterseite einhalten -- so kann sich weder die Spulenkapazität noch die Güte beim Hantieren und Bestücken verändern (…bei schmalbandigen Pässen reichen nämlich schon winzigste Änderungen der Induktivitäten in der Promille - Gegend aus, um die Durchlasskurve gründlich zu verbiegen…). Und der Trick mit der Kupferschicht auf der Unterseite bedeutet: egal, ob man nun das Bauteil auf die Filterplatine lötet oder aufklebt -- all das hat keinen Einfluss mehr auf

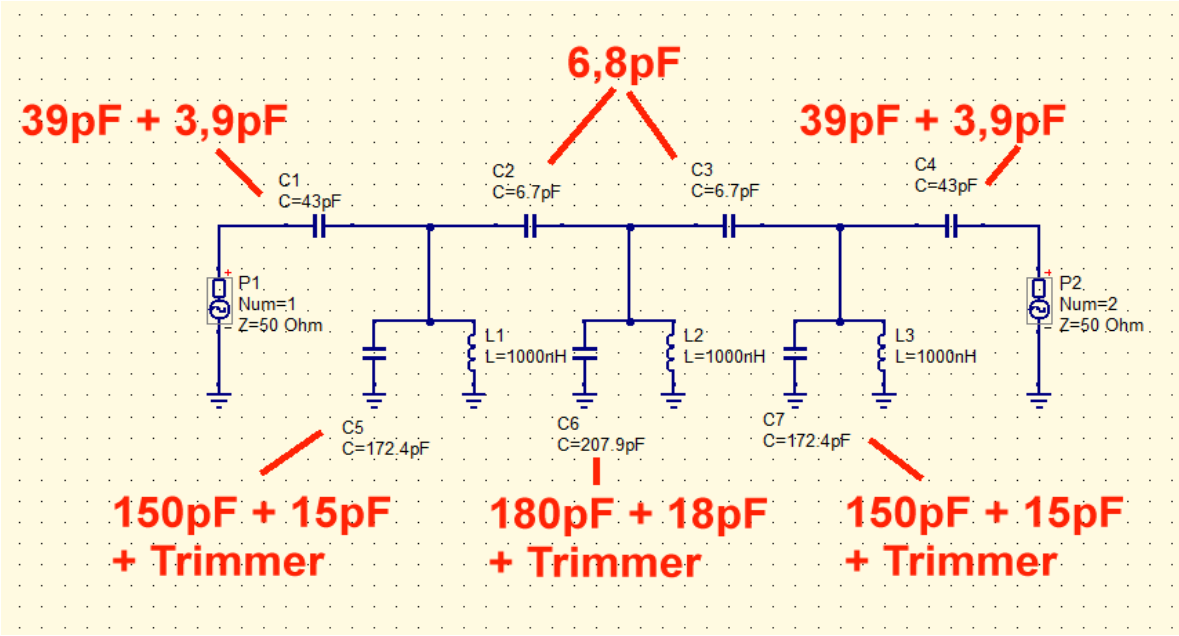
die Spulendaten.

Die einzige bange Frage war jedoch, ob die recht nahe Kupferschicht an der Unterseite der FR4-Scheibe sowie die Kleberei die Spulengüte beträchtlich verringert hat. Also wurden alle drei gefertigten Exemplare für die erste Filterschaltung wieder dem Boonton – RX – Meter zugeführt, um die Induktivitäten und Güten zu ermitteln.
Hier das Ergebnis:
Die Induktivitäten betragen 1,013 / 1,01 / 1,015 µH und die erzielten Gütewerte liegen bei 162 / 167 / 164.

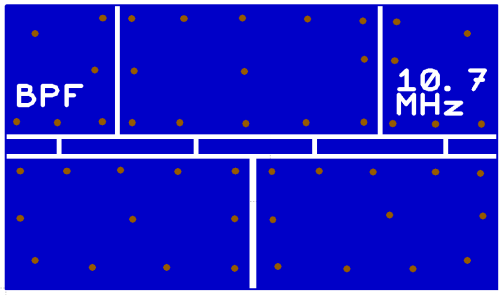
Hiermit wurden die Spulenprobleme als erledigt betrachtet.

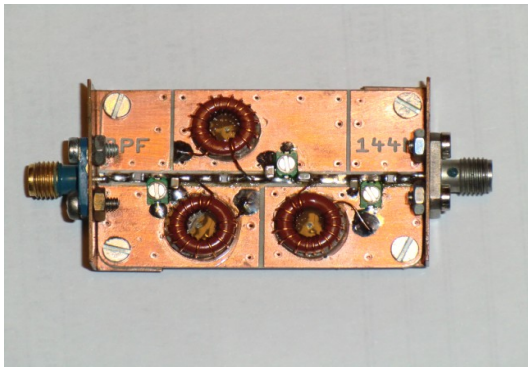
15.7. Der erste Test: Filterprototyp mit N = 3

Als erste Maßnahme muss das vom Filter-Designprogramm gelieferte Schaltbild in eine praktische Version übertragen werden, die sich fertigen lässt. Dazu werden zunächst die verschiedenen Koppelkondensatoren durch Parallelschaltungen aus SMD-Normwerten ersetzt, die den geforderten Wert möglichst genau (= mit höchstens 3% Abweichung) annähern. Bei den Kreiskondensatoren verfährt man ebenso, aber dort wird zusätzlich jeweils ein Trimmkondensators in jeden Kreis eingefügt. Damit lässt sich nicht nur bequem die geforderte Gesamt-Kreiskapazität mit ihrem krummen Wert einstellen, sondern auch die „Fertigungsstreuung“ der drei Ringkernspulen wird so korrigiert.



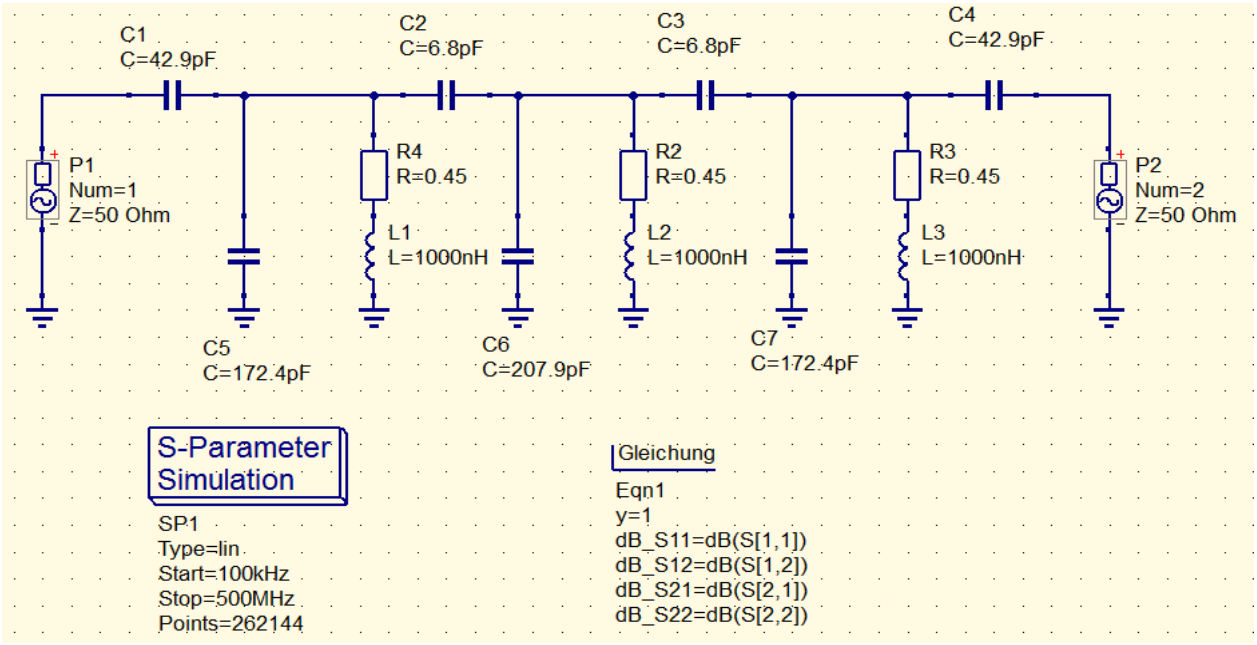
Der Prototyp der Filterschaltung wird auf dieser Platine mit den Abmessungen 30 mm x 50 mm aufgebaut. Gut zu erkennen sind die voneinander isolierten Masse-Inseln für die drei Schwingkreise samt den nötigen Durchkontaktierungen.

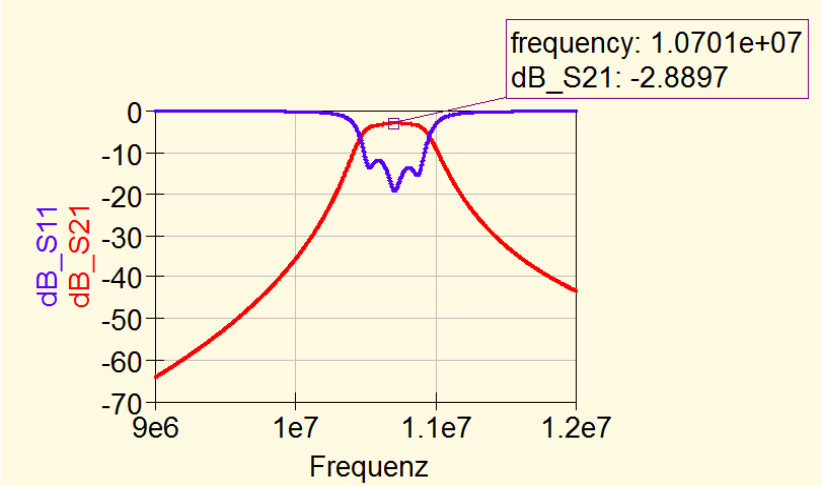




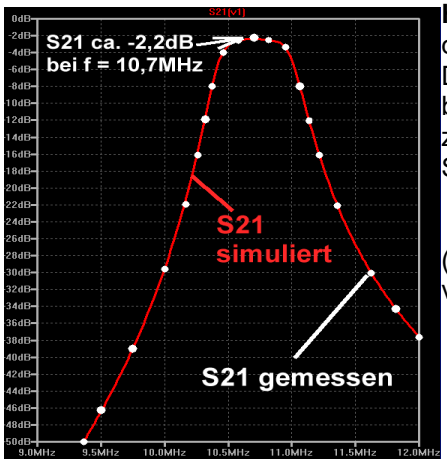
Nach der Bestückung ergibt sich der rechte Anblick, aber auf

eine Sache muss man dabei vorher beim Einlöten achten: **die Rotoren aller Trimmkondensatoren müssen mit Masse verbunden sein**! Andernfalls verändert sich die Durchlasskurve bereits, wenn man mit dem Abgleichschlüssel auch nur in die Nähe des betreffenden Trimmers kommt. Und SMD-Trimmkondensatoren tragen meist eine winzige Markierung zur Kennzeichnung des Rotor-Anschlusses. Diese praktische Schaltung wird nun in einer Simulation untersucht, bei der die Spulengüte in Form des Reihenverlustwiderstandes von 0,45 Ω einbezogen ist und die Bauteile aus SMD-Normteilen bestehen:





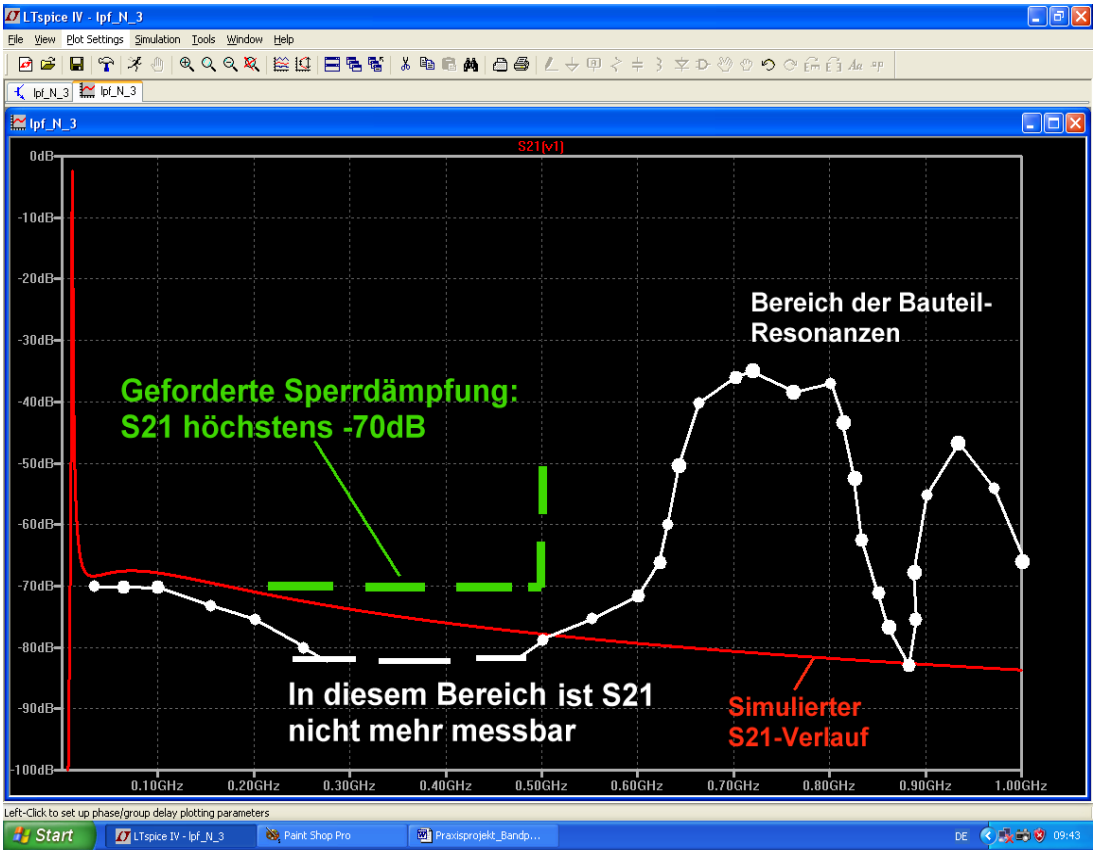
Das ist der simulierte Verlauf von S11 und S21.
Zu erwarten ist eine Durchgangsdämpfung von ca. **2,9 dB** bei 10,7 MHz.
Auch die „negativen Spitzen“ von S11 reichen wegen der Verluste nicht mehr so tief.



Hier ist das Ergebnis der S21-Messung nach erfolgreichem Abgleich der Kurve auf Maximum bei 10,7 MHz zu sehen. Die gemessene Durchlassdämpfung beträgt dort ca. 2,2 dB und das ist doch noch besser als die Simulation mit 2,9 dB. Die restlichen Messergebnisse zwischen 9 MHz und 12 MHz sind außerdem punktgleich mit dem Simulationsverlauf.

(Dieses Bild stammt aus meinem in den „UKW-Berichten“ veröffentlichten Artikel).

Nun interessierte natürlich die Weitab-Selektion (= das Verhalten im Sperrbereich) außerordentlich, denn erstens steht die Forderung mit fast 70 dB Sperrdämpfung bis 500 MHz im Raum. Zweitens ist man neugierig, wie sich die Schaltung bis zu 1 GHz benimmt.



Das ist die Antwort und da kann man ebenfalls sehr zufrieden sein. Dass oberhalb von 500 MHz die Eigenresonanzen der Bauteile Ärger machen, war zu erwarten. Trotzdem halten sich die Effekte in Grenzen und immerhin wird ein Maximum von rund -30 dB bei 720 MHz nicht überschritten.

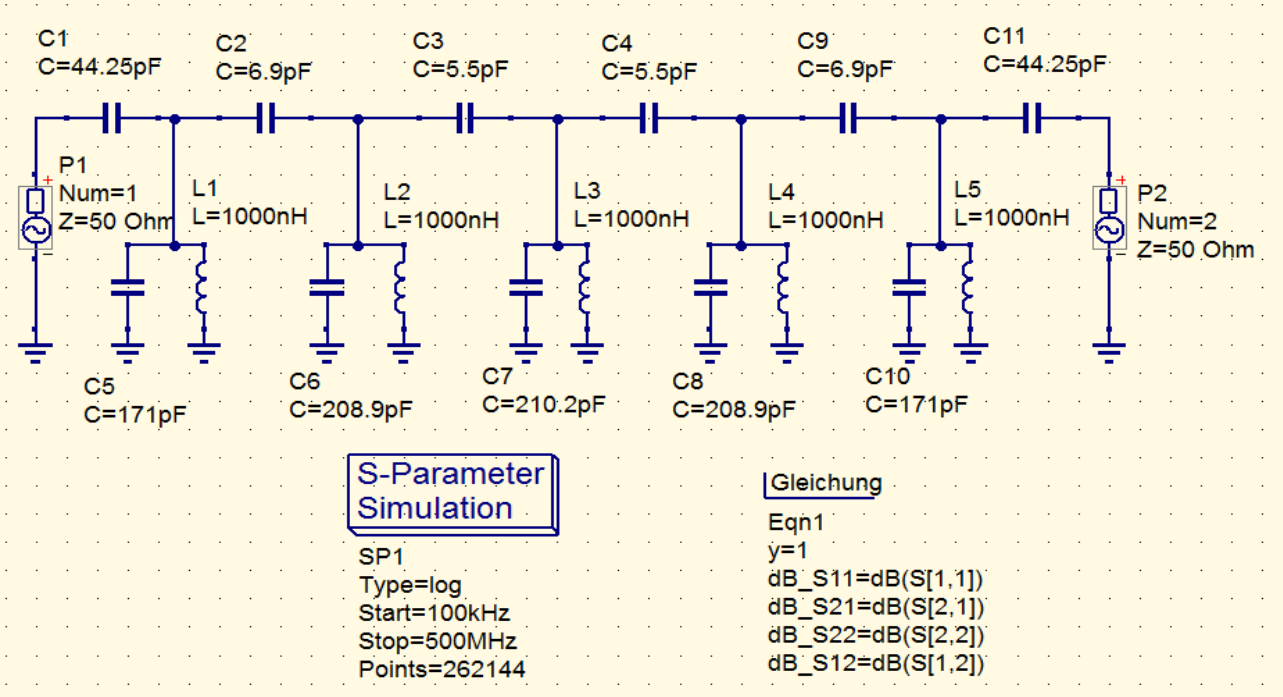
Zur Messung selbst:

Da diente ein guter Präzisions-Signalgenerator (hp 8640B) als Signalquelle zur Speisung des Filters und das Ausgangssignal beobachtete ein RACAL-DANA R.F. Millivoltmeter-True R.M.S. vom Typ 9301A mit 50 Ω Eingangswiderstand. Mit einem Sendepegel von +10 dBm und einem kleinsten einstellbaren Messbereich des Millivoltmeters von -50 dBm kommt man schon mal auf (50 + 10) dB = 60 dB Messumfang. Da noch weitere -10 dB auf der Skala des Millivoltmeters sauber kalibriert sind und sicher abgelesen werden können, ergibt das einen endgültigen Messumfang von (+10 + 50 +10) dB = 70 dB und damit entstand dieses obige Bild. Vorsichtshalber wurde das Ganze mit einem modernen Vectoranalyzer (ZVRE von Rohde und Schwarz) im Labor der Dualen Hochschule in Friedrichshafen nachgeprüft -- und das erwies sich als unnötige Maßnahme, da die Ergebnisse völlig identisch mit denen der „Heimwerker-Messung“ waren.

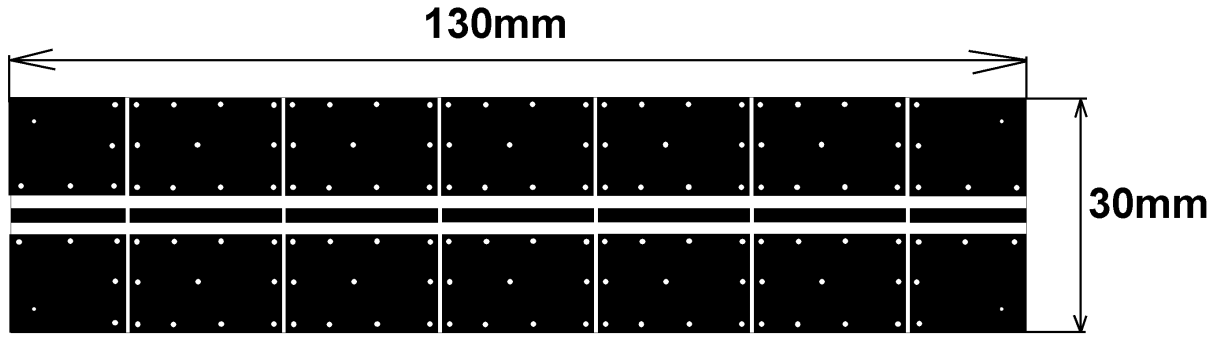
Tückisch und nur mit Vorsicht anzuwenden ist dagegen die Ermittlung der Weitabselektion mit dieser Kombination aus Pegelsender und Breitband-Millivoltmeter. Das Sender-Ausgangssignal weist nämlich Oberwellen auf und bei der verwendeten Gegentaktschaltung in der Senderendstufe täuschen speziell die stärker vorhandenen ungeraden Harmonischen plötzlich schlimmere Einbrüche in der Sperrdämpfung vor als in Wirklichkeit. So geschehen z. B. bei der Frequenz f = 240 MHz, denn deren dreifache Frequenz fällt genau auf das im obigen Bild zu sehende Maximum bei f = 720 MHz. Perfekte Abhilfe bringt natürlich ein zusätzliches Oberwellenfilter am Senderausgang oder ein Spectrum Analyzer als Spannungsmesser -- oder gleich der Einsatz des Vektoriellen Network-Analyzers. Ein Notbehelf ist dagegen bei der Breitbandmessung trotzdem möglich: man wähle den nächst höheren Ausgangsspannungsbereich (hier +20 dBm) und drehe anschließend am Feinregler den an die Schaltung abgegebenen Pegel wieder bis auf +10 dBm zurück. Durch die verminderte Aussteuerung der Endstufe (es sind ja jetzt 10 dB weniger) nehmen die Amplituden der ungeradzahligen Oberwellen um den dreifachen Wert, also um -30 dB, ab. Damit nähern sich die Ergebnisse der Breitbandmessung wieder den Aussagen des Vektoriellen Network-Analyzers.

15.8. Das endgültige Ziel: die fünf polige Version

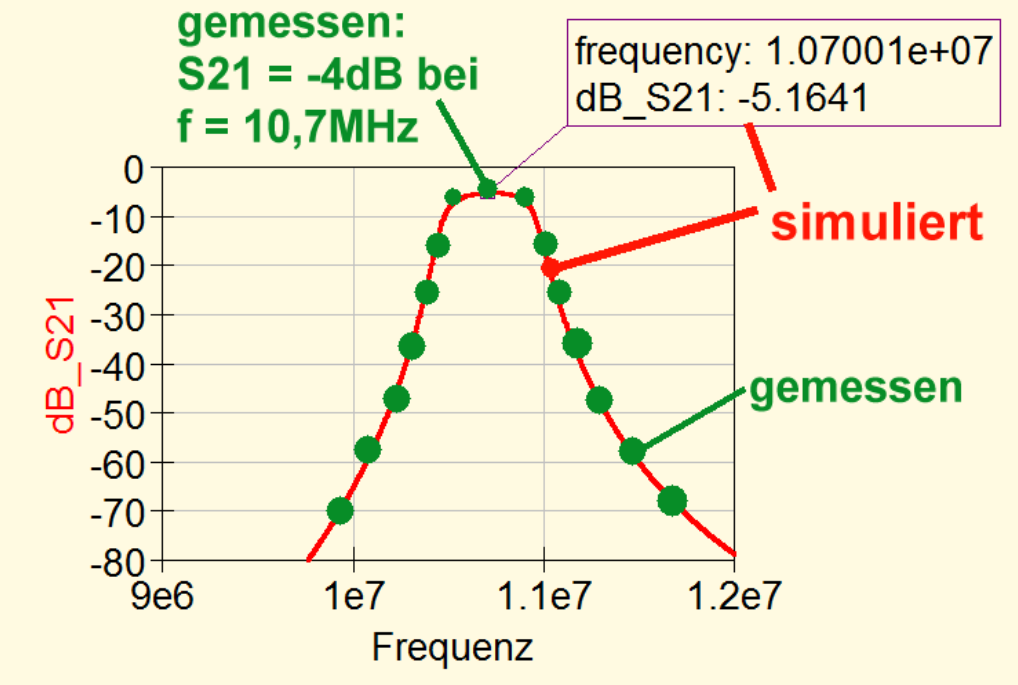
Die erforderliche Filterberechnung hatten wir ja schon in Kapitel 13.5., aber das Ergebnis muss erneut in diese praktische Schaltung mit SMD-Kondensatoren und Ringkernspulen und SMD-Trimmern umgesetzt werden. Das ist die Schaltung.....



...und das die vorgesehene Platine:

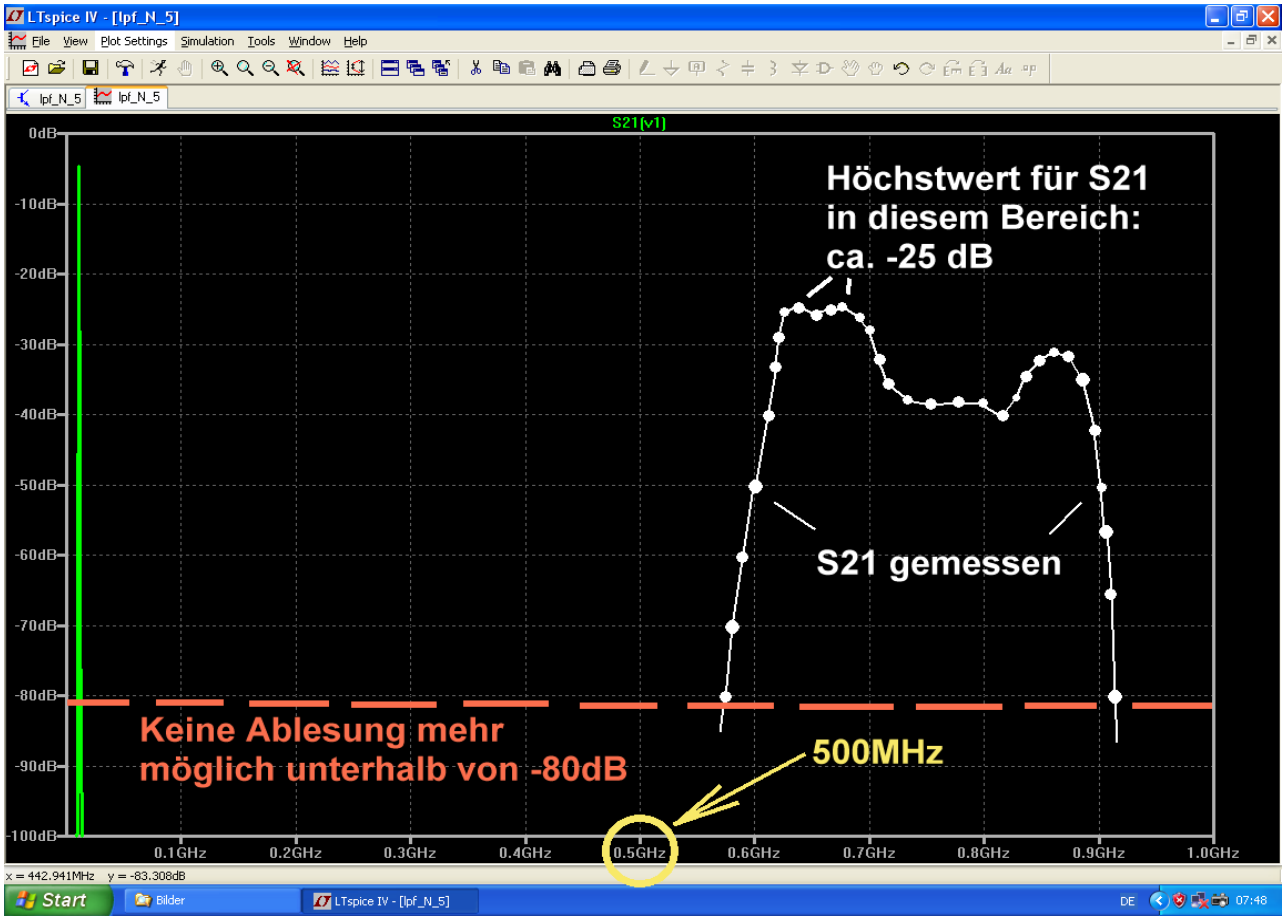


(Material wie immer: Rogers RO4003 mit einer Dicke von 32 MIL =0,813 mm). Man erkennt deutlich das Konzept mit den voneinander getrennten und mit vielen Durchkontaktierungen versehenen einzelnen Masse-Inseln. In der Mitte läuft eine 50 Ω – Microstrip – Leitung vom Eingang zum Ausgang, die für jeden erforderlichen Koppelkondensator durch einen kleinen Luftspalt („Gap“) unterbrochen ist.
Bei der Simulation ist man etwas im Zwiespalt, ob diese Leitungsstücke in die Simulation mit einbezogen werden sollen. Aber: unterhalb von 500 MHz bilden sie lediglich Zusatzkondensatoren und das kann in jedem Schwingkreis mit dem dort vorhandenen Trimmkondensator ausgeglichen werden. Wenn überhaupt, dann spüren wir die Transformationswirkungen der Leitungsstücke erst oberhalb von 1 GHz – und genau das bestätigte eine extra durchgeführte Kontrollsimulation.

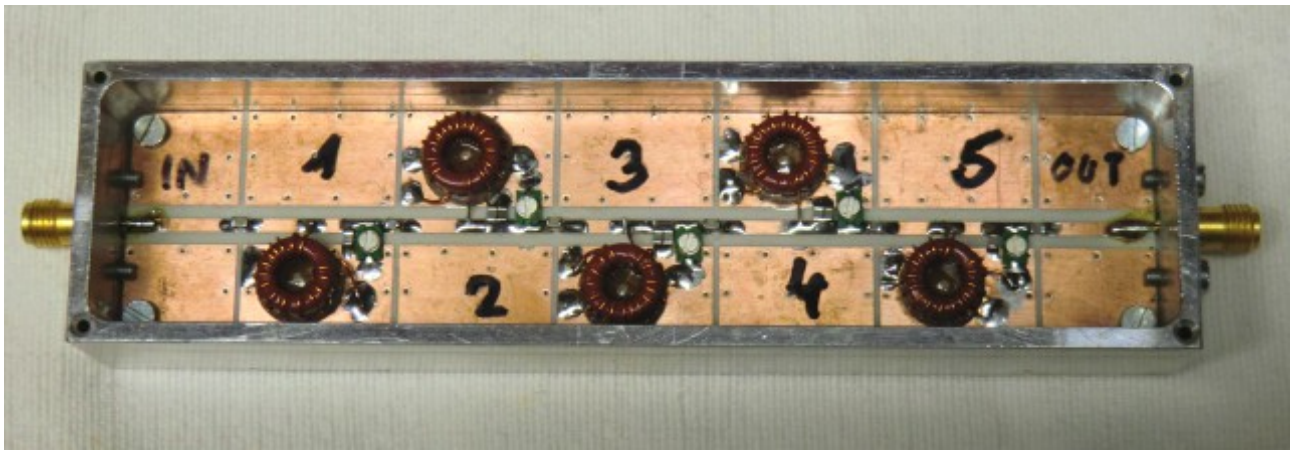


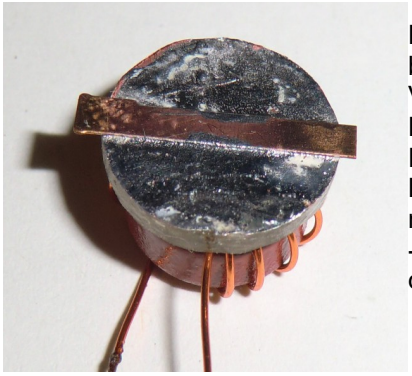
Also begnügen wir uns mit dieser S21-Simulation samt dem Mess-ergebnis.

Da interessiert auch wieder die Weitabselektion bis 1 GHz und die lässt wohl keine Wünsche offen. Gemessen wurde sie mit dem Network-Analyzer hp8410 und dieses Ergebnis anschließend nochmals mit dem Rohde und Schwarz – Analyzer ZVRE kontrolliert.

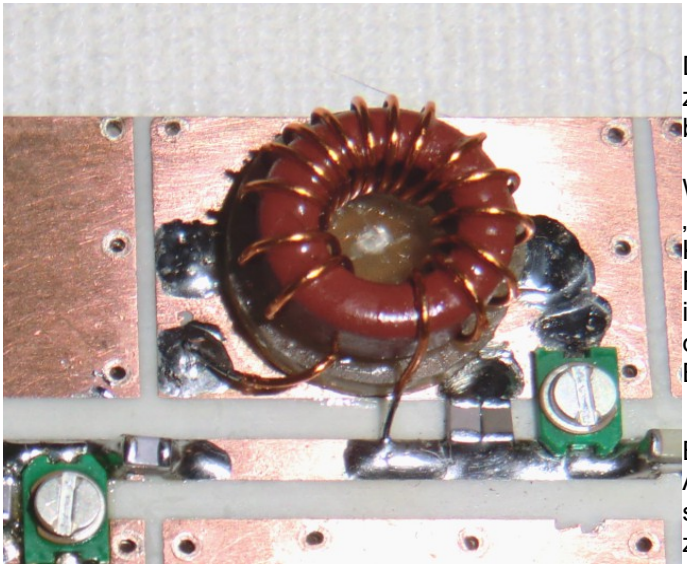


Die fertige Schaltung im Gehäuse ist hier zu bewundern, aber gegenüber der ersten, dreipoligen Filterversion wurde doch noch eine Verbesserung vorgenommen.





Die Ringkernspulen können nämlich gar nicht stabil genug ausgeführt und befestigt werden, denn bereits kleinste Veränderungen (wie das leichte Verbiegen eines Anschlussdrahtes beim Löten) verformen die Durchlasskurve merklich. Also wurde auf die Unterseite der kreisförmigen FR4-Trägerplatte für den Ringkern ein Kupferblechstreifen mit 15 mm Länge, 2,5 mm Breite und 0,4 mm Dicke aufgelötet. Damit lässt sie sich nun wie ein SMD-Bauteil einsetzen und bei Bedarf auch wieder auslöten -- jetzt sitzt sie unverrückbar und mit absolut starren Anschlussdrähten auf der Platine.



Die Aufnahme bestätigt diese Tatsache, denn sie zeigt einen der fünf Schwingkreise mit allen beteiligten Bauteilen.

Wer genau hinschaut, sieht außerdem die „Huckepack-Ausführungen“ der Koppelkondensatoren wegen der erforderlichen Parallelschaltungen. Das zuverlässig hinzukriegen ist schwieriger, als es aussieht und deshalb muss doch noch etwas zum Thema „Löten von SMD-Bauteilen“ angemerkt werden.

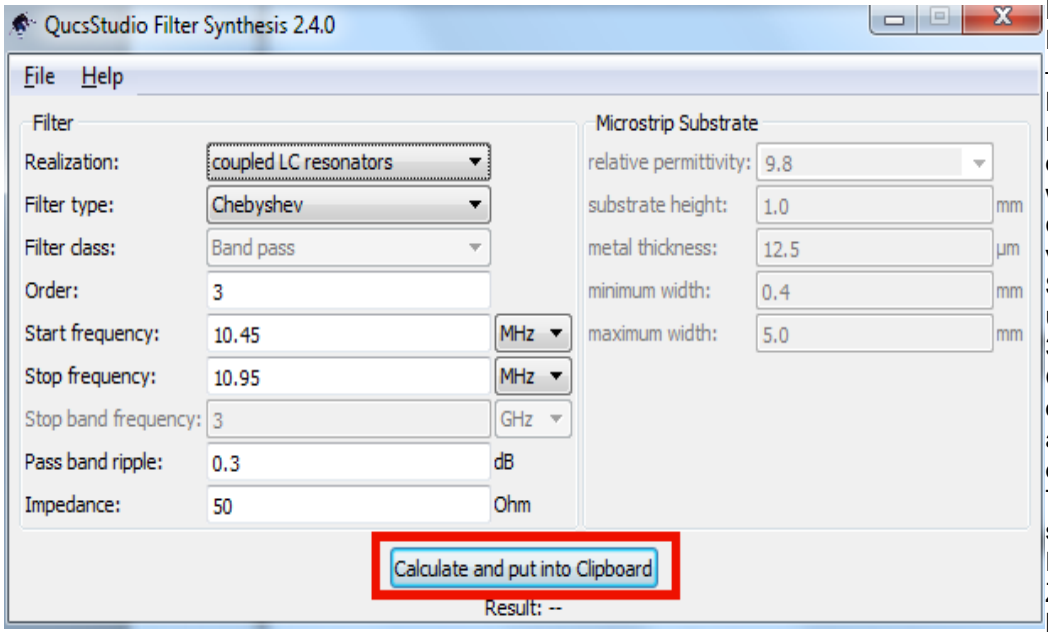
Bernd Kaa hat in seinem exzellenten Anleitungsartikel in Heft 2 / 2011 der UKW-Berichte schon alle wichtigen Informationen zusammengetragen. Aber die Sache mit der dort empfohlenen SMD-Lötpaste hat doch einige kleine Häkchen. Nicht vom Erfolg oder von der Handhabung her -- ganz im Gegenteil, sie ist hervorragend, ist die Standardmethode in der industriellen Fertigung und das Ergebnis ist überzeugend! Aber: für den Kleinverbraucher recht teuer, neigt (meiner Erfahrung nach) zur Alterung und sollte deshalb zügig verbraucht werden. Da das nicht immer möglich war, bin ich wieder reumütig zur spottbilligen Tube mit „Löthonig“ der Firma Bürklin-Elektronik zurückgekehrt…mit derselben Tube (zu 1,20 Euro) löte ich schon mehrere Jahre ohne Alterungsprobleme meine SMD-Schaltungen und den fehlenden Zinnzusatz der Lötpaste ersetze ich durch einen winzigen, frisch zugeführten Tropfen von 0,5 mm-SMD-Lötzinn an der Lötkolbenspitze. Bitte mal selbst ausprobieren - da kann man nur staunen, wie schön das funktioniert! Aber jedes Wort in Bernd Kaa’s Artikel kann ich nur bestätigen. Bei Nichtbeachtung seiner Empfehlungen artet die SMD-Löterei nämlich in Murks aus und verursacht grandiosen Ärger. Nicht nur in der Schaltung, sondern auch beim Anwender.

15.9. Zum Vergleich: Entwurf mit dem neuen Filterdesigner von qucsstudio (ab Version 2.4.0)

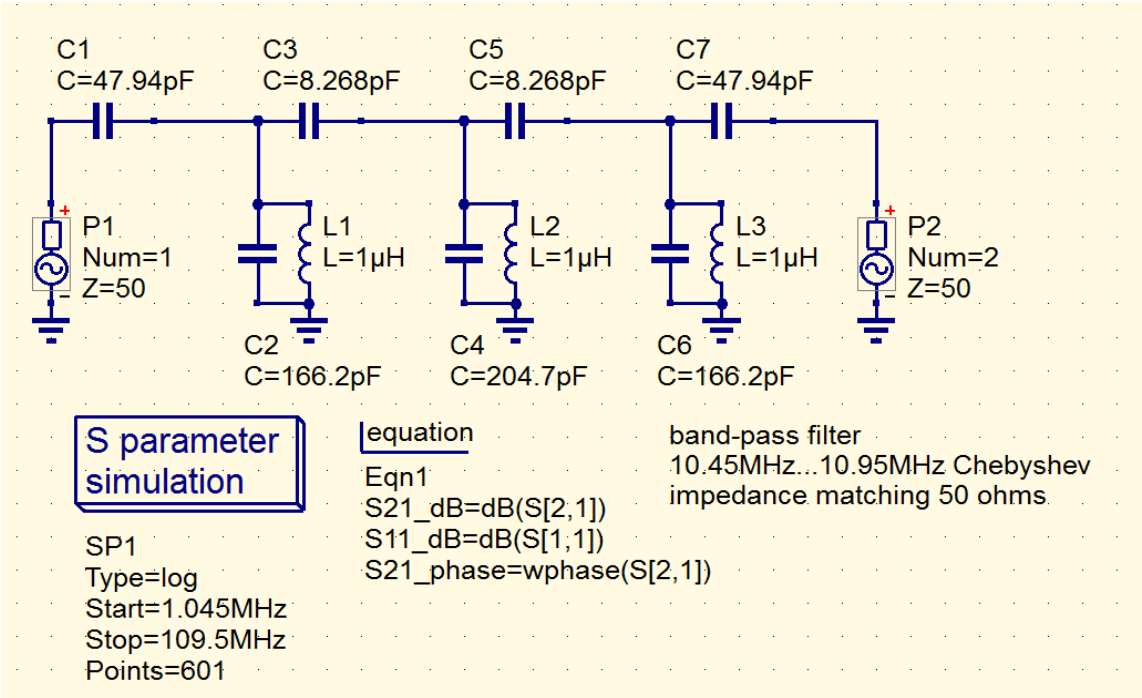
Schön, dass diese Lücke mit dem „Narrow Bandpass“ ab qucsstudio Version 2.4.0. geschlossen ist, denn wenn alles klappt, ist ja die Sache mit der DOS-Box überflüssig. Kontrollieren wir mal den realisierten Entwurf der dreipoligen Filterversion. Das waren die Spezifikationen für das Programm „fds.exe“:

Center Frequency	**= 10.7 MHz**
Poles	**= 3**
Ripple	**= 0.3**
3 dB or ripple width	**= 3 dB**
Bandwidth	**= 0.5 MHz**
Zsource	**= 50**

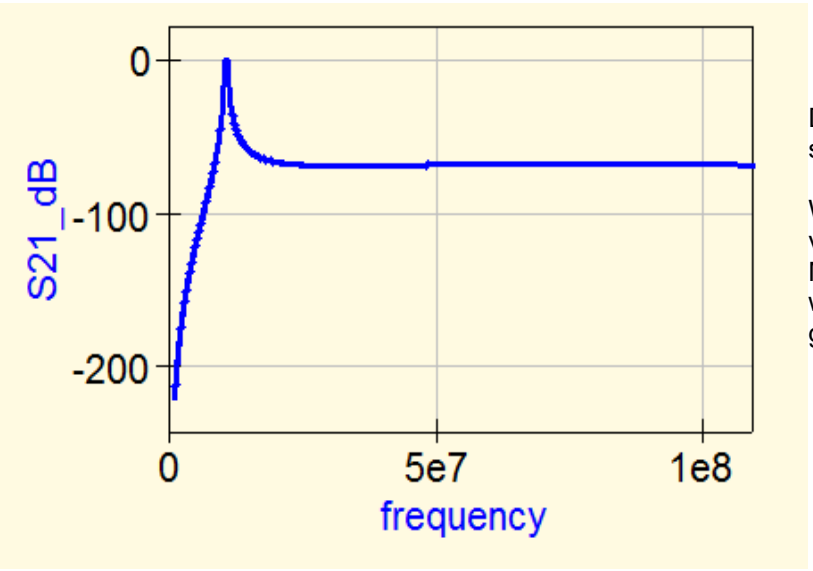
Also starten wir qucsstudio 2.4.0. und gehen in das „Tool“ -Menü. Dort finden wir „Filter Synthesis“ und wenn wir das öffnen, können wir gleich auf „c**oupled LC resonators**“ umstellen:



Die Eingabe der Daten ist problemlos – **lediglich die Bandbreite kann nicht direkt eingegeben werden. Sie wird durch die Angabe von Start- und Stoppfrequenz (= unterer und oberer 3 dB – Grenzfrequenz) ersetzt.** Drückt man anschließend auf die rot markierte Taste, dann wird simuliert und das Ergebnis in die Zwischenablage kopiert.

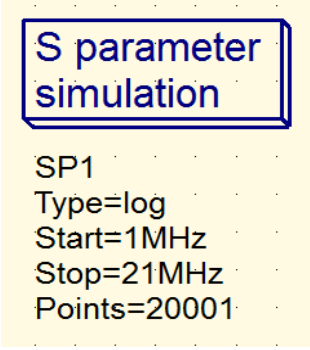


Offensichtlich unterscheiden sich der filter calculator von qucsstudio und „fds.exe“ etwas in ihren Berchnungsmethoden, denn wir finden Unterschiede zwischen 5 und 10 % bei den vorgeschlagenen Bauteilwerten.
Aber auch der Qucsstudio-Calculator arbeitet mit der in Kapitel 14.5. vorgeschlagenen Induktivität von 1µH in allen drei Kreisen.



Das Simulationsergebnis für S21 in dB sieht beim ersten Durchgang so aus.

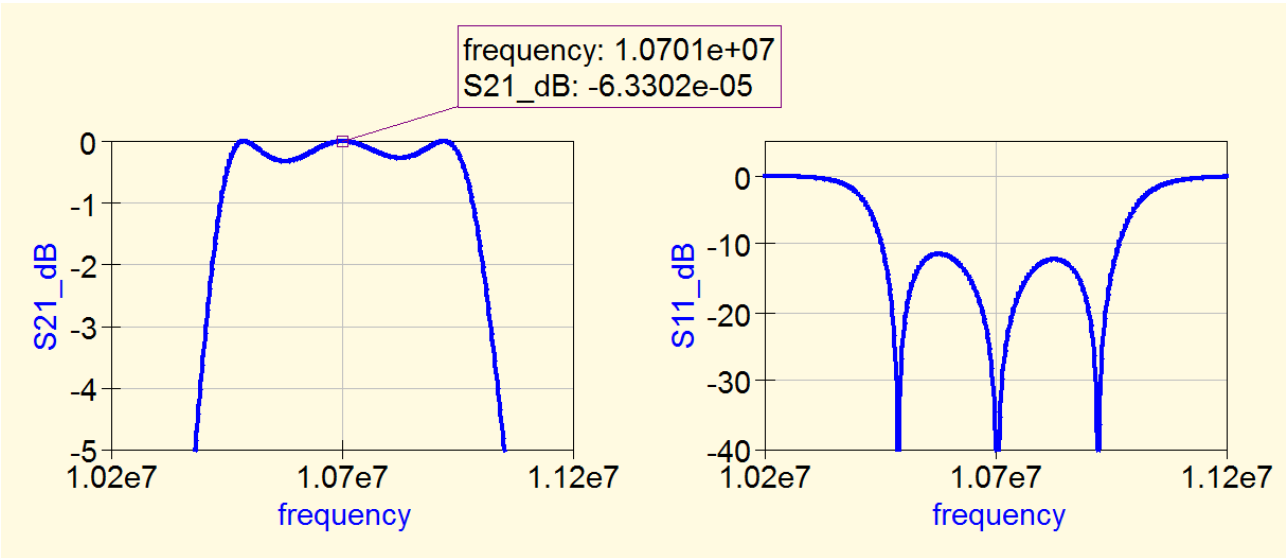
Wieder ist die schlechte Sperrdämpfung von nur ca. 70 dB (simuliert bis ca. 110 MHz) zu erkennen. Aber darüber hatten wir ja schon bei der fds-Simulation gesprochen.



Allerdings ist die automatisch vorgesehene Anzahl von 601 Punkten für einen Bereich von 1...110 MHz viel zu grob, um Details auszuwerten.

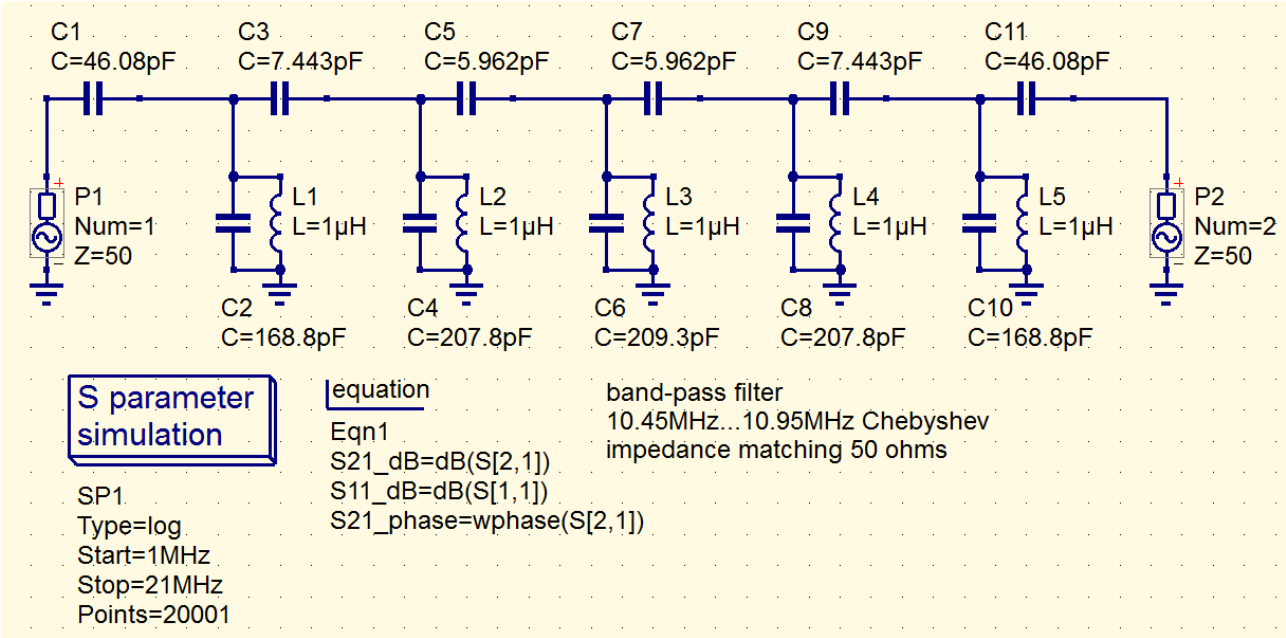
Deshalb wiederholen wir das Ganze nur für 1...21 MHz, aber dafür mit 20001 Punkten. Dann können wir uns gleich die S21-Tschebyschef-Wellen sowie die Eingangs-Reflektion S11 im Bereich von 10,2....11,2 MHz genau ansehen.

Da sollte man sich aber auch zusätzlich die Freude mit einem Frequenzmarker machen, bei dem man die Anzeige auf „5 Digits“ umstellt:

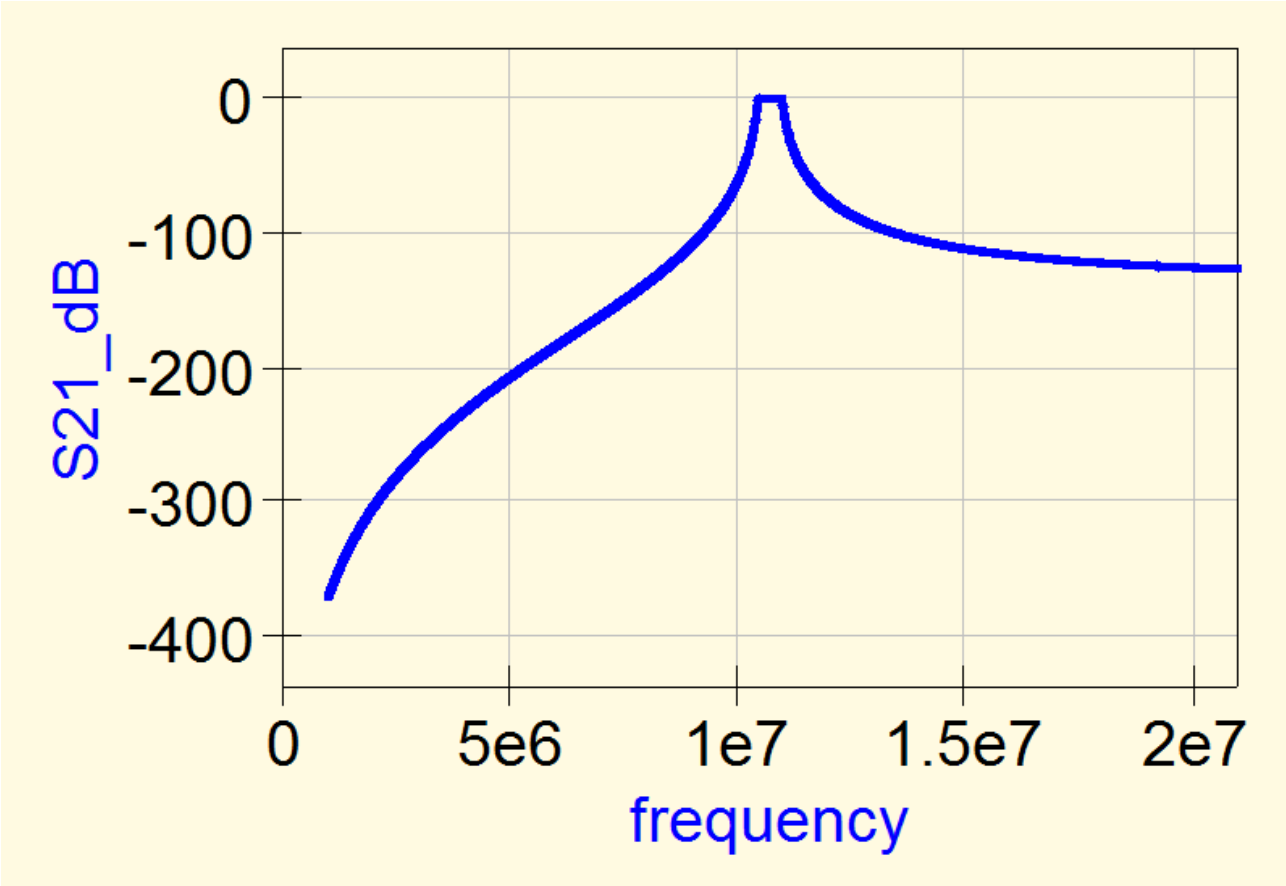


Das passt alles....

Da macht es richtig Freude, auch die fünfpolige Version auszuprobieren:



Im obigen Bild wurde gleich anschließend der Simulationsbereich auf 1MHz bis 21 MHz und 20001 Punkten umgestellt.
Das kam dabei heraus. Immerhin reicht der Anzeige-Umfang der vertikalen Achse nun von Null bis -400 dB.....und die Sperrdämpfung zwischen 12 und 100 MHz ist nun mehr als ausreichend...



16. Entwicklung einer WLAN (= 2,45 GHz) - Patchantenne für lineare Polarisation

16.1. Zuerst etwas Antennen-Grundlagen

Eine Antenne soll elektrische Energie in Form von elektromagnetischen Feldern, die in einer Leitung (= z. B. zwischen 2 Drähten) geführt werden, an den freien Raum übergeben (= Freiraum-Welle). Der freie Raum kann dabei z. B. als Hohlleiter betrachtet werden und weist einen Wellenwiderstand von 120π = 377 Ω auf. **Da hierbei Wirkleistung von der Antenne wegtransportiert wird, müssen Elektrisches Feld (entspricht der Spannung U) und Magnetisches Feld (entspricht dem Strom I) räumlich zusammen auftreten, also in Phase sein (das vektorielle Produkt ergibt dann den Poyntingschen Vektor für die abgestrahlte Leistung pro Flächeneinheit). Dabei stehen aber die elektrischen und magnetischen Feldlinien immer senkrecht aufeinander.** Diese dauernde Energie-Abgabe an den freien Raum muss man natürlich am Antenneneingang in Form eines dort auftretenden Lastwiderstandes, nämlich dem **Strahlungswiderstand** (radiation resistance) merken, der dann von der angelegten Spannungsquelle gespeist werden muss.

Dieser Strahlungswiderstand hängt von der Antennenkonstruktion, dem Durchmesser der Antennenstäbe, der Länge der Antennendrähte, der Speisefrequenz, dem Einfluss der Umgebung usw. ab. Da muss man dann in der Fachliteratur nachlesen….

Viele Antennen sind „Schmalband-Antennen“ und deshalb „**Offene Schwingkreise**. Das bedeutet:

d) Sie weisen eine **Resonanzfrequenz auf, bei der alle Blindanteile verschwinden**. Es bleibt dort eine Mischung an Ohm’schen Widerständen übrig, nämlich der Strahlungswiderstand (Siehe oben) sowie die Verluste im Antennenwerkstoff, in der Speiseleitung, im Erdboden usw.

e) Weicht man mit der Speisefrequenz von der Resonanzfrequenz ab, dann treten -- wie bei jedem Schwingkreis! -- immer stärkere kapazitive bzw. induktive Blindanteile auf.

f) Der Aufbau einer Antenne als „Offener Schwingkreis“ bedeutet, dass wir in der engen Umgebung der Antenne (= **Nahfeld**) sehr starke elektrische und magnetische Felder finden. Die sind aber (…wie sich das für Blindelemente gehört…) um 90 Grad gegen den Wirkstrom phasenverschoben und speichern nur Blindleistung. Aber sie können wegen der zugehörigen hohen Feldstärken gefährlich für Organismen werden! Diese Felder klingen mit der dritten Potenz des Abstands zur Antenne ab.

g) **Ab ca. fünf, aber spätestens ab zehn Wellenlängen Abstand zur Antenne befinden wir uns garantiert im Fernfeld**. Hier transportieren die Felder nur noch Wirkleistung von der Antenne weg, müssen **also räumlich phasengleich** sein. Wie erwähnt, stehen elektrische und magnetische Feldlinien an jeder Stelle im Raum immer senkrecht aufeinander. **Diese Felder nehmen linear mit (1 / Abstand zur Antenne)** ab und damit erzielen wir die große Reichweite der Funkwellen!

Da sich die abstrahlte Energie beim Wegwandern von der Sendeantenne dauernd auf immer größere Raumwinkel verteilt, nimmt auf der Empfangsseite die Energie pro Flächeneinheit nach der „**Friis“-Beziehung** ab. Das ergibt die berühmte **Formel für die Freiraumdämpfung**. Sie lautet, wenn man die Gewinne von Sende- und Empfangsantenne einbezieht:

$$P_{Empfang} = P_{Sender} \bullet G_{Sendeantenne} \bullet G_{Empfangsantenne} \bullet \left(\frac{\lambda}{4\pi \bullet d} \right)^2$$

„d“ ist dabei der Abstand zwischen Sende- und Empfangsantenne. Mit „λ“ ist natürlich die Wellenlänge gemeint, bei der die Berechnung erfolgen soll und die lässt sich über die Lichtgeschwindigkeit und die Sendefrequenz errechnen.

16.2. Grundlagen der Patchantennen

Sie bestehen aus einem Stück Leiterplattenmaterial (= „PCB“) das beidseitig mit Kupfer kaschiert ist. Die Unterseite bildet eine durchgehende Massefläche und auf der Oberseite der Platine finden wir in der einfachsten Form ein Quadrat oder Rechteck aus Kupfer (= „Patch“). Dabei muss die Platine für eine korrekte Arbeitsweise deutlich größer sein als der Patch (Richtwert für den Überstand: wenigstens 3…. 5 mal Platinendicke -- und noch mehr ist noch besser.
Dieser Patch wird nun so ausgelegt, dass er eine elektrische Länge von etwa λ/2 aufweist. Der genauere Wert ist etwa

Strahlerlänge = 0,49 x (λ / 2)

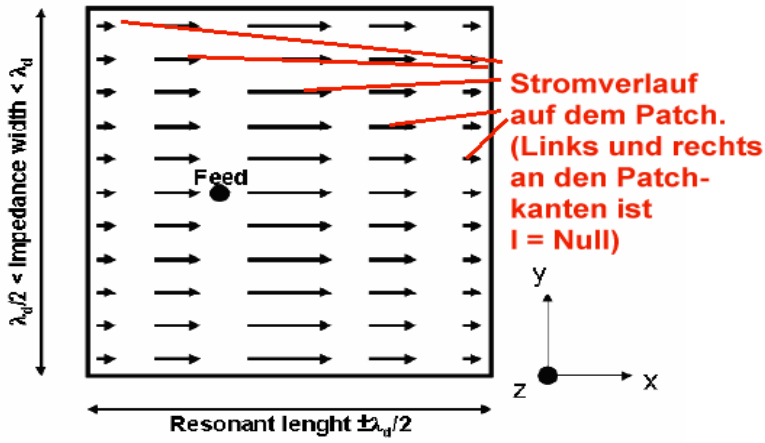
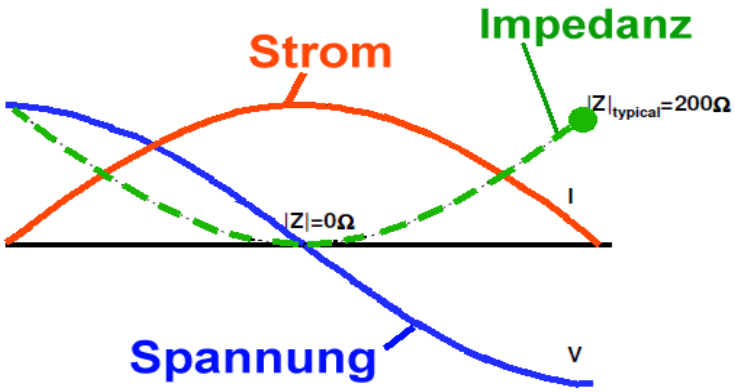


Figure 2: Current distribution on the patch surface

Speist man diese Leitung am Eingang (= l**inke senkrechte Patchkante**) mit einem Sinussignal, dessen Frequenz genau der Resonanzfrequenz entspricht, dann erhält man die nebenstehende Strom- und Spannungsverteilung.

Am Leitungsende ist die Spannung gleich groß, aber gegenphasig (λ/2 bedeutet nämlich 180 Grad Phasenverschiebung).
Genau in Patchmitte ist die Spannung Null.

Beim Strom ist es umgekehrt: bei einer reinen verlustlosen Leitung ist er am Leitungsanfang und am Leitungsende Null, hat aber in Leitungsmitte sein Maximum (Siehe Bild)



Zur Repräsentation der Abstrahlung müssen wir uns am Leitungsanfang sowie am Leitungsende noch je einen gleich großen Strahlungswiderstand denken (im Diagramm sind typisch 200Ω eingetragen). Also sind es an jeder Patchkante 400Ω, denn die beiden Lastwiderstände sind von der Leistungsaufnahme her parallel zu denken.

Genau in Patchmitte geht die Spannung durch Null und damit ist dort der Eingangswiderstand auch Null.
Also gibt es irgendwo zwischen der Mitte und der Patchkante einen Punkt, bei dem der Eingangswiderstand genau 50Ω betragen wird -- das ist der markierte „Feed point“.

Die Patchbreite (= Leitungsbreite) beeinflusst bei dieser Bauweise die Eigenresonanzfrequenz nur wenig. Man geht in der Praxis dabei immer vom quadratischen Patch aus und dann gilt: wird sie erhöht (= der Patch breiter als länger gemacht), dann steigt die Bandbreite und der Strahlungswiderstand wird kleiner.
Der Entwurf solcher Antennen erfolgt in der Praxis stets mit passender Software.
Moderner Stand des Entwurfs ist der Einsatz eines „**EM-Simulators für Flächenstrukturen“** und der bekannteste Vertreter ist hier „**SONNET**“. Für den Privatanwender gibt es dazu im Internet eine äußerst **leistungsfähige und kostenlose LITE-Version** dieses Programms, die sich großer weltweiter Beliebtheit erfreut.
Aus der Homepage des Autors (www.gunthard-kraus.de) kann dazu ein Tutorial in Deutsch oder Englisch heruntergeladen werden, in dem der komplette praktische Entwurf einer Patchantenne für 5,8 GHz im Detail beschrieben ist.

Eine Frage wurde allerdings bisher nicht beantwortet:

Weshalb und wie strahlt ein solcher Kupferfleck überhaupt?

Und dazu brauchen wir das nächste Bild:

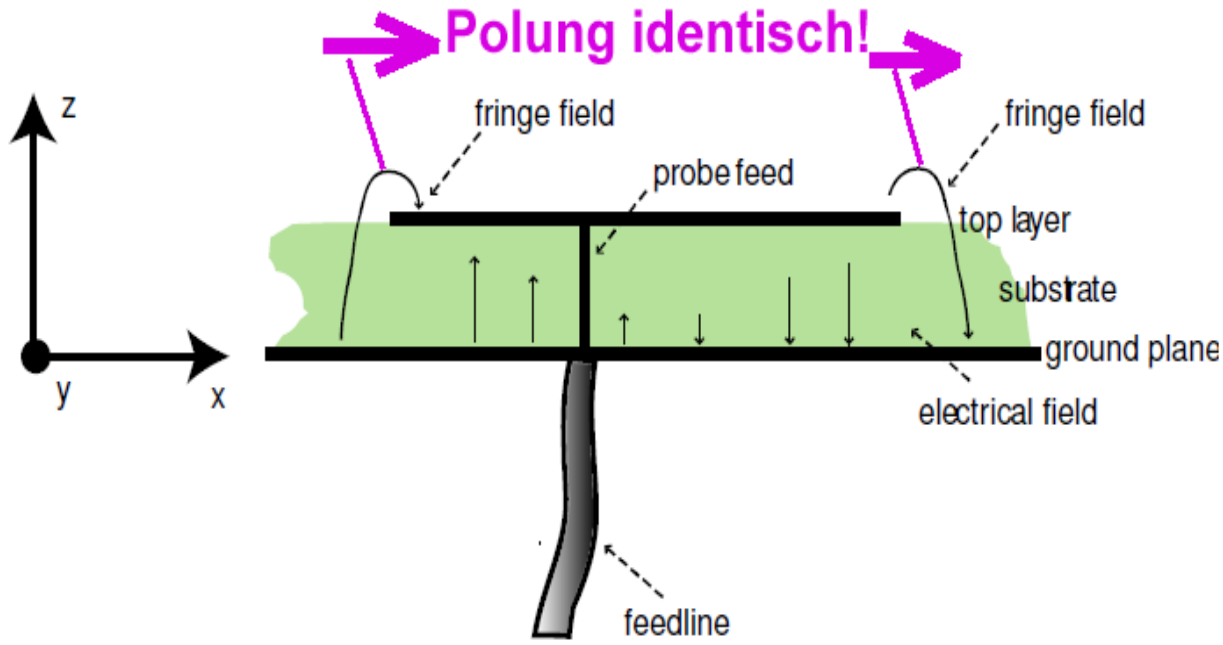


Figure 1: Cross section of a patch antenna in its basic form

Wer genau hinschaut, der sieht die an jeder Patchkante überstehenden **elektrischen Streufelder** (= fringe fields) und da steckt die Lösung! An der linken und rechten Patchkante ist die Spannung ja gegenphasig (Siehe das Bild auf der vorhergehenden Seite), aber interessanterweise zeigen dort die Feldlinien auf beiden Seiten in dieselbe Richtung und **sind deshalb gleichphasig!.**

Damit wirken diese beiden Patchkanten mit ihren Streufeldern als zwei parallel geschaltete „Schlitzstrahler“

(…ein Schlitzstrahler ist der „Komplementärtyp“ zu einem Antennenstab. Bei dem haben wir einen Draht und als Umgebung die Luft. Beim Schlitzstrahler ist das genau vertauscht: der Antennenstab wird durch Luft ersetzt und anstelle der umgebenden Luft haben wir nun Kupferflächen. Dadurch vertauschen sich auch die Richtungen von elektrischem und magnetischem Feld).

Damit ist die Frage der **Polarisation des ausgestrahlten elektrischen Feldes** gleich mit beantwortet, denn die entspricht natürlich exakt den beiden violetten Pfeilen im obigen Bild….
Und wenn die Metallisierung der Platinenunterseite genügend gegenüber dem Patch übersteht, dann wirkt sie als Abschirmung und verhindert die Abstrahlung „nach rückwärts“ (…im obigen Bild wird die Antenne also nur nach oben strahlen).
Das Richtdiagramm einer einfachen Dipolantenne ist die berühmte „liegende Acht“. Bei der Patchantenne fehlt dann einfach eine Hälfte dieser Acht und man erhält einen einfachen Kreis ohne Abstrahlung nach rückwärts.

16.3. Messung der verschiedenen Antenneneigenschaften

a) Resonanzfrequenz, Impedanzverlauf und Stahlungswiderstand
Das erledigt man am besten mit dem Vektoriellen Network-Analyser, an den man die Antenne anschließt. Der Impedanzverlauf bildet eine kreisförmige Schleife im Smith-Diagramm, die bei Resonanz genau durch den Mittelpunkt des Diagramms laufen sollte. Das wäre dann exakt ein reeller Eingangswiderstand von 50 Ω. Läuft die Kurve am Mittelpunkt vorbei, dann ergibt der geringste Kurvenabstand zu ihm den minimalen Eingangs-Reflektionsfaktor bei Resonanz. Über ihn kann dann der Eingangswiderstand berechnet werden.

Vorsicht:
Patchantennen haben beim Impedanzverlauf mehr als eine Resonanzstelle, aber nur eine, bei der wirklich abgestrahlt wird! Sie ist leicht zu bestimmen, denn erstens liegt sie dort, wo die Patchlänge genau einer halben Wellenlänge der speisenden Frequenz entspricht. Zweitens kann man bei dieser Frequenz sehr leicht die wirksame Abstrahlung prüfen, indem man die Handfläche auf die Antenne zu- und wieder wegbewegt. Durch das von der Hand reflektierte und empfangene (und sich bei Bewegung dauernd in der Phase verändernde Signal) schlägt die Anzeige des Network-Analyzers wilde Wellen.

b) Kreuzpolarisation
Zu dieser Messung verwendet man zwei völlig identische Antennen. Eine wird an einen Messsender angeschlossen, die andere speist einen Spektrum-Analysator als Empfänger. Nun werden beide Antennen genau aufeinander ausgerichtet und gesendet. Lockert man nun die SMA-Steckverbindung am Empfänger leicht, so lässt sich diese Antenne verdrehen. Bei 90 Grad Verdrehung der Antenne sollte kein Empfang mehr möglich sein, aber ein schwaches Signal wird in der Praxis trotzdem empfangen (Dämpfung meist zwischen 30 und 40 dB). Dieser Pegelunterschied in dB zwischen dem „Optimalen Empfang“ und dem schwachen Signal bei 90 Grad Verdrehung stellt die „Kreuzpolarisation“ dar.

c) Horizontales Richtdiagramm
Dazu werden Sende- und Empfangsantenne parallel zueinander ausgerichtet. Dann muss die Empfangsantenne „im Kreis um sich selbst gedreht“ und der Pegelunterschied zur Hauptempfangsrichtung (in dB) ermittelt werden.
Für die Ermittlung des vertikalen Richtdiagramms wäre ein Hubschrauber praktisch.

d) Antennengewinn
Diese Eigenschaft bestimmter Antennen beschreibt die „Richtwirkung“, also die Konzentration der abgestrahlten Energie auf bestimmte Richtungen (und natürlich die Verminderung oder sogar Unterdrückung der Abstrahlung in andere Richtungen. Dabei vergleicht man z. B. die gerade empfangene Energie mit derjenigen Leistung, die ein „Isotroper Kugelstrahler“ liefern würde (…bei dem in alle Raumwinkel und Richtungen gleiche Leistung gesendet bzw. beim Empfang erhalten wird). Dieser Pegelunterschied wird als Antennengewinn bezeichnet und in

dBi

angegeben.

Manchmal bezieht man sich auch auf den **einfachen Dipol**, der aber gegenüber dem Isotropen Kugelstrahler schon wieder **2,15 dB Gewinn** besitzt in Folge seiner Richtwirkung (Richtdiagramm = „Liegende Acht“) . Diese Gewinnangabe erfolgt dann in

dBd

und ihr Wert wird natürlich um diese **2,15 dB niedriger sein als der dBi-Wert.**

Der Gewinn unserer Patchantenne lässt sich nun mit Hilfe der am Anfang erwähnten „Friis“-Beziehung über einen recht einfachen Versuchsaufbau bestimmen:
Man arbeitet mit zwei identischen Antennen, wobei die eine sendet und die andere empfängt. Diese beiden Antennen werden in einem Abstand „d“ voneinander aufgestellt und sorgfältig aufeinander ausgerichtet, der bereits dem Betrieb im Fernfeld entspricht (also mehr als 10 Wellenlängen = 10 x λ beträgt). Sendet man nun mit einem genau bekannten Pegel, dann gilt für den an der Empfangsantenne messbaren Pegel

folgende Beziehung:

$$P_{Empfang} = P_{Sender} \bullet G_{Sendeantenne} \bullet G_{Empfangsantenne} \bullet \left(\frac{\lambda}{4\pi \bullet d} \right)^2$$

Wenn nun die Gewinne von Sende- und Empfangsantenne gleich sind (…weil man identische Antennen verwendet..) und man auf die logarithmische Darstellung in dB (mit den Leistungsangaben in dBm) umsteigt, wird das Ganze recht einfach:

$$G_{Antenne} = \frac{Emfangspegel - Senderpegel - 20 \bullet \log(\frac{\lambda}{4\pi \bullet d})}{2}$$ **in dB**

Deshalb als abschließender Hinweis für eigene Messungen: **bei einem solchen einfachen Patch liegt der Gewinn bei etwa 6,5 dBi.**

16.4. Entwurf einer WLAN-Patchantenne für lineare Polarisation

16.4.1. Patch-Dimensionierung mit qucsstudio

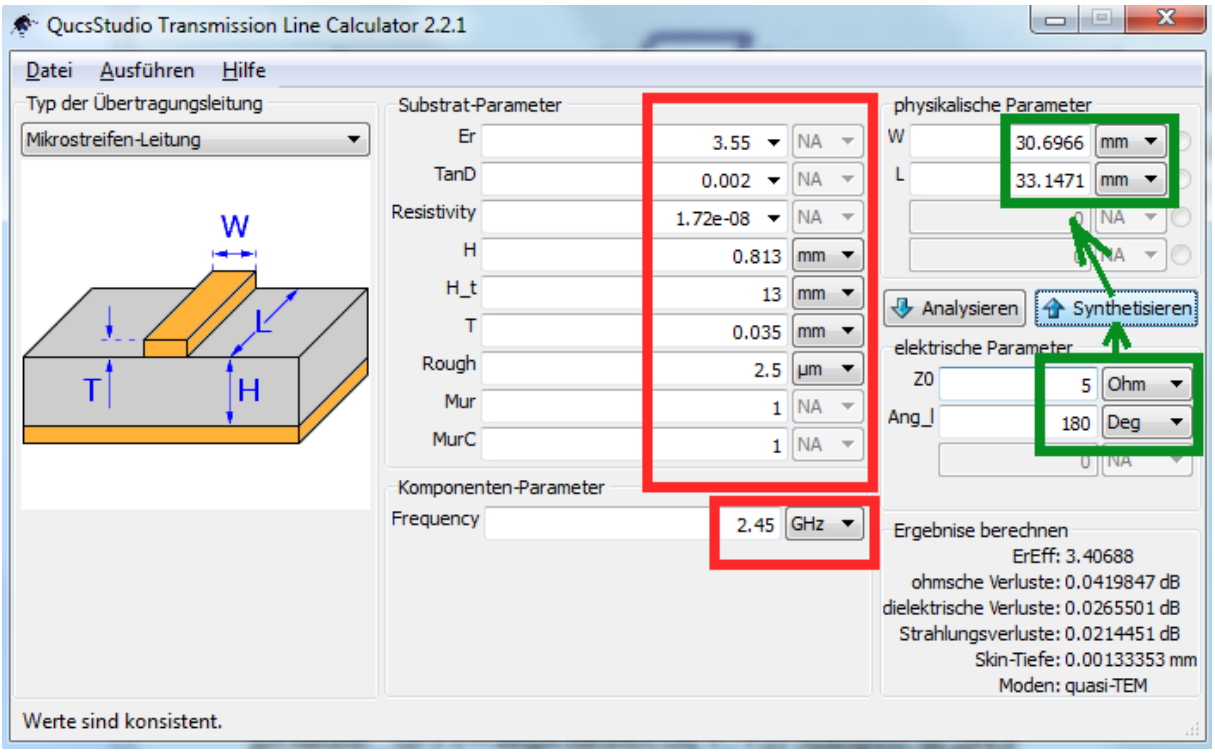
Wer die vorigen Einführungskapitel genau gelesen hat, der weiß, dass die **elektrische Länge der als Antenne verwendeten Microstrip-Leitung „L" = 180 Grad betragen muss.** Die **Patchbreite** bestimmt dabei die Eingangsimpedanz, die Güte und damit die Bandbreite (….**breiterer Patch gibt niedrigeren Strahlungswiderstand und damit kleinere Güte und damit größere Bandbreite**). Wir nehmen uns jedoch zum Einstieg einen **quadratischen Patch** vor.

Gegeben sei die als Standard verwendete RO4003-Platine (Dicke = 32 mil = 0,813 mm, ε_r = 3,55). Die Sendefrequenz ist 2450 MHz.

1. Schritt
Wir starten qucsstudio und legen ein neues Projekt „**Patchantenne**" an. Das anschließend sich automatisch öffnende Arbeitsblatt speichern wir als **„patch_01".**

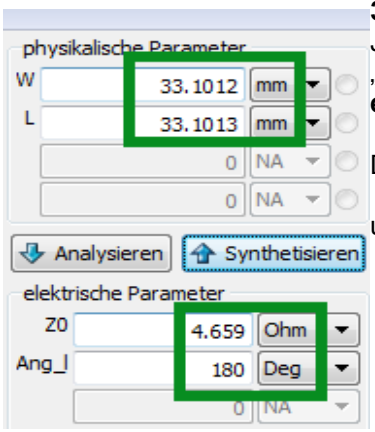
2. Schritt
Unter „**Werkzeuge**" finden wir „**Leitungsberechnung**" und stellen darin „**Mikrostreifen-Leitung**" ein.
In der linken Liste werden zunächst alle erforderlichen Platinendaten eingetragen:

Platinendicke = 32mil = 0,813 mm/ er = 3,55 / TAND = 0,002 / Kupferdicke = 35 µm / Rauigkeit = 2,5 µm



Die Betriebsfrequenz ist **2,45 GHz** und dort wünschen wir uns eine elektrische Leitungslänge von **180 Grad**. Der Patch soll quadratisch sein und deshalb gilt „**Länge = Breite**". Die Erfahrung zeigt, dass eine so breite Leitung einen Wellenwiderstand in der Größenordnung von **ca. 5 Ω** aufweisen wird. Also tragen wir diese Werte in das rechte untere (und grün markierte) Feld ein. Ein Klick auf „**Synthesisieren**" liefert uns im oberen grünen Feld die Werte

L = 33,15 mm und W = 30,7 mm



3. Schritt:
Jetzt wird abwechselnd die Breite etwas verändert und dann auf „Synthetisieren“ geklickt, **bis Patch-Länge und Patch-Breite für eine elektrische Länge von 180 Grad identisch sind.**

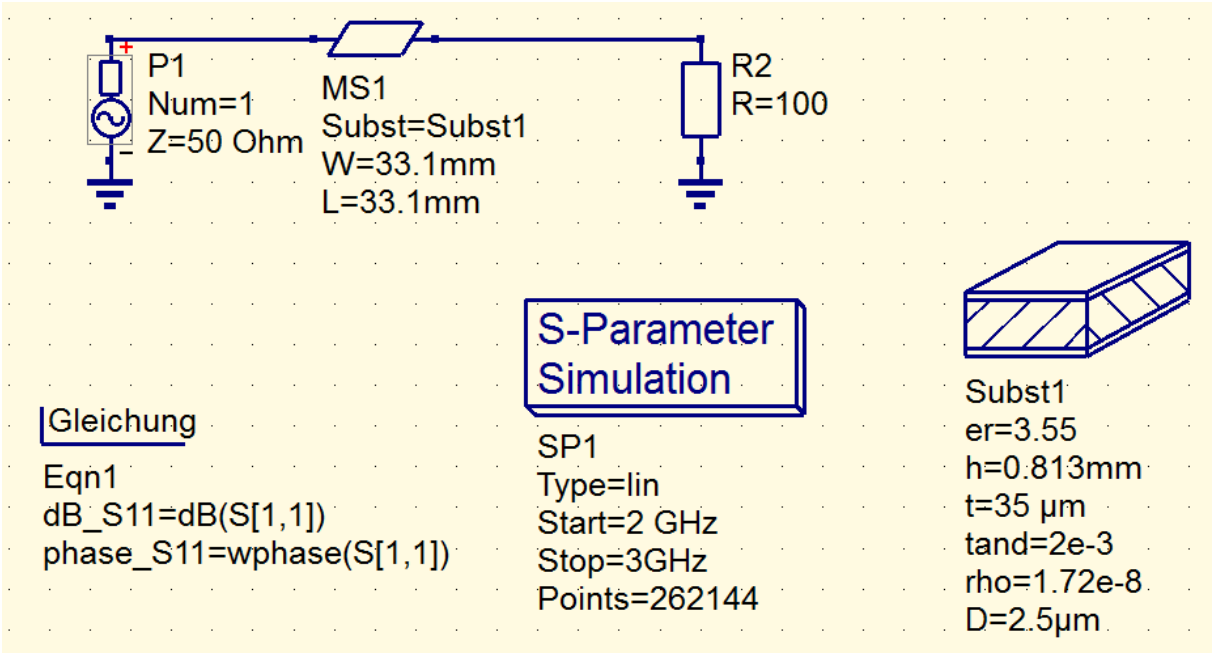
Das ist bei **L = W = 33,1 mm** der Fall

und dazu gehört **Z = 4,659 Ω**.

4. Schritt:
Nun testen wir dieses Ergebnis mit einer besonderen Eigenschaft dieses Leitungsstücks, dessen elektrische Länge genau eine halbe Wellenlänge beträgt:

Schließt man diese Leitung mit einem bestimmten Widerstand ab, dann erscheint nur exakt bei dieser Frequenz der Abschlusswiderstand als Eingangs-Impedanz – ohne irgendwelchen Blindanteile und Phasenverschiebungen!

Wählen wir einen Wert von 100 Ω als Abschluss:



Erläuterung:

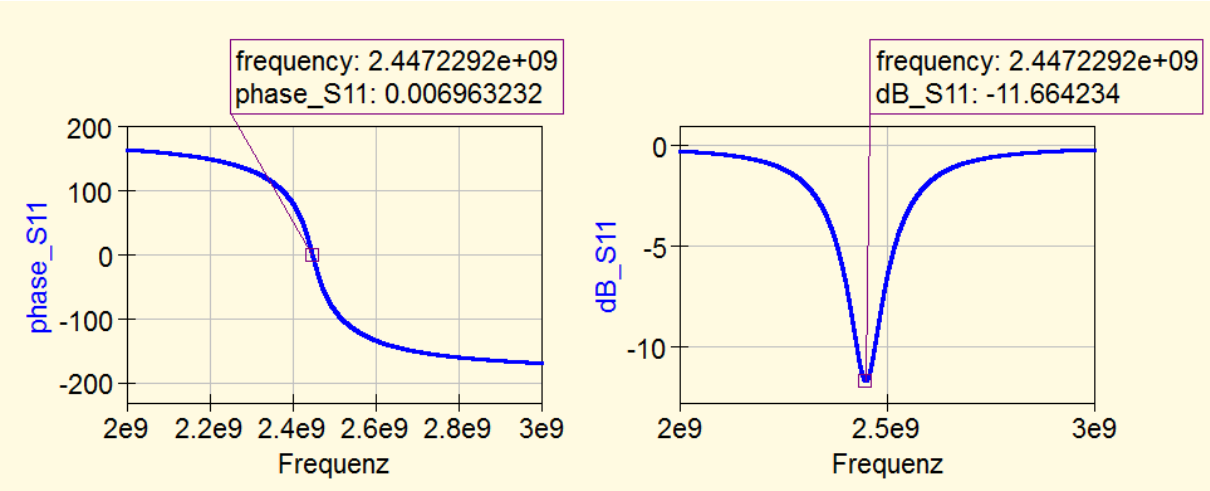
a) Die Microstrip-Leitung weist eine mechanische Länge von 33,1 mm und eine mechanische Breite von 33,1mm auf (...sie findet sich unter „Komponenten / Übertragungsleitungen / Mikrostreifen-Leitung“).

b) Die Substrat-Parameter hatten wir vorhin schon beim Leitungscalculator verwendet und deshalb kopieren wir diese Aufstellung in unser Projekt.

c) Wir formulieren die Gleichungen für Betrag und Phase von S11.

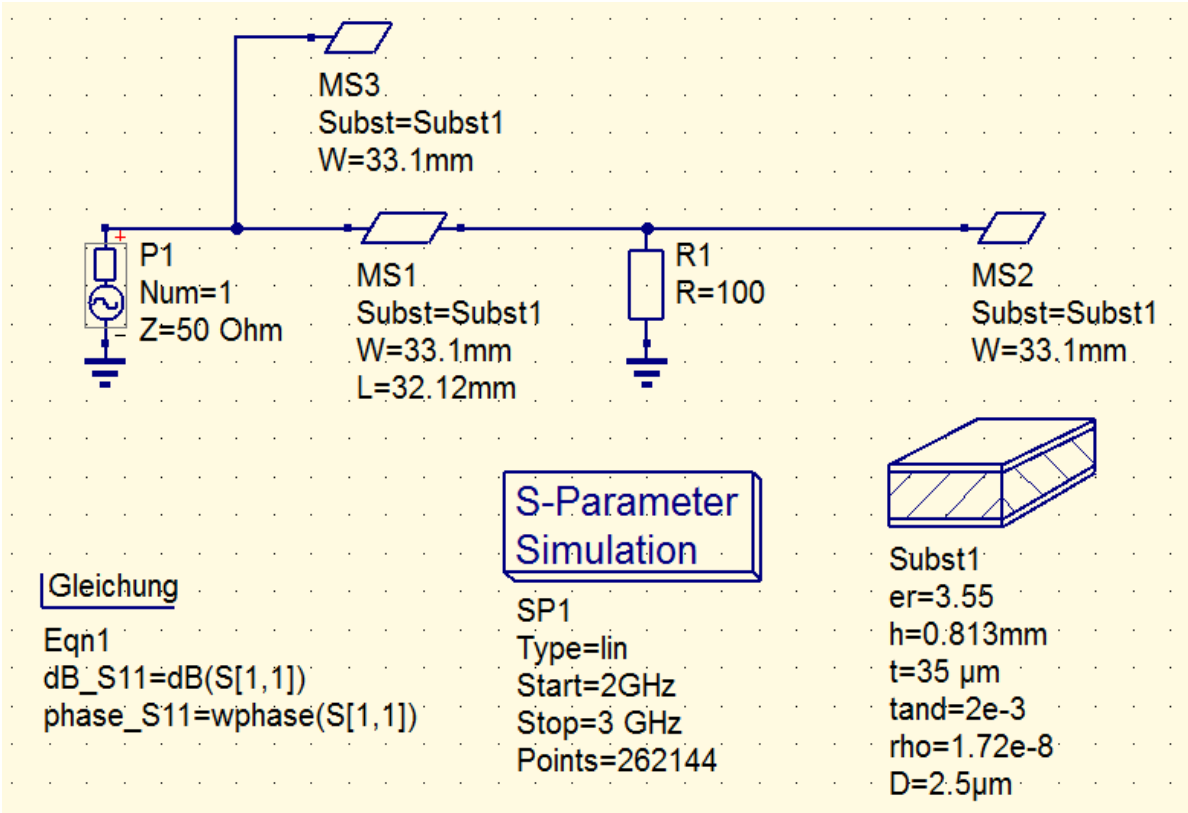
d) Wir simulieren linear von 2....3 GHz mit 262 144 Punkten.

Das Ergebnis sieht so aus:

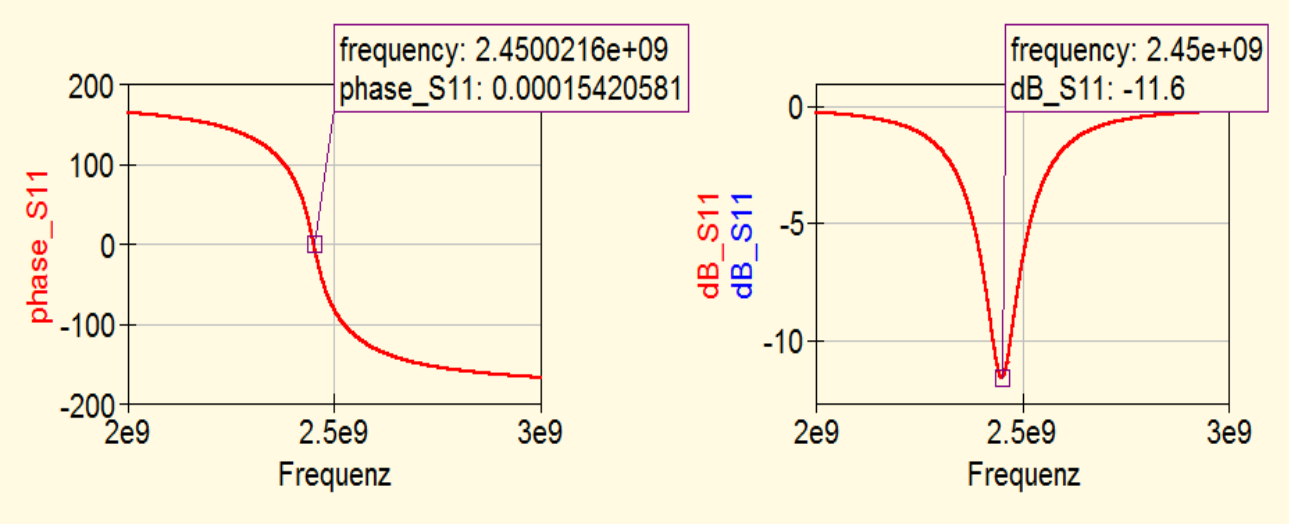


Die Frequenz für den „Nulldurchgang der Phase“ liegt nur knapp 3 MHz neben 2,45 GHz und das ist eine Abweichung von etwa 0,1%. Da wollen wir ganz zufrieden sein....

5. Schritt:
Nun folgt eine wichtige Sache, denn wir müssen noch die Sache mit den „**überstehenden Feldlinien“ (= Fringing)** berücksichtigen.



Dieses „Überstehen“ können wir durch ein Bauteil „**Mikrostreifen-Leitung Leerlauf**“ aus dem Komponenten-Vorrat „**Übertragungsleitungen“** an jedem Ende des Leitungsstückes berücksichtigen. Anschließend wird die Leitung solange verkürzt und erneut simuliert, bis wir wieder **den Nulldurchgang der Phase exakt bei 2,45 GHz sehen:**



Die neuen Daten für die Leitung und damit für die Patchantenne lauten damit:
Breite = 33,1 mm
Länge = 32,12 mm

Wenn man jedoch den Strahlungswiderstand der Antenne wissen will -- da muss „qucsstudio“ passen! Das ist dagegen ein leichter Fall für das kostenlose Sonnet Lite, denn dort muss man nur alle Verluste (im Leiterwerkstoff und im Dielektrikum und im Metall) auf Null setzen und nochmals simulieren. Was dann als Eingangswiderstand bei Resonanz übrig bleibt, muss der Strahlungs-widerstand sein und das Ergebnis lautet:

An der Patchkante messen wir R = 457 Ω

Jetzt muss man noch wissen, dass diese 457 Ω die **Parallelschaltung von zwei gleich großen Einzelwiderständen mit je 914 Ω** sind, denn wir haben zwei strahlende Schlitze vor uns und zwischen beiden Schlitzen befindet sich eine Lambda-Halbe-Leitung, die den Abschlusswiderstand nicht transformiert.

Also wirken an jeder Patchkante 914 Ω

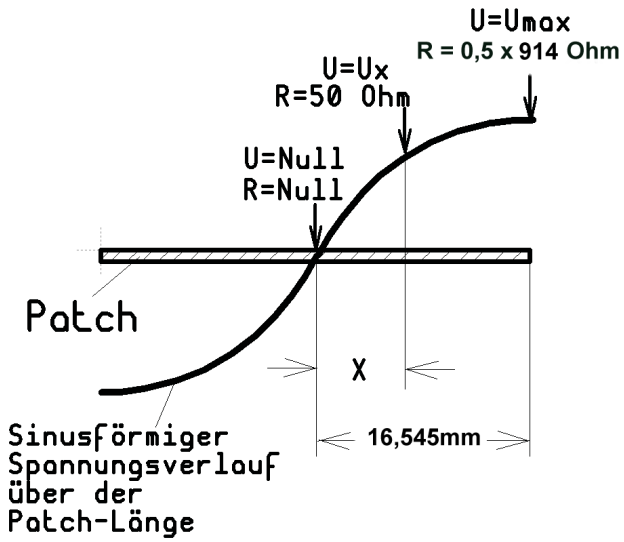
…und die müssen wir zusammen mit den Verlusten des Patchs irgendwie an 50 Ω anpassen….

16.4.2. Anpassung der Antenne an 50 Ω

Da haben wir prinzipiell zwei Möglichkeiten:

a) Anpassung mit **einer Lambda – Viertel – Transformationsleitung** oder

b) Speisung mit einem **„Coaxial Feed**“ von der Platinen-Unterseite her.



Wir wollen uns die Lösung b) vornehmen und die Antenne von der Unterseite her speisen.

Sehen wir uns dazu erst das benützte Anpassungsprinzip an und dazu gab es folgendes hübsches Bildchen zum Thema „Strom / Spannung / Impedanz“ bei einer Patchantenne“ im Grundlagen-Kapitel:

Wer genau hinsieht, der erkennt:

a) In **Patchmitte** ist die Spannung = Null, aber der Strom hat ein Maximum. Also haben wir hier die Impedanz „**Null Ohm“.**

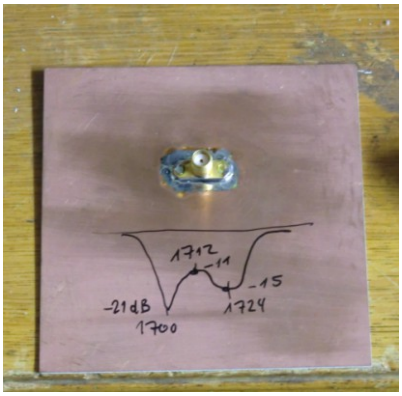
b) An jeder **strahlenden Patchkante** ist der Strom sehr klein, aber die Spannung hat dort ihr Maximum. Folglich ist an jedem Patchende die **Eingangs-Impedanz hoch** …bei einer idealen Antenne ist es die Parallelschaltung der beiden Strahlungswiderstände, also

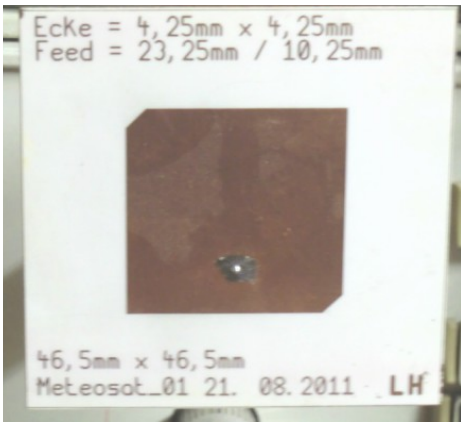
0,5 x 914 Ω = 457 Ω.

Logischerweise muss es dann zwischen diesen beiden Punkten eine Stelle geben, die eine Eingangs-Impedanz von exakt 50Ω besitzt!

Wenn wir dort ein winziges Loch bohren und diesen Punkt („von unten her“) mit dem Innenleiter einer SMA-Buchse verbinden, dann haben wir es geschafft. **Diese SMA-Buchse wird dann auf die kupferkaschierte Unterseite der Patchantenne aufgelötet.**

Die beiden Bilder einer für Meteosat-Empfang entwickelten Antenne sollen zeigen, wie das genau gemeint ist.



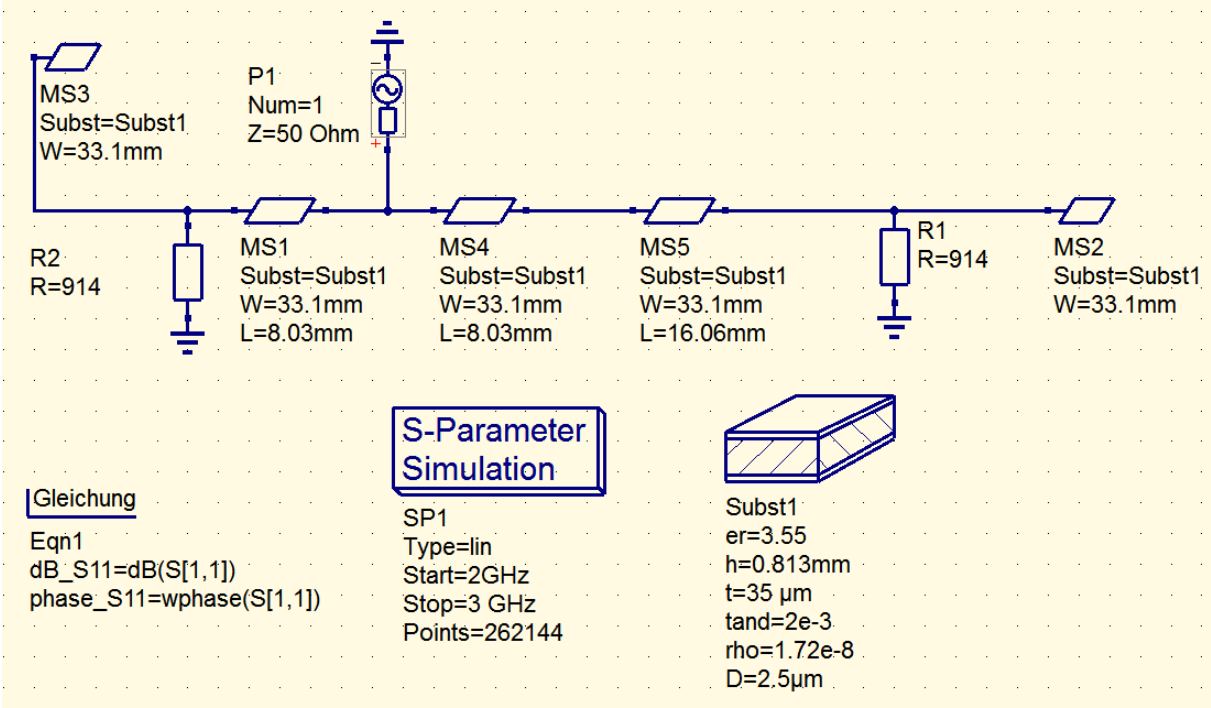


Diesen Feedpoint wollen wir nun in einer qucsstudio– Simulation finden.

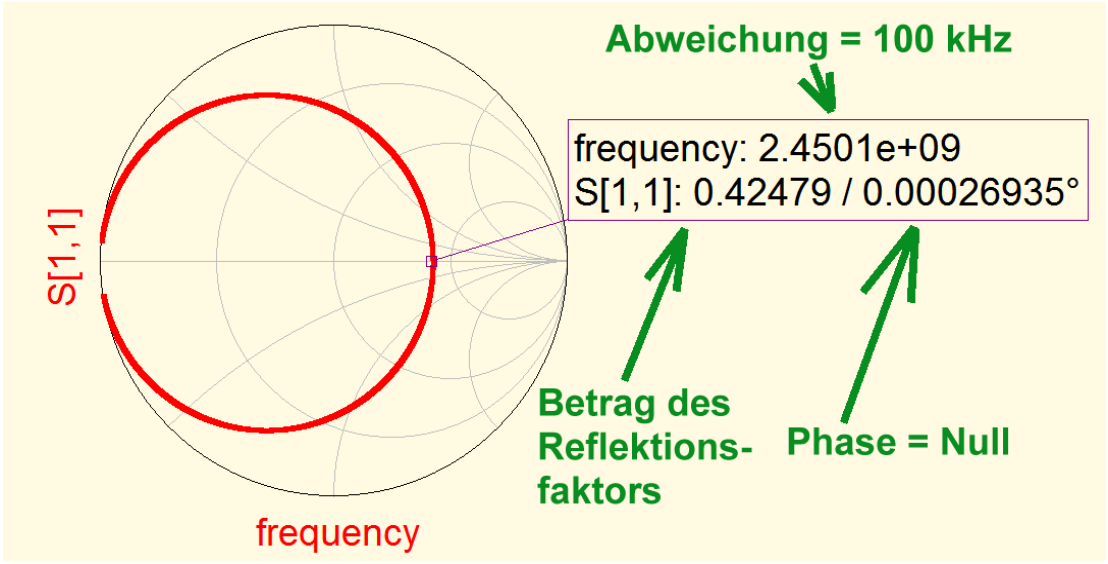
Wir teilen dazu die Antenne **in der Mitte** (bei Impedanz Null Ohm) und erhalten **zwei gleich lange Leitungsstücke mit derselben Breite von 33,1 mm** sowie identischen Längen (bei Berücksichtigung der Open End Extension = Fringing) mit

32,12 mm / 2 = 16,06 mm.

Die **linke Hälfte (von der Patchmitte bis zur linken Patchkante)** wird **nochmals in der Mitte geteilt** und an diesen Feedpoint **wird Port 1 angeschlossen**.
Das gibt zwei weitere Teilstücke mit je 16,06 mm / 2 = 8,03 mm.



Das S11-Ergebnis stellen wir im Smithchart dar und markieren die Resonanz (...= den Punkt, bei dem die Blindanteile verschwinden.



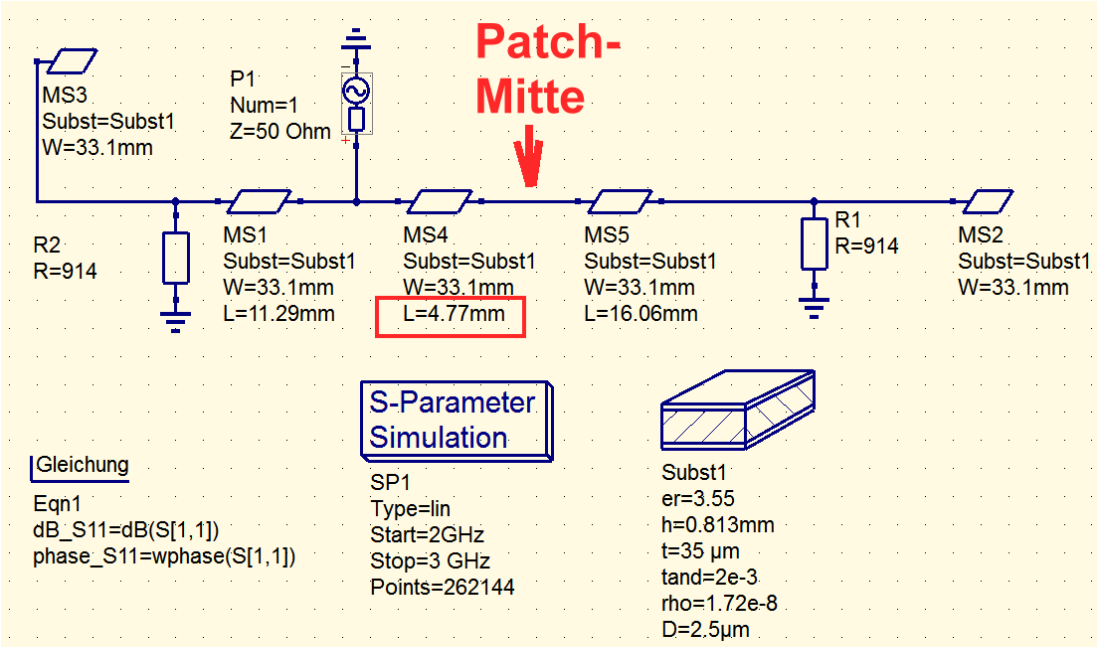
Perfekte Anpassung hätten wir, wenn der rote Kreis genau durch den Diagramm-Mittelpunkt verlaufen würde, aber:

man sieht, dass der **Eingangswiderstand Z_{in} noch viel zu groß ist** und wir das „Übersetzungsverhältnis des Leitungstransformators“ ändern müssen.

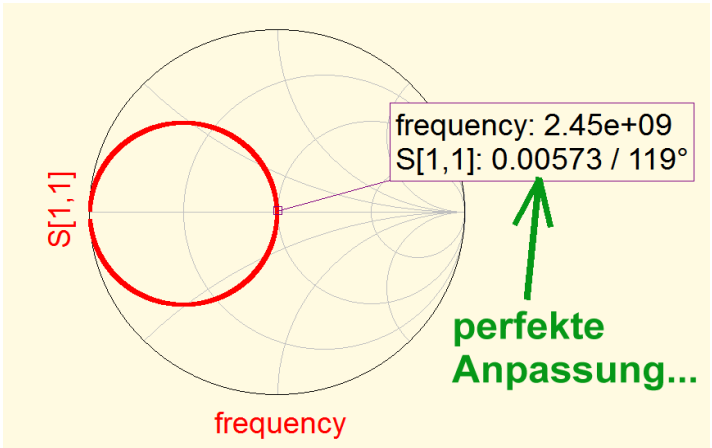
Dabei gelten folgende Spielregeln:

a) Verläuft die Kurve rechts vom Smithchart-Mittelpunkt, dann ist der Eingangswiderstand an Port 1 zu hoch. Dann muss das mittlere Leitungsstück verkleinert und das linke vergrößert werden, aber die Summe dieser beiden Längen muss weiterhin 16,06 mm betragen (= halbe Patchlänge).

b) Verläuft die Kurve links vom Smithchart-Mittelpunkt, dann ist der Eingangswiderstand an Port 1 zu klein. Dann muss das mittlere Leitungsstück vergrößert und das linke verkleinert werden, aber die Summe dieser beiden Längen muss weiterhin 16,06 mm betragen (= halbe Patchlänge).

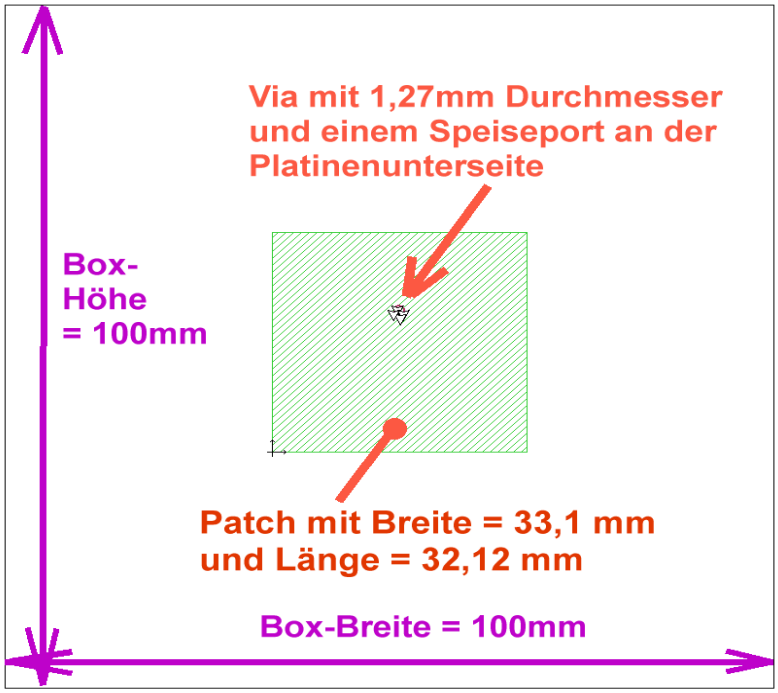


So sieht das Endergebnis aus und mit einem Abstand des Feedpoints von **4,77mm** (gemessen von der Patchmitte nach links) sind wir am Ziel.



Dazu gehören nun folgende Daten für den Platinenentwurf:

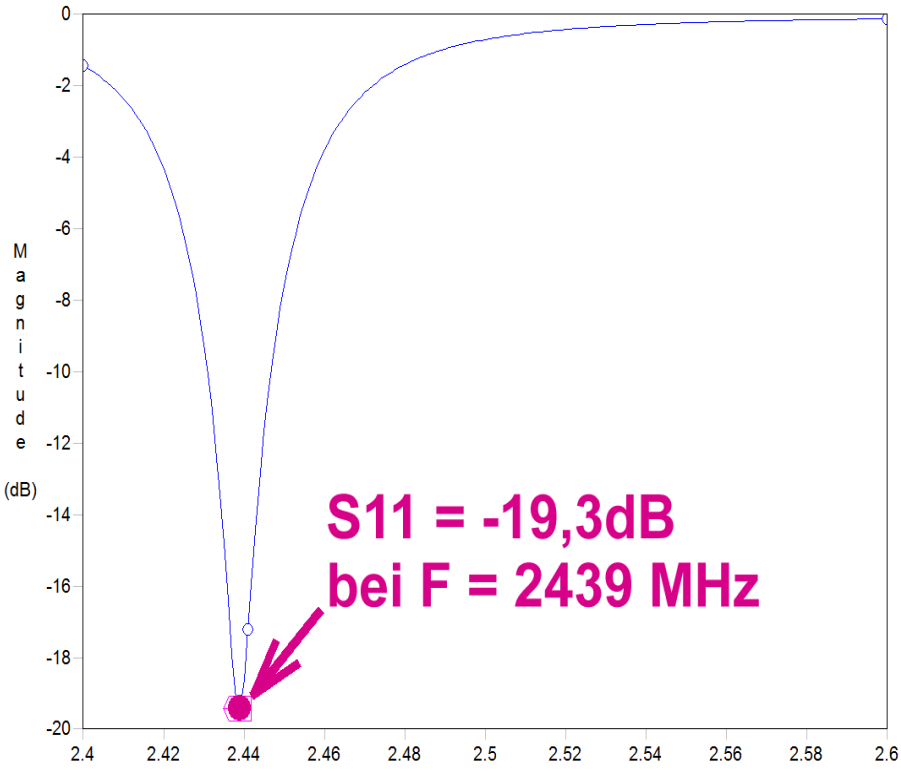
Breite = **33,1mm**
Länge = **32,12mm**
Feedpoint = **4,77mm links von der elektrischen Mitte der Patchlänge.**



Vom RECHTEN RAND aus gemessen ist das
(32,12mm / 2) +4,77mm = **20,83mm**

Man möchte ja nun schon wissen, ob das alles so stimmt und so sein kann. Der exakteste Weg dazu ist natürlich die Herstellung und Vermessung eines Prototyps --- aber das ist auch der teuerste und aufwendigste, und er kann dauern.
Also wurde nochmals die kostenlose EM-Software „**Sonnet Lite**“ aus dem Internet angeworfen und die komplette Antenne mit allen Verlusten simuliert.
Einschließlich des Innenleiters der SMA-Buchse -- er wird als „**Via mit 1,27mm Durchmesser**“ berücksichtigt!

Das Ergebnis ist höchst erfreulich:

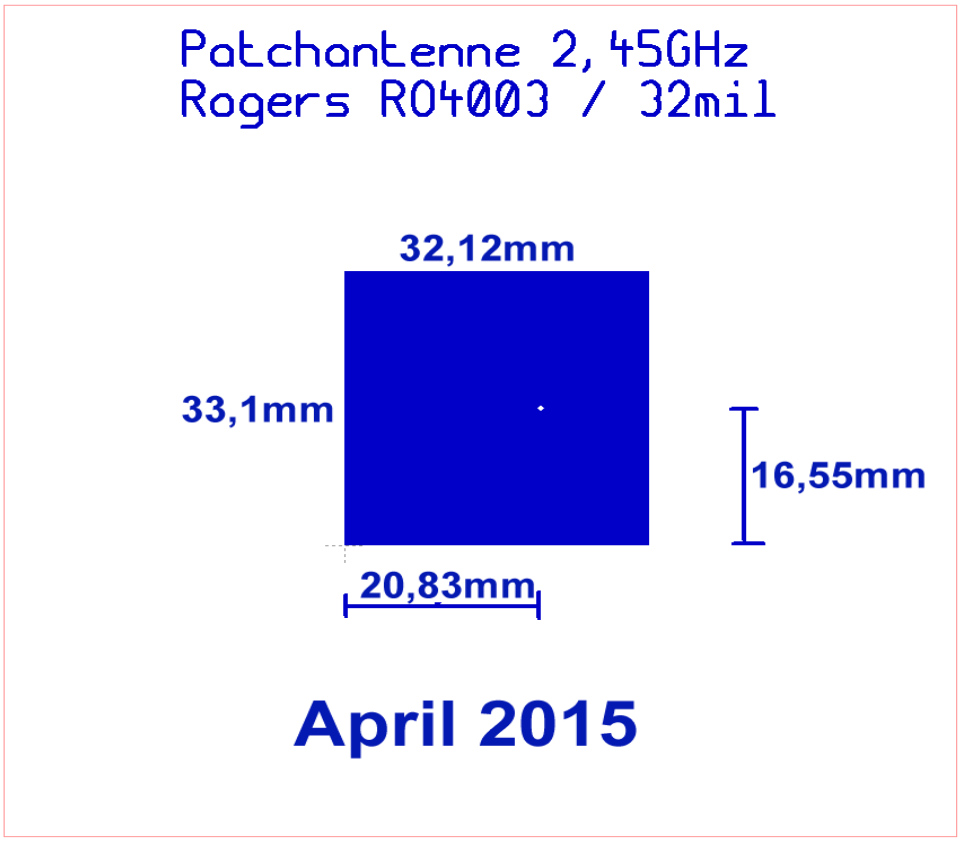


Das Via stellt eine Induktivität dar und **erniedrigt deshalb die Resonanzfrequenz**. So liegt sie nun bei 2439 statt bei 2450 MHz, aber die **Magnitude von S11 beträgt bereits knapp -20dB.**

Das passt doch gut zu unserer Arbeit und zu den Überlegungen – und viel genauer kriegt man es mit der Simulation alleine nicht hin.

Deshalb ist es Zeit für einen ersten Prototypen und entsprechende Kontrollmessungen.

Dieses File des Platinen-CAD-Programms geht nun an den Platinenmacher (….viele von ihnen wünschen sich jedoch gleich einen „**Gerber-Plot**“, mit dem die Belichtungsmaschine gefüttert wird).



Die Außenmaße der Platine betragen

100mm x 100mm

(um damit dem Patch eine unendlich große Massefläche vorzutäuschen)

Messergebnis bei diesem Prototyp:
Resonanz bei 2443MHz,
S11 besser als -30dB.

Na also..

17. Ein Ausflug in die Modulation

17.1. Einstieg

Vor ca. 100 Jahren hat man bei der „Drahtlosen Telegrafie“ mit der Morsetaste angefangen und dann folgte die drahtlose Übertragung von Sprache und Musik. Dabei galt es, folgendes Problem zu lösen:

Audiosignale haben eine viel zu große Wellenlänge, um sie über Antennen mit erträglichem Wirkungsgrad drahtlos abstrahlen zu können. Es gilt nämlich:

Die optimale Antennenlänge beträgt ein Viertel der Wellenlänge und unter einer Antennengröße von Lambda / 10 kriegt man immer weniger in die Luft.

Beispiel: ein 1 kHz-Ton hat eine Wellenlänge von 300 km. Also müsste die Antenne zwischen 30 km und 75 km lang sein…
Deshalb wählt man folgenden Weg: **Man nimmt einen Lastesel (= Trägersignal mit hoher Frequenz, damit die Antenne klein wird), dem man die Last (= Information) problemlos für den Transport aufbürden kann.** Dieser Vorgang heißt „**Modulation**“.

So entwickelten sich im Laufe der Zeit im Prinzip drei verschiedene Modulationsarten, deren Unterschiede dieses Bild zeigt (..es stammt aus einer Application Note der Firma National Instruments):

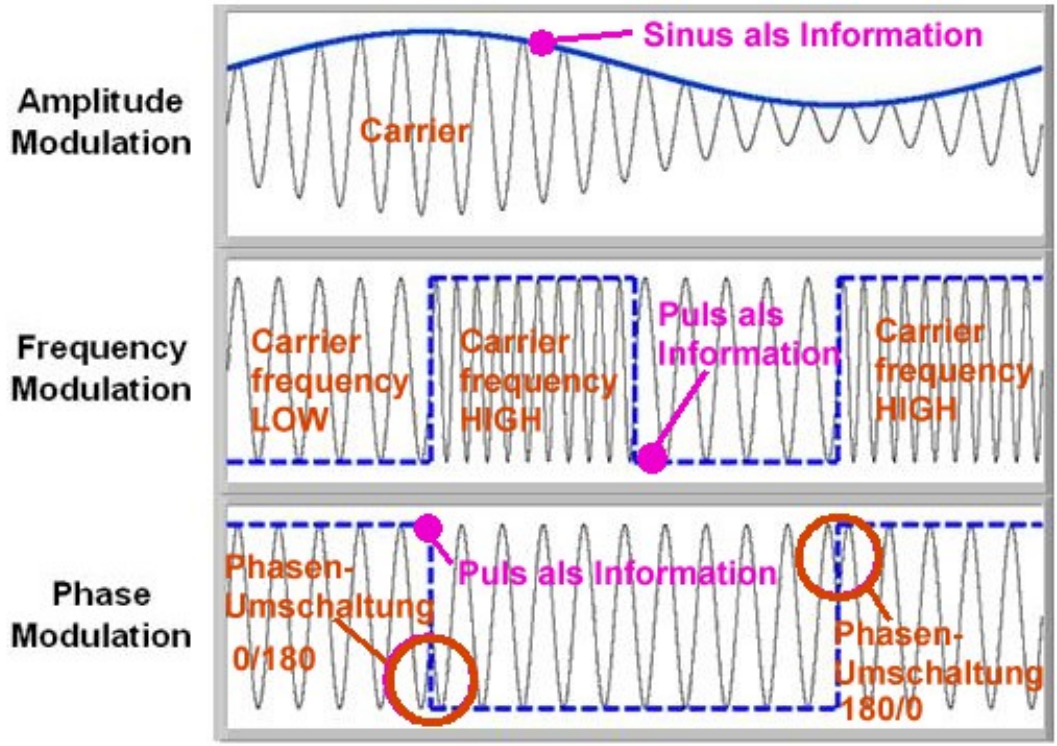


Figure 5. Time Domain of AM, FM, and PM Signals

Bei der **Amplitudenmodulation** ändert sich die **Amplitude des Trägers** im Rhythmus der Information.

Bei der **Frequenzmodulation** ändert sich die **„Augenblicksfrequenz“des Trägers** im Rhythmus der Information.

Bei der **Phasenmodulation** ändert sich die **aktuelle Phase des Trägersignals** im Rhythmus der Information.

Durch die vielen modernen Entwicklungen mit den Fortschritten in der digitalen Übertragung wuchs diese einfache Familie in unglaublichem Maß an.
Die folgende kurze Übersicht (aus Wikipedia) soll zeigen, was alles möglich und zu finden ist.

Hinweise zu AM, FM und PM **(aus Wikipedia):**

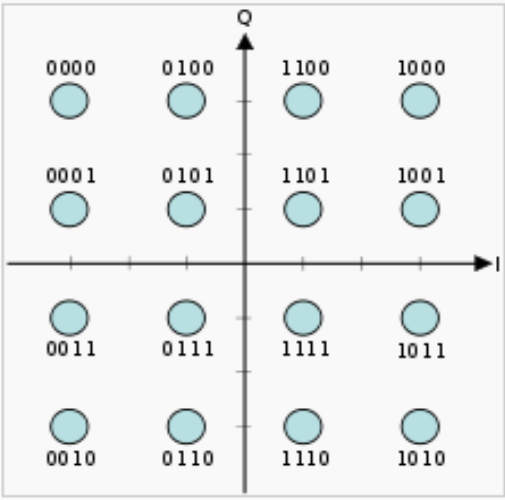
See also [edit]

- Amplitude and phase-shift keying or Asymmetric phase-shift keying (APSK)
- Carrierless Amplitude Phase Modulation (CAP)
- In-phase and quadrature components
- Modulation for other examples of modulation techniques
- Phase-shift keying
- QAM tuner for HDTV
- Random modulation

See also [edit]

- Amplitude-shift keying (ASK)
- Continuous-phase frequency-shift keying (CPFSK)
- Dual-tone multi-frequency (DTMF), another encoding technique representing data by pairs of audio frequencies
- Frequency-change signaling
- Multiple frequency-shift keying (MFSK)
- Orthogonal frequency division multiplexing (OFDM)
- Phase-shift keying (PSK)
- Federal Standard 1037C
- MIL-STD-188

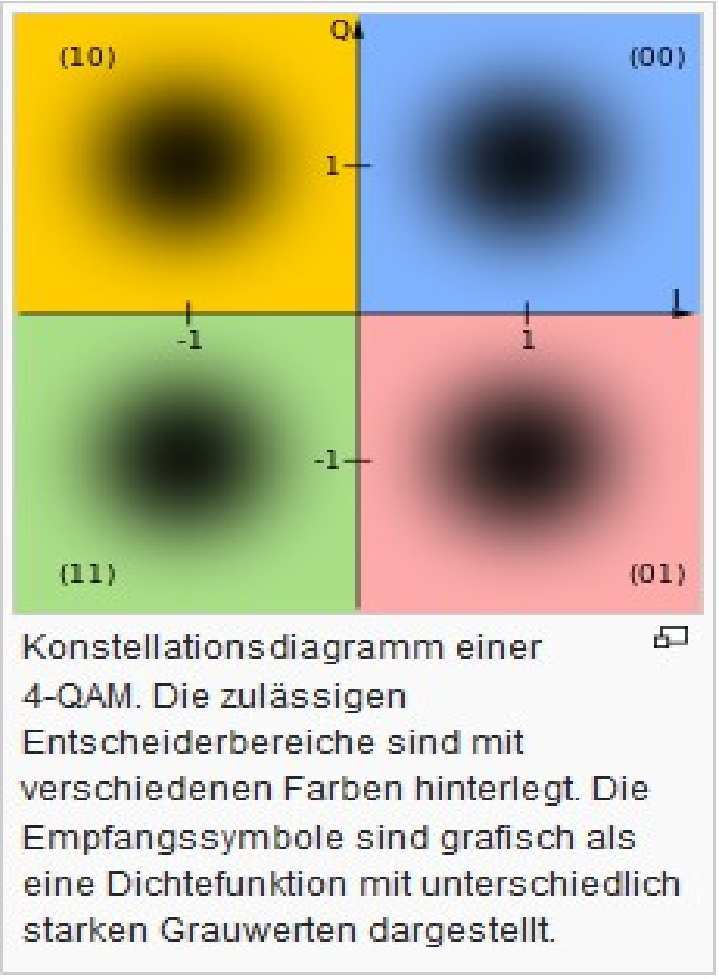
QAM



16-QAM mit Gray-Code. Jedes Symbol unterscheidet sich vom nächstgelegenen Nachbarsymbol in nur einer Bitstelle.

Heute kombiniert man Amplituden- und Phasenmodulation oft zur „**QAM**“-Technik.
Dabei arbeitet man mit einem orthogonalen Achsenkreuz. Die waagrechte Achse wird als **„I“ - Achse** (= In Phase Axis) bezeichnet, die senkrechte heißt **„Q“-Achse** (= Quadrature Axis).
Nun wird die „Zeigerlänge bis zu einem Punkt im Diagramm“ und der zugehörige Phasenwinkel zwischen Zeiger und „I“-Achse als Kriterium für einen übertragenen Augenblickswert (analog oder digital!) verwendet.

Das nebenstehende Bild zeigt die Zuordnung eines Punktes im Diagramm zur übertragenen Bit-Kombination bei einer 16-QAM. Von Punkt zu Punkt darf sich aber nur eine Bitstelle ändern.



Konstellationsdiagramm einer 4-QAM. Die zulässigen Entscheiderbereiche sind mit verschiedenen Farben hinterlegt. Die Empfangssymbole sind grafisch als eine Dichtefunktion mit unterschiedlich starken Grauwerten dargestellt.

Dieses Bild zeigt eine **4-QAM** und soll auf die Probleme beim Empfang und der Decodierung hinweisen.

Wir wollen uns aber in diesem Projekt speziell um die Grundlagen der einzelnen Modulationsarten und die Möglichkeiten zur Simulation kümmern.

Wer tiefer einsteigen will: Siehe oben, bei Wikipedia...

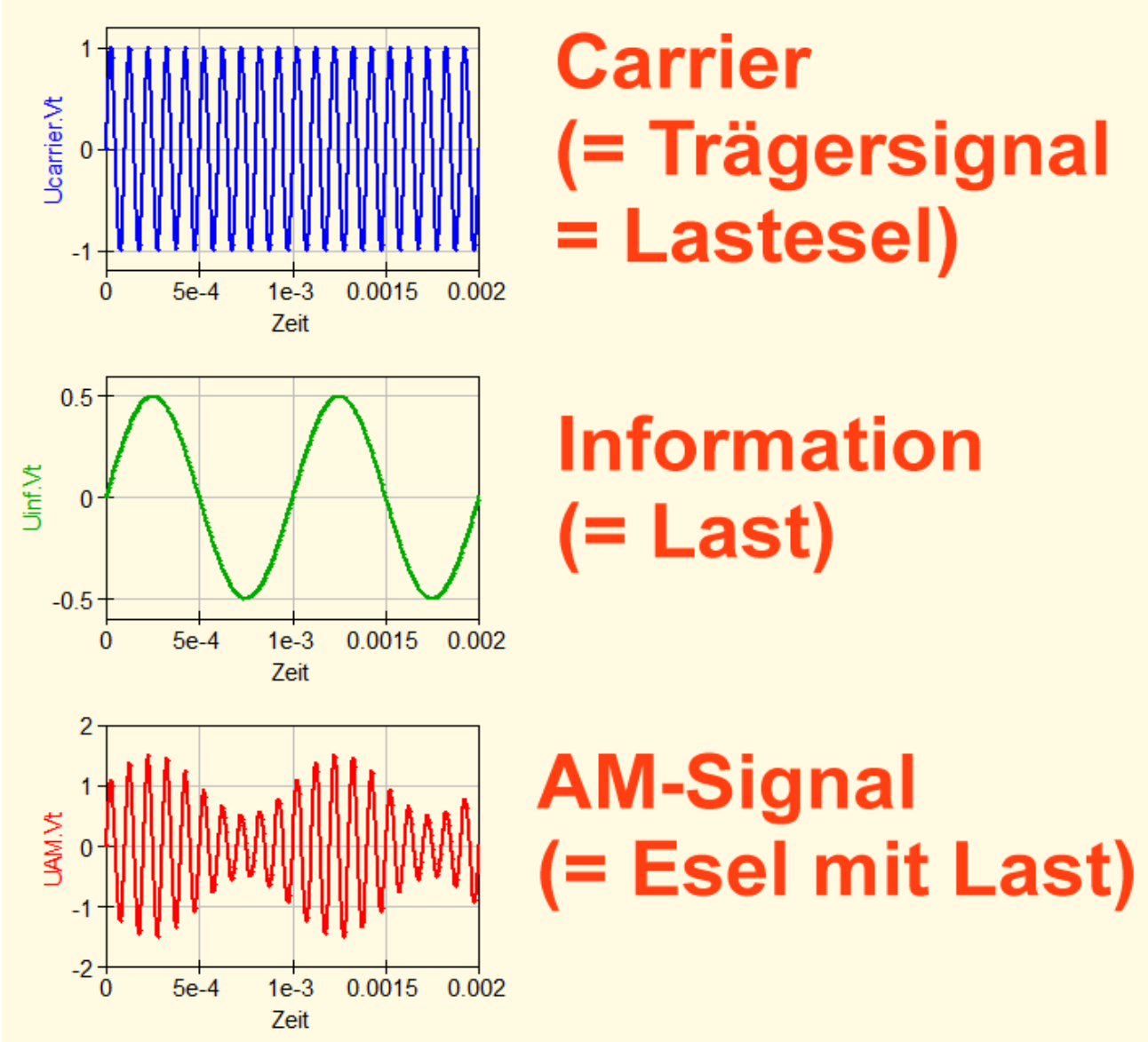
17.2. Die Amplitudenmodulation

17.2.1. Hintergrund

Das Prinzip ist sehr einfach:

Die Amplitude des Trägers wird im Rhythmus der Information geändert.

Nehmen wir als Beispiel eine sinusförmige Information mit f = 1 kHz und einen Träger mit 10 kHz:



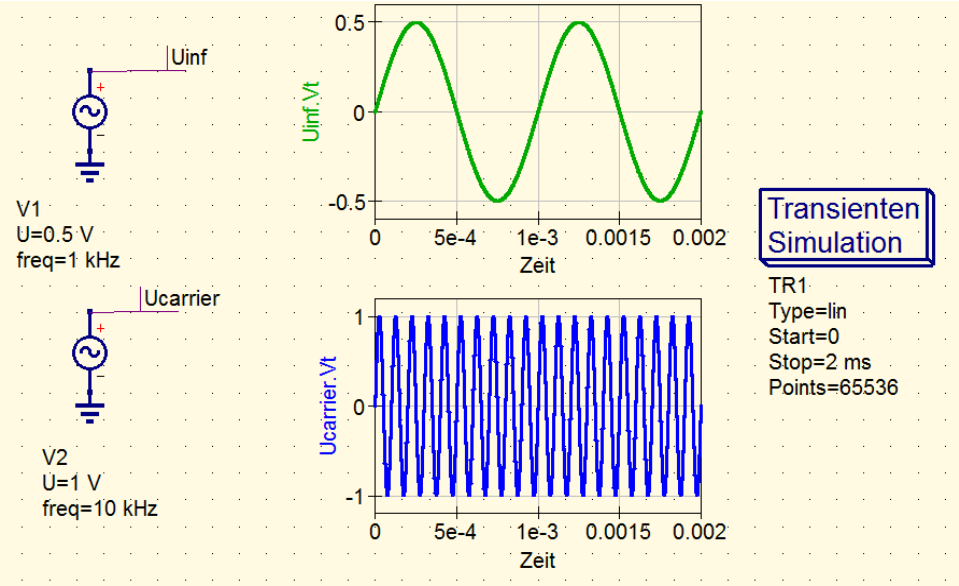
Mathematisch gesehen werden hier nacheinander

a) eine Multiplikation von zwei verschiedenen Signalen und

b) eine zusätzliche Addition durchgeführt.

Das wollen wir uns näher ansehen.

Zuerst holen und programmieren wir uns (aus Komponenten / Quellen) zwei Sinusquellen für die Information und das Trägersignal und simulieren ihren Verlauf mit den angegebenen Einstellungen:



Ganz oben ist die Information als Sinus mit 1000 Hz und 0,5 V Spitzenwert zu sehen.

Dann folgt die konstante Trägerspannung mit f = 10 kHz und einem Spitzenwert von 1 V.

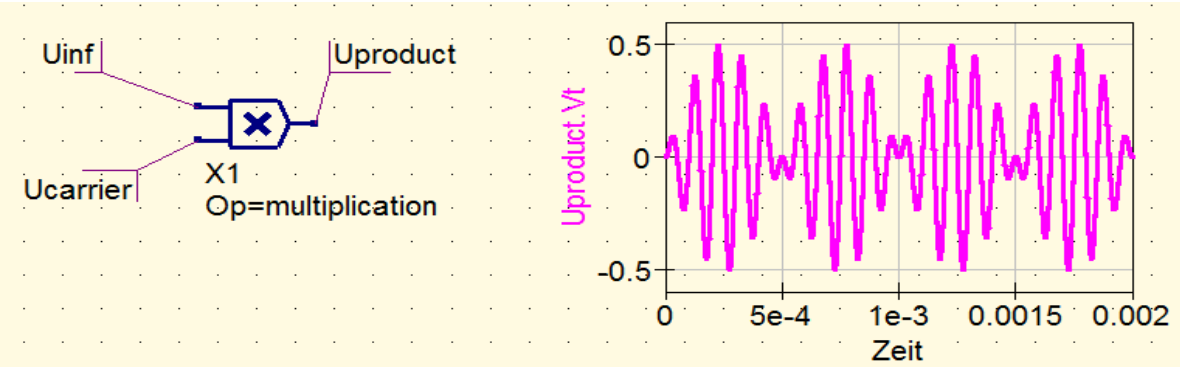
Wenn wir diese beiden Signale miteinander ***multiplizieren***, dann muss das Ergebnis ein Signal mit Trägerfrequenz sein, das

d) dem Amplitudenverlauf der Information folgt, aber

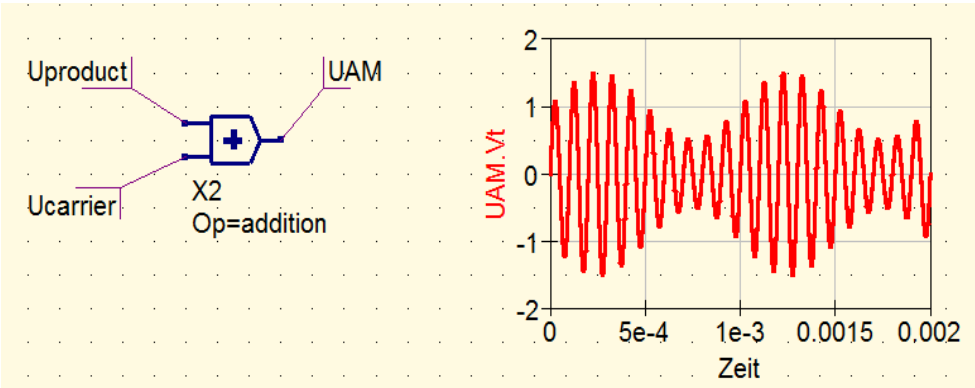
e) jedes Mal die Phase umkehrt, wenn die Informationsspannung negativ ist.

Dieses Ergebnis trägt den speziellen Namen „***Modulationsprodukt***“

Zur Simulation brauchen wir den Baustein „**Operation**“ aus **„Komponenten / Systemkomponenten“,** öffnen seine Eigenschaften und stellen auf „multiplication“ um.



Zu diesem Modulationsprodukt addieren wir die unmodulierte Trägerssspannung und dazu dient **ein weiterer Baustein „Operation“, der aber diesmal auf „addition“** eingestellt wird:



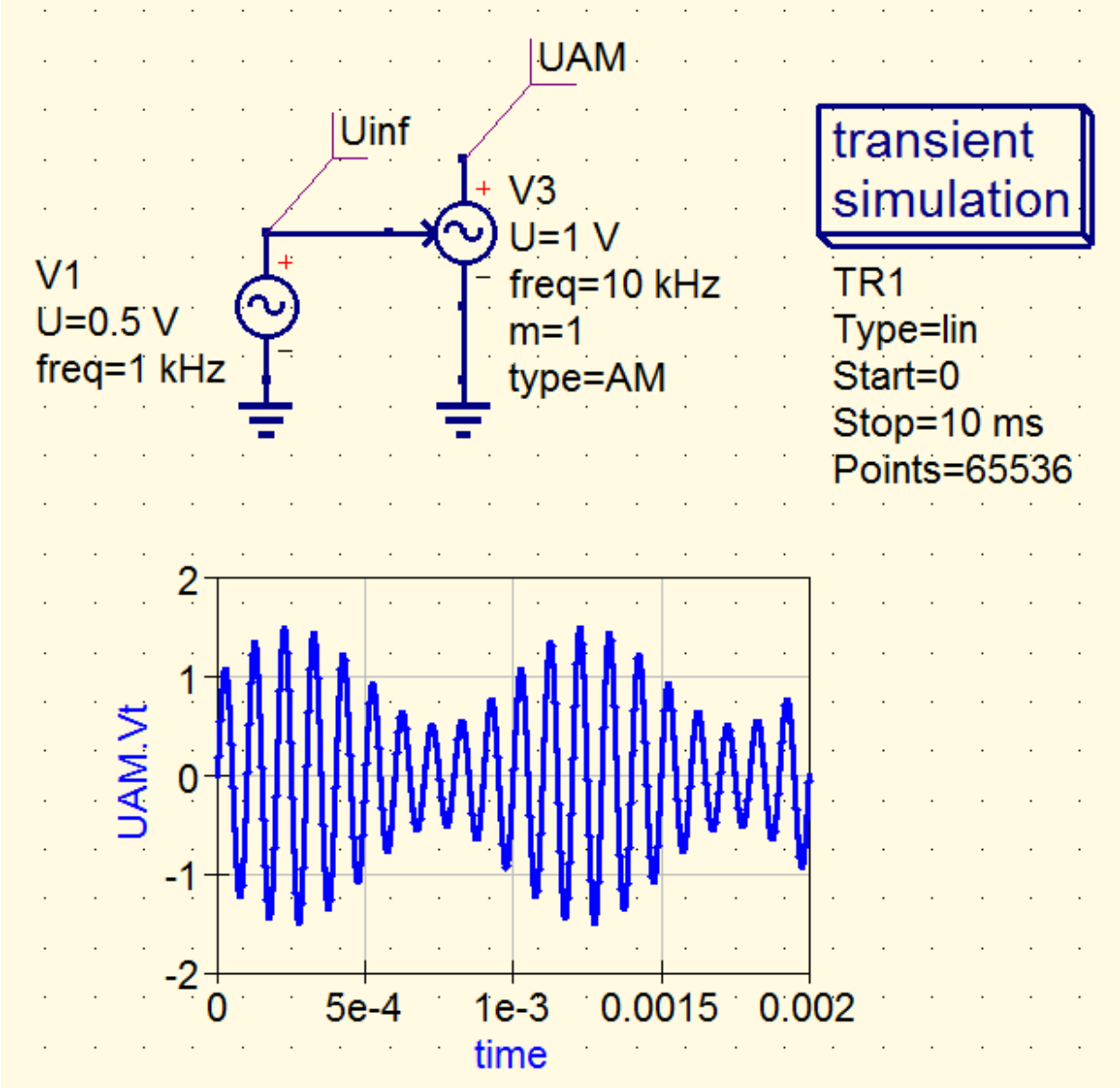
Schauen wir uns das AM-Signal mit seinem zeitlichen Verlauf etwas genauer an:

Wenn wir das Modulationsprodukt zum **unmodulierten Trägersignal addieren**, wird die **Gesamtspannung größer, wenn das Modulationsprodukt gleichphasig mit dem Träger** ist.
Ebenso erhalten wir **ein kleineres Gesamtsignal in denjenigen Abschnitten, wo Modulationsprodukt und Träger gegenphasig** zueinander sind.

Die Amplitude der Gesamtspannung im untersten Diagramm -- also unser AM-Signal -- schwankt nun wie gefordert im Rhythmus der Information.

Unser qucsstudio stellt für diese Aufgabe eine spezielle Spannungsquelle bereit („AM-modulierte Quelle“ in „Komponenten / Quellen“), die alle diese Aufgaben übernimmt und nach der Eingabe von Informationssignal und Trägerfrequenz den **Modulation-Index „m“ wissen möchte. Der steuert den AM-Modulationsgrad und verstärkt oder schwächt das Informationssignal, bevor damit moduliert wird. Deshalb setzen wir ihn hier einfach auf m = 1.**
Der endgültige AM – Modulationsgrad ist schließlich das Verhältnis der Spitzenwerte von Information und unmoduliertem Träger im Ausgangssignal und er beträgt in unserem Beispiel

AM - Modulationsgrad = Uinf / Ucarrier = 0,5V / 1V = 0,5



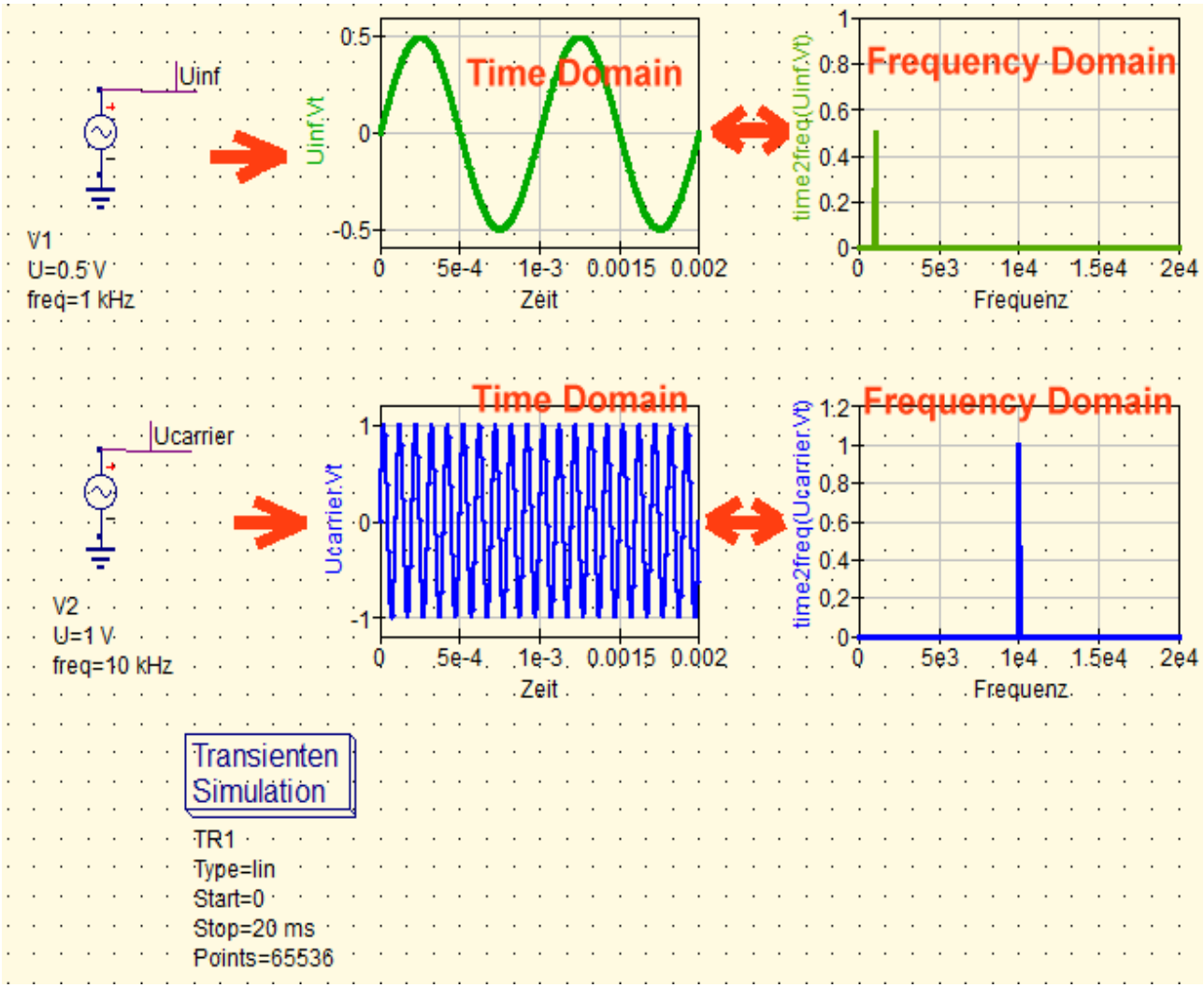
17.2.2. Spektrum des AM-Signals

Da sollte man sich fragen, wie viele unterschiedliche Frequenzen (bei sinusförmiger Information) hier im Spiel sind -- schließlich handelt es sich bei UAM nicht mehr um einen konstanten Sinus! Beim Träger und bei der Information ist es in unserem Beispiel einfach, denn das sind nun wirklich „konstante Sinus-Spannungen“ mit je nur einer Linie im Spektrum. Beim Modulationsprodukt müssen wir die Mathematik bemühen, denn da gibt es eine hübsche Formel für die Multiplikation von zwei Cosinus-Signalen:

$$\cos(\alpha) \bullet \cos(\beta) = \frac{1}{2}[\cos(\alpha+\beta)+\cos(\alpha-\beta)]$$

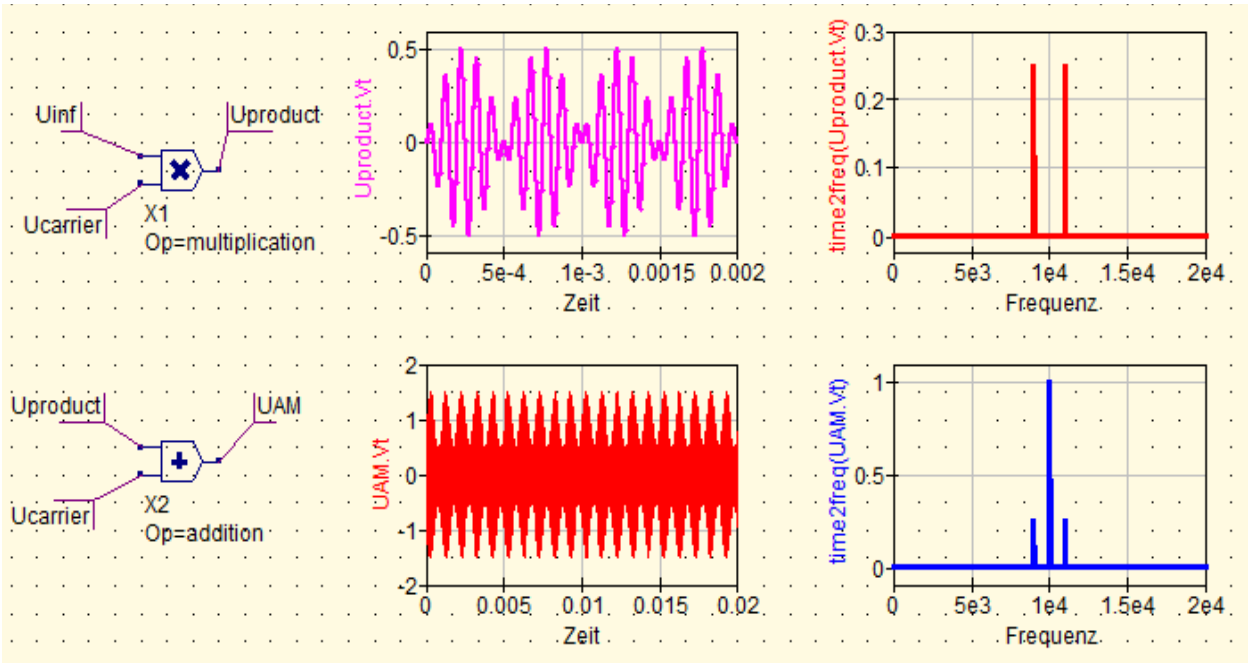
Wir erhalten als Modulations-Produkt also die Summen- und die Differenzfrequenz der beiden beteiligten Cosinussignale. Träger und Information selbst tauchen NICHT MEHR darin auf!

Das muss auch die Simulation zeigen und die beginnen wir mit der Information und dem Träger:



Wie erwartet, finden wir die beiden Signale als einzelne Linien mit den korrekten Amplituden in den simulierten Spektren bei den Frequenzen f = 500 Hz und f = = 10 000 Hz

Nun folgen die beiden Aktionen „Multiplikation“ und anschließend „Addition“ mit der Simulation der Spektren:



Beim **Modulationsprodukt** finden wir, wie erwartet und vorausgesagt, die **Summenfrequenz und die Differenzfrequenz** der beiden zugeführten Signale. Ihre **Amplituden betragen jeweils 0,25 V** und damit wird die Theorie bestätigt: sie sagt eine Reduktion auf die Hälfte der Eingangsamplitude der Information (0,5 V) voraus. Sie werden als **untere und obere Seitenfrequenz bezeichnet.**

Wird nun noch der **unmodulierte Träger mit 1V und 10 kHz dazuaddiert,** dann erhalten wir das komplette U_{AM}-Signal mit seinen drei Spektrallinien.

Hinweis:
Besteht die Information aus vielen Spektrallinien (Beispiel: Sprache, Musik usw.), dann spricht man beim AM-Signal nicht mehr von Seitenlinien, sondern von einem „**unteren und oberen Seitenband**“ = **lower and upper sideband** = **LSB und USB“** .

Bitte beachten:
Beim **unteren Seitenband LSB ist die „Frequenzskala gespiegelt“.**..darauf sollte man achten! Man spricht hier von „**Kehrlage**“.

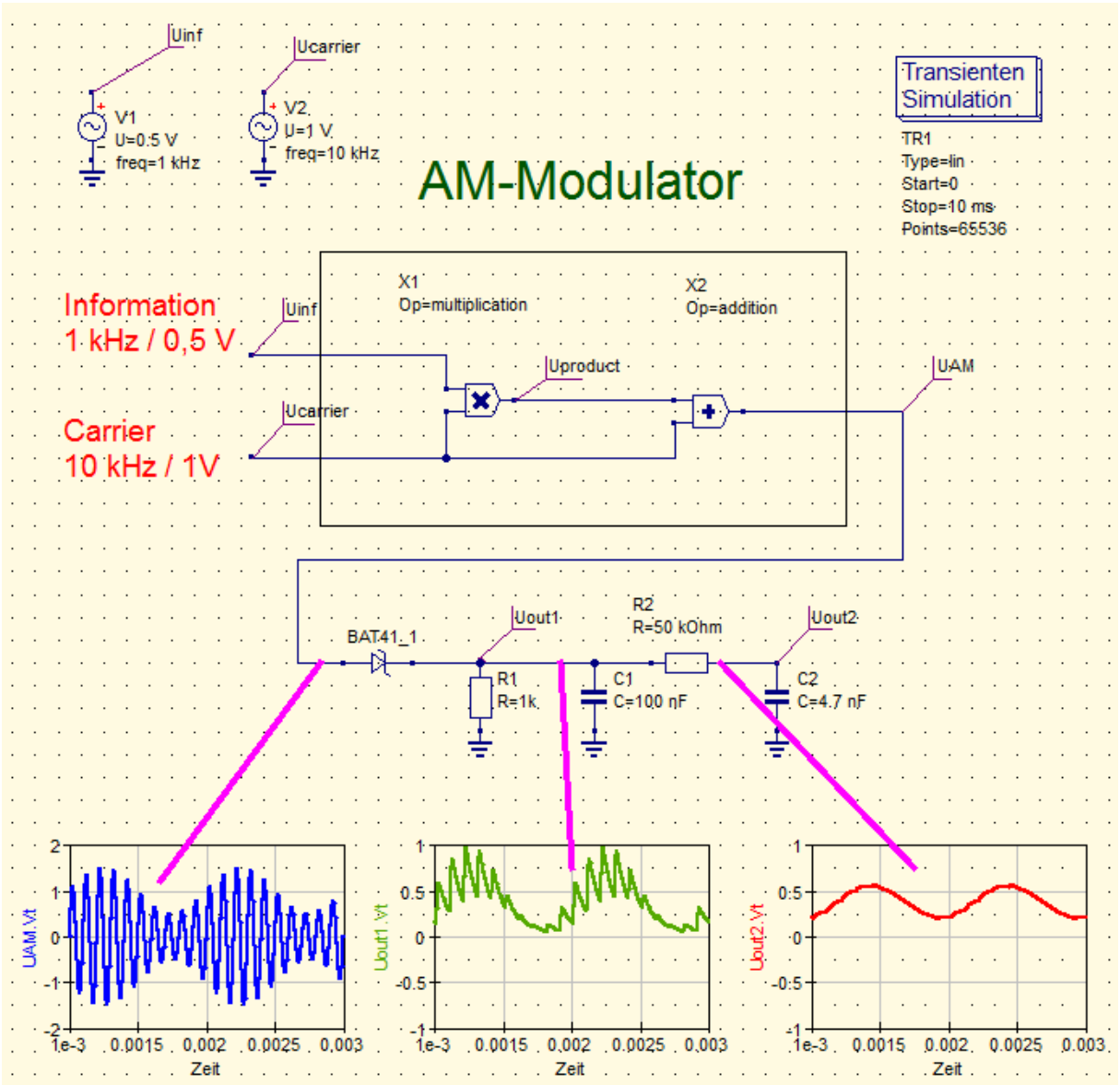
Außerdem gilt:
Beide Seitenbänder enthalten exakt dieselbe Information (..aber wie schon erwähnt, jeweils nur mit halber Informationsamplitude), der Träger selbst dagegen überhaupt keine Information. Da kann man, um Bandbreite oder Sendeleistung zu sparen, auf alle Ideen kommen...

17.2.3. Demodulation eines AM-Signals

17.2.3.1. Der klassische AM - Diodendemodulator

Dazu benötigen wir nochmals die vorige Simulations-Schaltung, mit der wir das „Zweiseitenband – AM – Signal“ erzeugt haben. Wir arbeiten weiterhin mit der sinusförmigen **Informations-Spannung von 1 kHz (Spitzenwert: 0,5V) und dem Trägersignal von 10 kHz / 1V. Dann geht es los:**
Wir speisen mit dem AM-Signal UAM eine **Einpuls-Gleichrichterschaltung** (= Schottky-Diode BAT41 mit nur 0,25V Schwellspannung) und einem Lastwiderstand mit 1 kΩ). Dem Lastwiderstand ist ein „Glättungs-Kondensator“ mit 100 nF parallel geschaltet.
Dann erhält man am Lastwiderstand bereits das Informationssignal Uout1, bei dem allerdings noch ein großer Trägerrest überlagert ist.

Anschließend speist man mit diesem „leicht geglätteten“ Signal Uout1 einen **Tiefpass** (R = 50 kΩ / C = 4,7 nF), der bei Uout2 die Reste der Trägerspannung unterdrückt.
Simuliert wird von 0....10 ms mit 65536 Punkten. (Hinweis: Die Schottky-Diode findet man als BAT41 unter „**Bibliothek / Schottky Diodes**“).



17.2.3.2. AM-Demodulation mit einem Produktdetektor

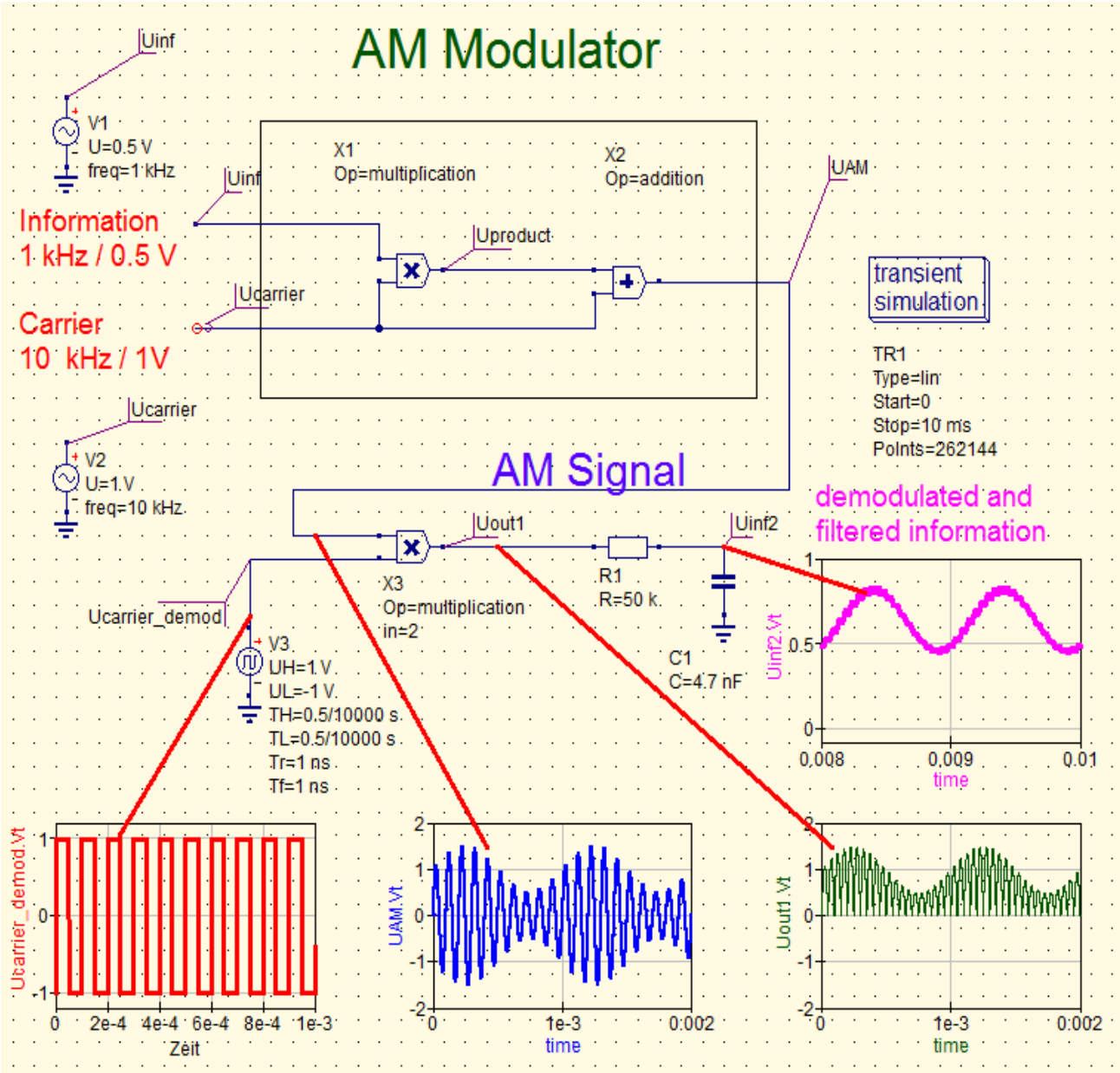
Das ist der moderne Stand der Technik. Er erfordert einen beträchtlich höheren Aufwand gegenüber dem Diodendetektor, aber er arbeitet bereits bei AM-Signalen mit Amplituden von einigen Mikrovolt praktisch verzerrungsfrei!

Der Grundgedanke ist sehr einfach: man multipliziert das AM-Signal mit einem Rechtecksignal, das dieselbe Trägerfrequenz und -Phasenlage, aber die Amplituden „+1“ und „-1“ aufweist.

Wird die Forderung

Frequenz und Phase des „Demodulator – Carriers“ völlig identisch mit dem AM-Signal

strikt eingehalten, dann werden die negativen Halbwellen des AM-Signals durch die Multiplikation mit dem Wert „-1“ des Carriers umgepolt und die Anordnung arbeitet wie ein **Zweipuls-Gleichrichter**…..selbst bei kleinsten Amplituden. Anschließend siebt ein LPF die Information als arithmetischen Mittelwert heraus.



17.3. SSB = Single Side Band Betrieb

Beim Spektrum eines „Zweiseitenband-AM-Signals“ gilt:

Nicht der Carrier, sondern nur die beiden „Seitenfrequenzen“ bzw. „Seitenbänder“ enthalten Informationen -- und dazu sind beide Informationen identisch!

Das führte schon früh dazu, dass man in der professionellen Funktechnik folgende Überlegungen anstellte:

Man kann den Träger reduzieren oder sogar komplett unterdrücken, denn er selbst enthält ja keine Information und frisst deshalb nur unnötig Sendeleistung. Diese Betriebsart heißt offiziell „Zweiseitenband-AM mit reduziertem oder unterdrücktem Träger“.

Wird der **Träger komplett unterdrückt**, dann haben wir immer noch **zwei Informations-Seitenbänder** (= LSB und USB) vor uns, die sich symmetrisch nach links bzw. rechts von der „Trägerfrequenz aus als virtuellem Nullpunkt“ erstrecken. Beide sind identisch -- bis auf die „Frequenzspiegelung“ des unteren Seitenbandes. Deshalb ist es naheliegend, **eines der beiden Seitenbänder auch noch zu unterdrücken** und die gesamte verfügbare Sendeleistung in das verbleibende andere Seitenband zu stecken. Das ist dann der SSB-Betrieb und er ist die Standard-Betriebsart z. B. auf Kurzwelle, wenn noch nicht digital gearbeitet wird.

Vorteile:
Nur noch halbe Bandbreite beim Sendesignal erforderlich.

Alle Sendeleistung im verbleibenden Seitenband bedeutend größere Reichweite bzw. besseren Signal-Rauschabstand beim Empfang

Bei Sprech- oder Sendepausen wird keine Sendeleistung benötigt oder abgestrahlt. Das steigert den Wirkungsgrad und spart Energie.

Nachteil:
Der technische Aufwand ist sowohl beim Sender wie auch beim Empfänger beträchtlich höher. Eine Demodulation mit dem einfachen Diodengleichrichter ist nicht mehr möglich. Das geht nur noch mit dem im vorigen Kapitel besprochenen **Produktdetektor.**

Aber:
Man muss genau gesagt bekommen, ob es sich um ein unteres Seitenband (= Lower Side Band = LSB) oder um ein oberes Seitenband (= Upper Side Band = USB) handelt, denn beim LSB haben wir bekanntlich eine „Kehrlage“ und der ehemalige Punkt „Null Hertz“ beim Informationsspektrum befindet sich nun an seinem „oberen Ende“. Und den müssen wir genau kennen...

Die Erzeugung von SSB-Signalen ist eine eigene Wissenschaft und arbeitet z. B. mit **Quarzfiltern** zur Unterdrückung des Trägers und nicht benötigten Seitenbandes.

Bei einer anderen Methode funktioniert das über die **Kompensation (= Auslöschung) des nicht benötigten Seitenbandes** durch Addition eines gegenphasigen Signals („Phasenmethode“).

Modernster Stand der Technik ist die Digitale Signalverarbeitung mit **IQ-Signalen, die für diese Zwecke einen „Half Complex Mixer“ einsetzt.**
Da wollen wir doch mal kurz reinschnuppern und sehen, was hinter dieser Technik steckt.

17.4. Reale, imaginäre und komplexe Signale

Alle elektrischen Signale, die wir sehen (Beispiel: Oszilloskop) oder hören können (Siehe: Mikrofon und Lautsprecher), spielen sich in der Wirklichkeit ab und stellen deshalb „**Reale Signale“** dar.

Signale, die wir nur anhand ihrer Auswirkung erahnen oder nur indirekt auf Umwegen aufspüren können, werden als „**Imaginäre Signale**“ bezeichnet.

Haben wir es dagegen mit einer Mischung aus diesen beiden Größen zu tun, dann spricht man von „**Komplexen Signalen“**.

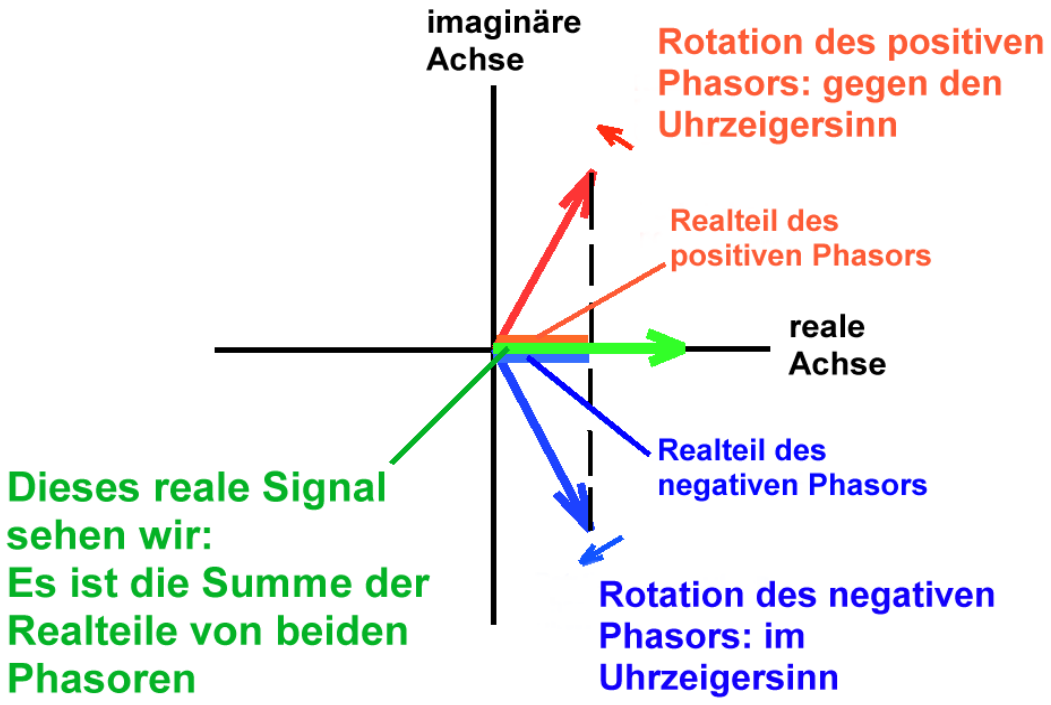
Diese Verhältnisse lassen sich in bekannter Weise für Sinus- und Cosinussignale mit **still stehenden Zeigern (englisch: Phasors)** in einem Achsenkreuz darstellen. Ein solcher Zeiger oder Phasor weist die beiden Eigenschaften **„Zeigerlänge = Signal-Amplitude**“ und „**momentane Phasenlage in einem Bezugssystem**“ auf. Er sieht also aus wie ein Uhrzeiger und steht fest, deshalb sieht man keinen zeitlichen Signal-Verlauf (…den würde man erst erhalten, wenn dazu das Achsenkreuz im Uhrzeigersinn mit der Signalfrequenz rotierte und man dabei den X-Anteil betrachten würde. Oder wenn das Achsenkreuz still steht – dann müssten sich die Zeiger darin drehen....)

Die **waagrechte Achse** des Diagramms ist dabei **real (= Wirklichkeit)**, die **senkrechte Achse** dagegen **imaginär (= nur vorstellbar)**. **Der Verlauf eines Oszillogramms in einer Schaltung ist dann nichts anderes als der Realteil des zugehörigen „komplexen Signals“.** Dadurch kommen wir zur Definition:

a) Rotiert ein solcher Zeiger = Phasor **gegen den Uhrzeigersinn**, dann weist das komplexe Signal nur „**positive Frequenzen“** auf. Folglich rotieren Phasors mit „negativer Frequenz“ im Uhrzeigersinn und ergeben für das komplexe Signal ein Spektrum, bei dem nur im negativen Frequenzbereich Spektrallinien zu finden sind.

b) Ein komplexes Signal ist nichts anderes als einer unser Phasoren mit seiner Summe aus Realteil und Imaginärteil, wenn man die Rotation mit seiner Eigenfrequenz dazu nimmt.

Sehen wir uns das am Beispiel einer so simplen und realen Sache wie dem 1 kHz – Ton aus einem Signalgenerator an. Er besteht aus zwei gegensinnig rotierenden Zeigern = Phasors mit gleicher Rotationsfrequenz von f = 1 kHz -- aber mit Zeigerlängen, die jeweils nur die halbe Amplitude des gemessenen 1 kHz- Signals besitzen. Ist ja auch logisch, denn wenn beide Phasors gerade in Richtung der waagrechten, realen Achse zeigen, dann muss man als Summe exakt den Spitzenwert des Generatorsignals erhalten.



Auch die mathematische Beschreibung ist nicht sehr schwierig, denn Dank Herrn Euler lässt sich das damit sehr anschaulich zeigen:

Der „positive Phasor“ wird durch den Ausdruck

$$U_{positiv} = U_{positiv_max} * e^{j\omega t} = U_{positiv_max} * [\cos(\omega t) + j \sin(\omega t)]$$

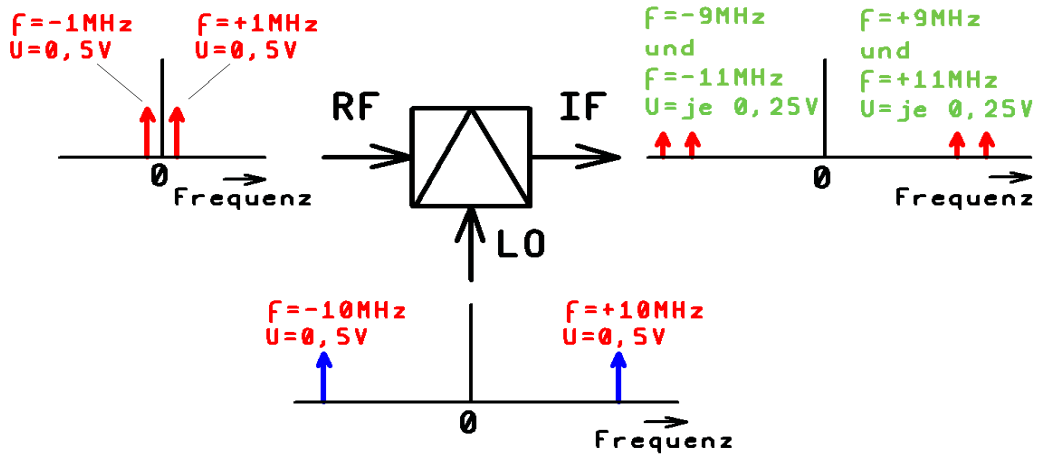
beschrieben.
Und für den negativen Phasor gilt entsprechend:

$$U_{negativ} = U_{negativ_max} * e^{-j\omega t} = U_{negativ_max} * [\cos(\omega t) - j \sin(\omega t)]$$

Addiert man nun die beiden Phasors in unserem Beispiel, dann heben sich die **gegenphasigen imaginären Anteile heraus**, während die beiden realen Teile als Summe die reale Cosinusschwingung mit der Maximal-Amplitude

$$U_{gesamt_max} = U_{positiv_max} + U_{negativ_max} = 2 * U_{positiv_max}$$

erzeugen, zu der die Frequenz f = 1 kHz gehört. Das zugehörige Frequenzspektrum des realen Signals muss nun auch den negativen Bereich umfassen und dort findet sich der negative Phasor mit der Frequenz f = -1 kHz.
Wem das etwas merkwürdig oder unglaublich vorkommt, der erinnere sich mal daran, was in einem analogen RF-Mixer passiert, wenn in ihm zwei verschiedene **reale Cosinussignale** (z. B.: RF = 1 MHz, LO = 10 MHz, Amplituden jeweils = 1V) miteinander multipliziert werden. Da finden wir nämlich am Ausgang (sowohl im negativen wie auch positiven Frequenzbereich) symmetrisch zur LO-Frequenz nur das ehemalige RF-Signal, aufgeteilt in einen LSB- und einen USB-Anteil mit jeweils nochmals halbierter Amplitude. Also ist durch die Multiplikation eine **Frequenzverschiebung des Pärchens aus „positivem und negativem RF-Phasor“ zur LO-Frequenz als neuen Nullpunkt erfolgt** und dadurch bekommen wir auch plötzlich den geheimnisvollen negativen RF-Phasor als LSB-Signal sogar auf dem Schirm eines Spektrum Analysers zu sehen.
Dieses Bild zeigt anschaulich diese Verhältnisse, sowohl für den positiven wie auch für den negativen Frequenzbereich.



Und für Leute mit Freude an der komplexen Mathematik sieht das so aus, wenn man **zwei reale Cosinussignale** (mit den Frequenzen f_1 und f_2) miteinander multipliziert:

$$(\cos\omega_1 t) \cdot (\cos\omega_2 t) = \left[\frac{\left(e^{j\omega_1 t} + e^{-j\omega_1 t}\right)}{2}\right] \cdot \left[\frac{e^{j\omega_2 t} + e^{-j\omega_2 t}}{2}\right] =$$

$$= \frac{\left[e^{j(\omega_1+\omega_2)t} + e^{-j(\omega_1+\omega_2)t}\right] + \left[e^{j(\omega_1-\omega_2)t} + e^{-j(\omega_1-\omega_2)t}\right]}{4} = \frac{1}{2} \cdot [\cos(\omega_1 + \omega_2)t + \cos(\omega_1 - \omega_2)t]$$

17.5. Analytic Pairs

Klingt sehr geheimnisvoll, ist es aber nicht. Im Normalfall haben wir es ja nicht nur mit einer einzelnen Informationsfrequenz wie einem Sinuston zu tun, sondern wir hören z. B. bei Musik eine Vielzahl von Einzelfrequenzen und deren Oberwellen, die dazu pausenlos ihre Werte ändern. Will man bei einem solchen Signal die Bandbreite halbieren, dann pickt man sich je nach Wunsch entweder nur den positiven oder negativen Frequenzbereich heraus -- man **geht also von einem realen Signal zu einem komplexen Signal über**. Das geht so:

Das reale Ausgangssignal wird als „I"-Signal (= In Phase Signal) betrachtet und unverändert weiterverarbeitet.

Dieses reale „I"-signal speist zusätzlich eine besondere Schaltung mit dem Namen „Hilbert Transformator". An ihrem Ausgang finden wir zwar dieselben Amplituden bei allen Spektralanteilen, aber bei jeder Spektrallinie und bei jeder Frequenz erfolgte eine **Phasenverschiebung um 90 Grad! Dieses künstlich erzeugte Signal trägt den Namen „Q" = „Quadrature Signal"**.

Arbeitet man mit Digitaler Signalverarbeitung, dann stehen uns jetzt zwei digitale Datenströme „I" und „Q" am Ausgang zur Verfügung, die nun die beiden Anteile des komplexen Signals darstellen.

Das ist das berühmte „Analytic Pair" und wir finden jetzt -- je nach Vorzeichen beim Imaginärteil -- entweder nur noch positive oder nur noch negative Frequenzen und damit nur die halbe belegte Bandbreite vor! (Oft wählt man eine um +90 Grad voreilende Phase und es gibt dadurch nur positive Frequenzen). Das kennen wir als „**SSB-Signal**".

17.6. Beispiel: Half Complex Mixer zur Erzeugung eines SSB-Signals

Dazu benötigen wir folgende Zutaten:

a) Ein reelles Informationssignal **„U_{inf}" = „I_{inf}"**, (z. B. Sprache oder Musik), das mit Unterstützung eines **Hilbert Transformers** in ein Analytic Pair umgewandelt wird. Es nimmt dann folgende Form an:

$$U_{inf} = I_{inf} + j * Q_{inf}$$

b) Einen komplexen Träger $U_{carrier} = U_{carrier_max} * e^{+j\omega_c t}$ der als Sinus- **und** Cosinussignal zur Verfügung stehen muss und nur eine positive Frequenz aufweist.

c) Zwei **Multiplizierer.**

d) Einen **Subtrahierer**.

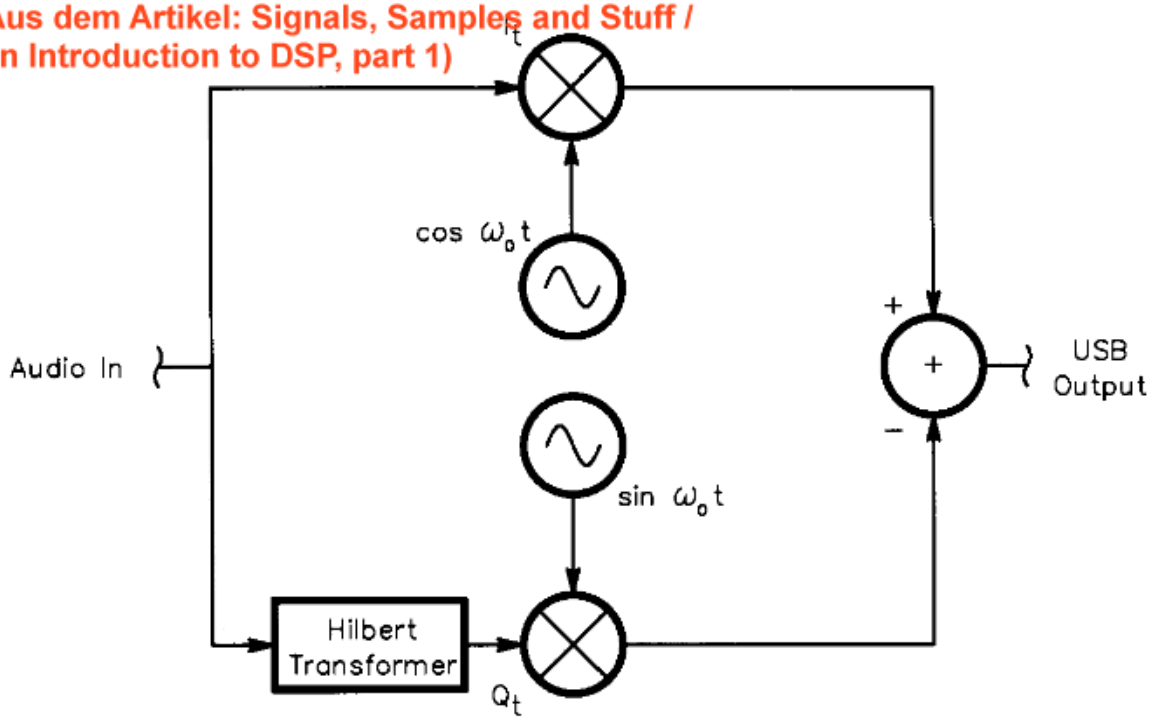
Nun multiplizieren wir das Analytic Pair der Information mit dem komplexen Träger, der nur eine positive Frequenz aufweist:

$$U_{inf} * U_{carrier} = (I_{inf} + j * Q_{inf}) * [\cos(\omega_c t) + j * \sin(\omega ct)] =$$

$$[I_{inf} * \cos(\omega_c t) - Q_{inf} * \sin(\omega ct)] + j * [Q_{inf} * \cos(\omega_c t) + I_{inf} * \sin(\omega ct)]$$

Rot markiert ist der Realteil. **Nur den können wir per Radiowellen in die Luft schicken** und er muss nun durch folgende praktische Schaltung realisiert werden".

(Aus dem Artikel: Signals, Samples and Stuff / an Introduction to DSP, part 1)



Anmerkungen:
Der Hilbert-Transformer mit seiner frequenzunabhängigen Phasenverschiebung von 90 Grad ist in digitaler Ausführung kein Problem. Versucht man das jedoch mit einer Analog-Schaltung, dann funktioniert das nur in einem begrenzten Frequenzbereich korrekt. Leider....

Diese Schaltung liefert das **„Upper Side Band = USB“.**

Braucht man dagegen das „Lower Side Band = LSB“, dann muss das Vorzeichen im unteren Zweig umgekehrt und damit der Subtrahierer am Ausgang durch einen Addierer ersetzt werden.

17.7. Demodulation von amplitudenmodulierten IQ-Signalen

Das ist ganz schnell erklärt:
Die beiden Anteile des **Analytic Pairs „I“ und „Q“ sind stets um 90 Grad gegeneinander phasenverschoben** und deshalb gilt hier der „Satz des Pythagoras“:

(Zeigerlänge)^2 = I^2 + Q^2

Man erhält folglich die Zeigerlänge, also den Amplitudenverlauf, durch das Ziehen der Quadratwurzel aus der Summe der beiden quadrierten Komponenten „I“ und „Q“. Und so etwas beherrscht ein Digitaler Signalprozessor ohne große Mühe.

17.8. Demodulation von SSB-Signalen

Wenn wir die Sache „**analog“** lösen, dann benötigten wir dazu ein **Trägersignal mit exakt derselben Frequenz, die der bei der SSB-Erzeugung unterdrückte Träger aufwies**. Damit wird der **Produktdetektor** aus Kapitel 2.2.2. gespeist und das SSB-Signal mit diesem „Ersatzträger“ multipliziert. Dann bekommen wir wieder exakt unsere gewünschte Information am Ausgang (als Differenzfrequenz).

Mit **DSP und IQ-Signalen und Hilbert Transformer** sieht das etwas aufwendiger aus und dazu gibt es manche Informationen im Internet. Dieser Artikel ist besonders hübsch:

A DSP SSB DEMODULATOR

Figure 13 shows an example of a DSP SSB phasing method demodulator. Once the complex-valued BFO of $e^{-j(\omega_c n t_s)} = \cos(\omega_c n t_s) - j\sin(\omega_c n t_s)$ down-converts the RF SSB to zero Hz, it's sensible to decimate the multipliers' outputs to a lower fs sample rate to reduce the processing workload of the Hilbert transformer. We could have performed decimation by a factor greater than 10, but doing so would make the design of the post-D/A analog lowpass filter more complicated. The digital LPFs, whose positive-frequency cutoff frequency is slightly greater than 4 kHz, attenuate any unwanted out-of-baseband spectral energy in the down-converted signal and eliminate any spectral aliasing caused by decimation.

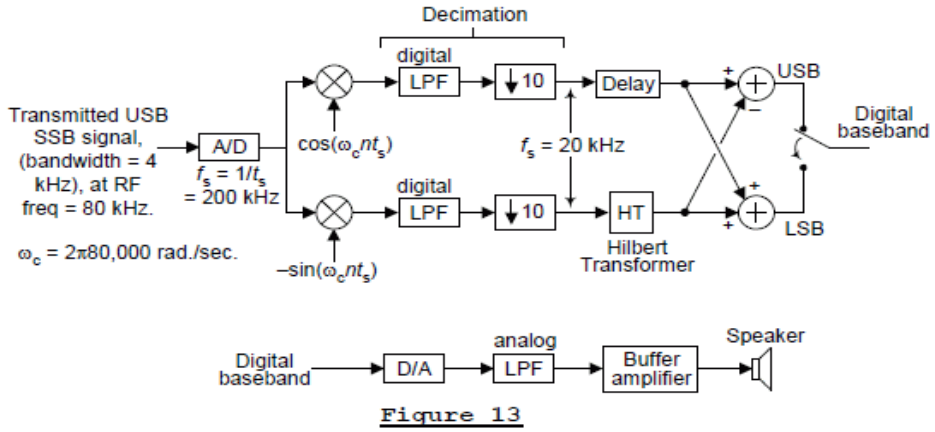


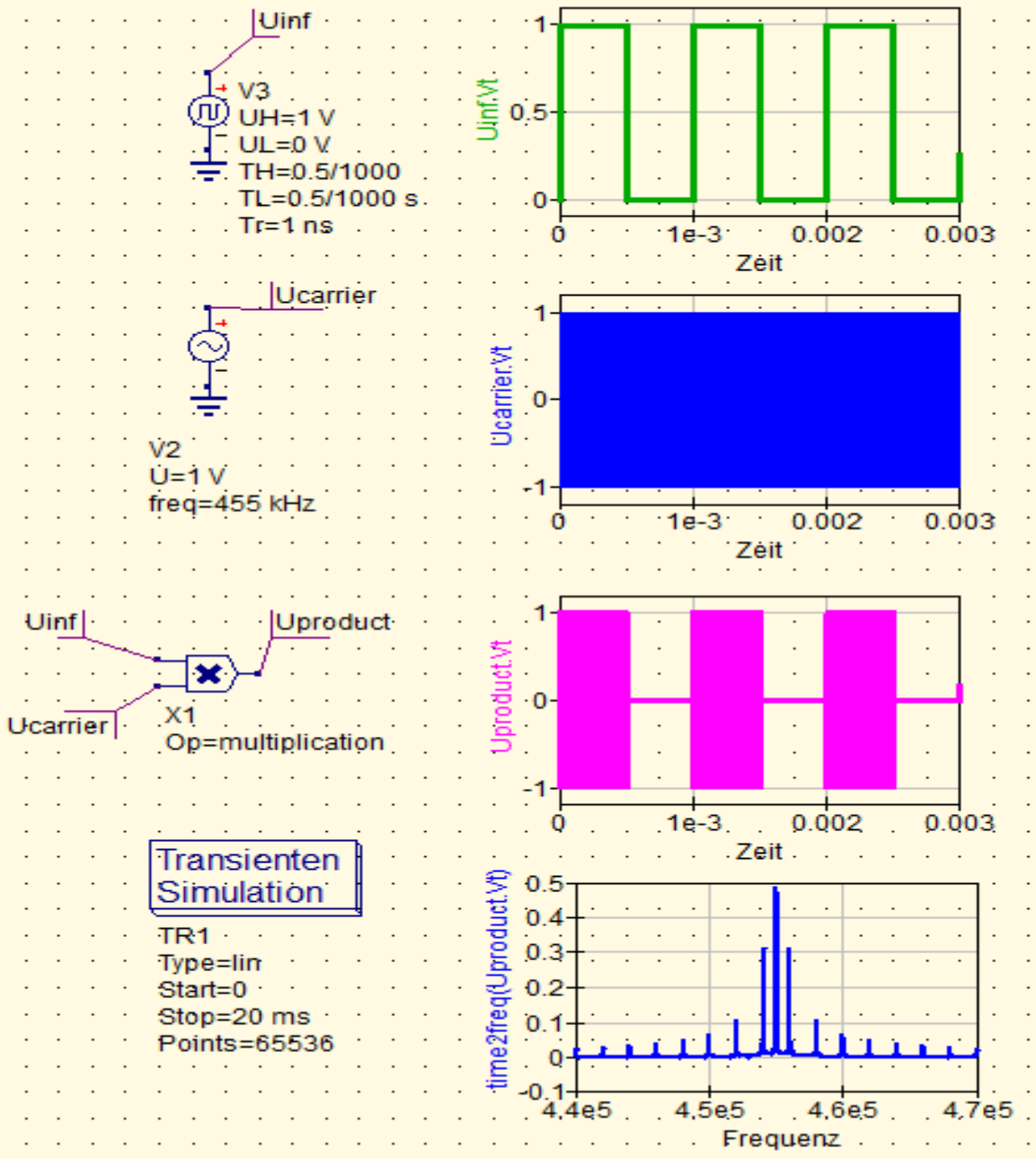
Figure 13

The Delay element in the upper path in Figure 13 is needed to maintain data synchronization with the time-delayed Hilbert transformer output sequence in the bottom path. For example, if a 21-tap digital Hilbert transformer is used, then the upper path's Delay element would be a 10-stage delay line [2].

Quelle:
Understanding the 'Phasing Method' of Single Sideband Demodulation
aus: **http://www.dsprelated.com/showarticle/176.php**

17.9. Amplitude Shift Keying (ASK)

Damit ist nun wirklich die die gute alte Morsetaste mit ihrem „EIN“ und „AUS“ gemeint. Diese so simple Methode hat auch heute noch ihren Platz in der modernen Übertragungstechnik, denn da werden den beiden Zuständen einfach die logischen Pegel Null und Eins zugeordnet und so eine Digital-Übertragung durchgeführt. Das funktioniert erstaunlich gut und leidlich störsicher (…wenn man mit zusätzlicher Begrenzung und einem Schmitt-Trigger als Schwellwert-Komparator arbeitet). Außerdem hält sich der Hardware-Aufwand in Grenzen.
Bei der Simulation **multiplizieren wir nun den Träger mit einem Pulssignal.** Dessen kleinster Spannungswert sei Null Volt, das ergibt „Taste offen“. Die Maximalspannung sei dagegen 1V und dadurch ist „die Taste gedrückt“. Die Wiederholfrequenz sei 1 kHz.
Beim Trägersignal nehmen wir 455 kHz (= die AM-Zwischenfrequenz in einem Radio) und bleiben bei einem Spitzenwert von 1 V. Das ergibt diese Simulations-Schaltung.



Am Ende folgt das Spektrum des ASK-Signals und was man da sieht, war zu erwarten. Die symmetrische rechteckförmige Information enthält nämlich außer der Grundwelle noch alle **ungeradzahligen Oberwellen von 1kHz** und das Ergebnis sind folglich **ein LSB bzw. USB mit all diesen Spektralanteilen der Information, die sich nach links und nach rechts vom Träger ausbreiten.**

17.10. Die Phasenmodulation

Eine ganz raffinierte Sache, denn man fummelt diesmal nicht an der Amplitude des Trägers herum. Ja, sie ist sogar absolut unwichtig und wird z. B. in einem entsprechenden Empfänger durch Begrenzung von allen Schwankungen und Störspitzen befreit. Dafür ändert man die **Phase der Trägerfrequenz** **im Rhythmus der Informationspannung**. Das kostet zwar deutlich mehr Bandbreite und wesentlich höheren technischen Aufwand, aber der Erfolg ist überzeugend -- man muss nur mal auf UKW gute Musik anhören und die Qualität mit einem Mittel- oder Kurzwellensender vergleichen. Dort wird zwar Frequenzmodulation (FM) eingesetzt, aber die ist mit der PM sehr eng verwandt und wir werden später noch darauf eingehen.

17.10.1. Erzeugung einer Phasenmodulation

Da uns im qucsstudio ein Phasenmodulator als fertiger Baustein zur Verfügung steht, sehen wir uns den gleich an. Und wer in der mitgelieferten Technischen Dokumentation von qucsstudio nachschaut, findet dies:

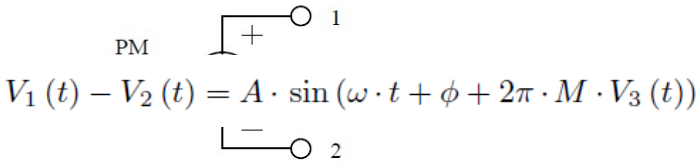
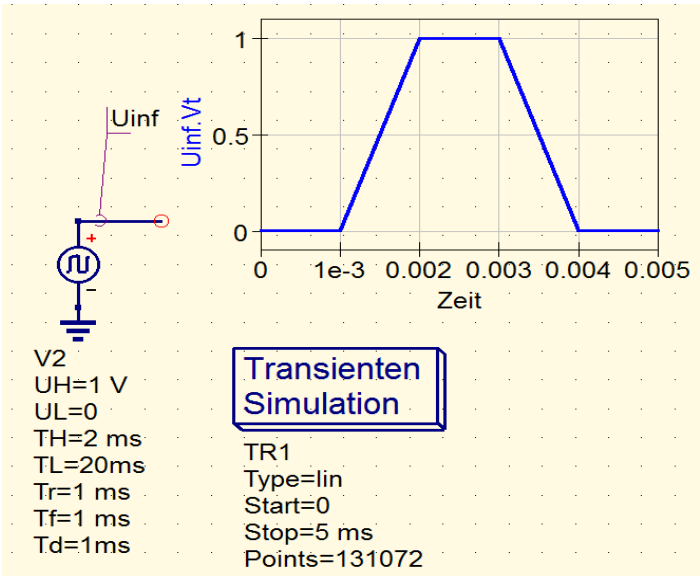


Figure 9.14: PM modulated AC source

Das Ausgangssignal ist also eine **Sinusschwingung [A*sin(ώ*t)]**, die außerdem eine **Phasenverschiebung „Φ“** beim Start aufweist.
Die **Phasenmodulation** erfolgt über den dritten Anteil in der Klammer, der durch die Spannung V3(t) gesteuert wird. Den wollen wir uns näher ansehen:
2π*M*V3(t) können wir nämlich schreiben als **[2π*M] * V3(t)**

und da sehen wir, dass wir in der ersten Klammer einen interessanten Ausdruck haben. „M“ heißt nämlich „**Modulationsindex**“ und wenn er mit „2π“ multipliziert wird, dann stellt das Ergebnis die „Phasenverschiebung im Bogenmaß“ – also in Radians! – dar, die wir unserer Sinusschwingung verpassen! (...Bitte dran denken, dass „2π“ nichts anders bedeutet als „einmal im Kreis rum“, also einen Winkel von 360 Grad...).

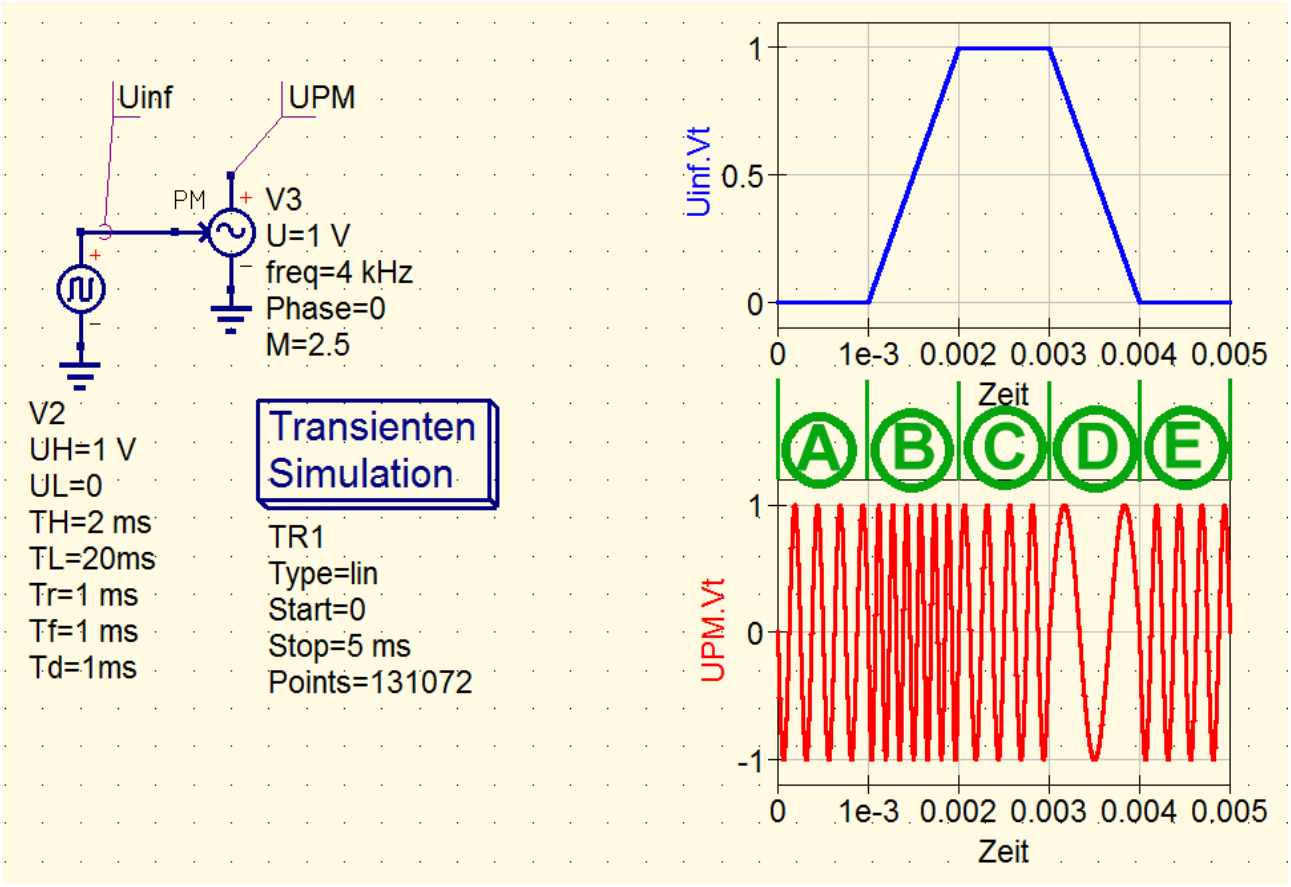
Durch den weiteren Faktor V3(t) machen wir anschließend diese Phasenverscheibung im Rhythmus von V3(t) zeitabhängig!



Das blickt man natürlich erst so recht nach einigen Beispielen. Fangen wir also mit etwas Einfachem an:

Wir nehmen ein Puls-Signal, das erst nach einer Millisekunde startet und eine Anstiegszeit von einer Millisekunde aufweist. Die H-Amplitude lassen wir für eine weitere Millisekunde auf 1 Volt, dann folgt eine Millisekunde Abfallzeit. Simuliert wird für 5 Millisekunden. So sieht das Signal aus wie ein „Puls mit müder Anstiegsflanke.
Die „LOW-Zeit“ betrage 20 Millisekunden, es werden 131072 Punkte simuliert.
Bitte diese Schaltung erstellen und austesten!

In diese Schaltung bauen wir nun zusätzlich die **PM-modulierte Quelle**
ein und analysieren anschließend die **5 Zeitabschnitte von A bis E.**



Bereich „A“:
Die Informationsspannung hat den Wert „Null“. Also erhalten wir nur eine **Dauerschwingung mit der konstanten Träger-Mittenfrequenz** nach der Formel V3 = sin(ωt)

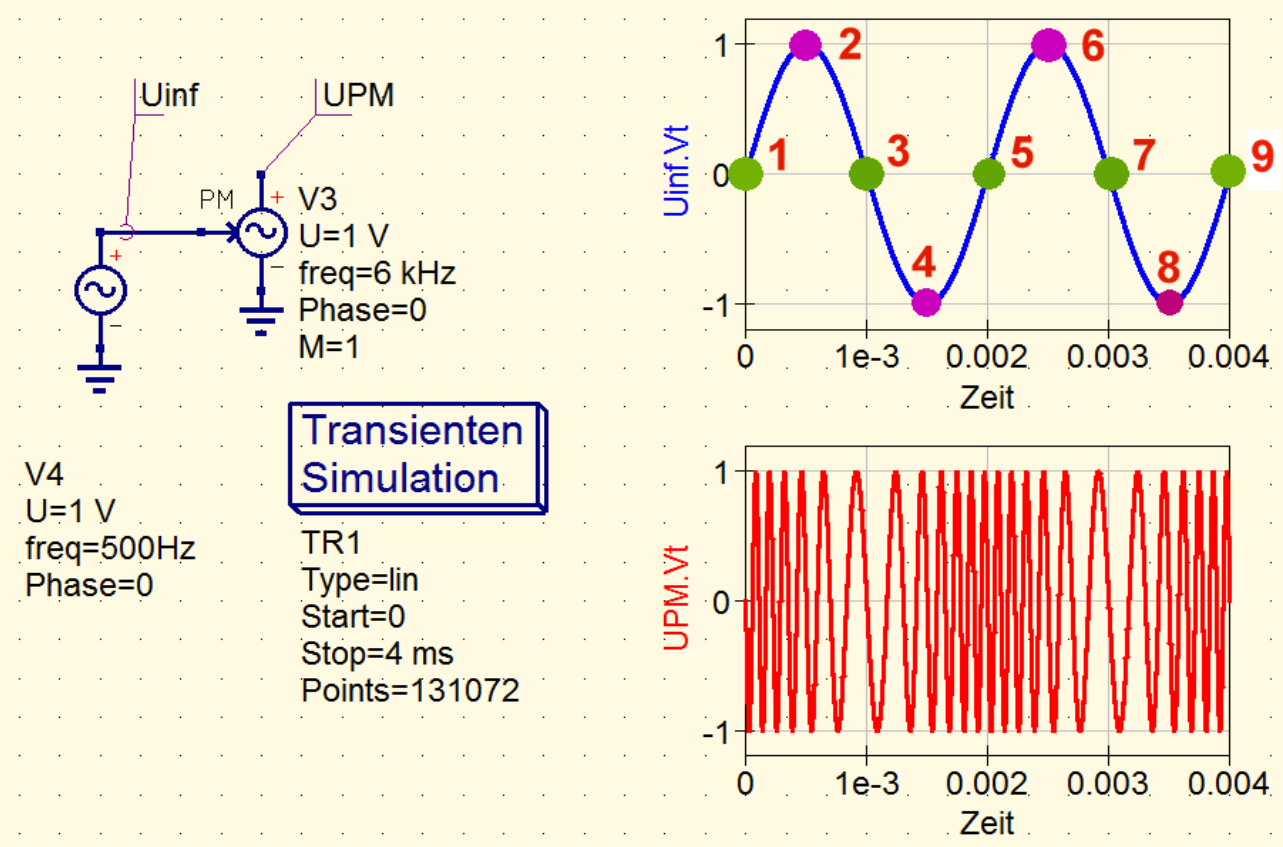
Bereich „B“:
Die Informationsspannung **steigt linear mit der Zeit an und erzwingt damit eine linear mit der Zeit zunehmende Phasen-Voreilung**. Das ist nur möglich durch eine höhere, aber konstante Schwingfrequenz, denn dadurch „dreht sich der Propeller schneller und schafft einen gleichmäßig ansteigenden Vorsprung““

Bereich „C“:
Die Informationsspannung ist nun wieder konstant und damit steigt die Phasenverschiebung nicht mehr weiter an. Das Ergebnis ist erneut eine **Dauerschwingung mit der konstanten Träger-Mittenfrequenz nach der Formel V3 = sin(ωt)**, aber mit der zuletzt erreichten Wert der Phasen-Voreilung.

Bereich „D“:
Die Informationsspannung ist nun negativ und das erzwingt eine gleichmäßige Verminderung der Phasenverschiebung. Dazu gehört aber eine **tiefere, aber wieder konstante Schwingfrequenz in diesem Abschnitt.**

Bereich „E“:
Da diese Abfallzeit genau so lang ist wie die Anstiegszeit im Bereich „B“, drehen wir die Phase wieder exakt bis zum Startwert bei „A“ zurück. Also beobachten wir erneut die **Dauerschwingung mit der konstanten Träger-Mittenfrequenz nach der Formel V3 = sin(ωt)**,

17.10.2. Ergebnis bei sinusförmiger Informationsspannung



Punkte „1“ bis „2“
Die ansteigende Informationsspannung erzwingt ein Voreilen der Phase (...und damit eine dazu erforderliche Frequenzerhöhung). Aber die wird umso geringer, je näher wir dem Punkt „2“ kommen. Genau beim Punkt „2“ wird diese Voreilung zu Null, die höhere Augenblicks-Frequenz ist jetzt unnötig und wir sind wieder bei der „Ausgangs-Mittenfrequenz“ angelangt.

Punkte „2“, „3“ und „4“:
Die Phasen-Voreilung nimmt nun wieder ab. Wir durchlaufen den Ausgangspunkt „3“ und es geht weiter abwärts. Also wird aus der Voreilung ein Nacheilen und in dieser Zeit muss dazu die „Augenblicks-Frequenz“ niedriger als die Ausgangs-Mittenfrequenz“ sein. Erst beim Punkt „4“ kommt die Sache wieder zum Stillstand und die Schwingung weist wieder die Ausgangs-Mittenfrequenz auf.

Punkte „4“, „5“ und „6“:
Die Phasenverschiebung nimmt wieder Fahrt auf, geht wieder in die positive Richtung und deshalb brauchen wir dazu erneut eine höhere Augenblicks-Frequenz. Erst beim Punkt „6“ beobachten wir wieder die Ausgangs-Mittenfrequenz – natürlich kombiniert mit entsprechender Voreilung.

Punkte „6“ bis „8“
Jetzt geht es im gleichen Stil wieder rückwärts mit der Phase und damit zeigt sich wieder eine tiefere Augenblicks-Frequenz. Erst beim Punkt „8“ gibt es wieder einen Stillstand der Phasenverschiebung und die Ausgangs-Mittenfrequenz hat wieder ihre Chance. Usw.....

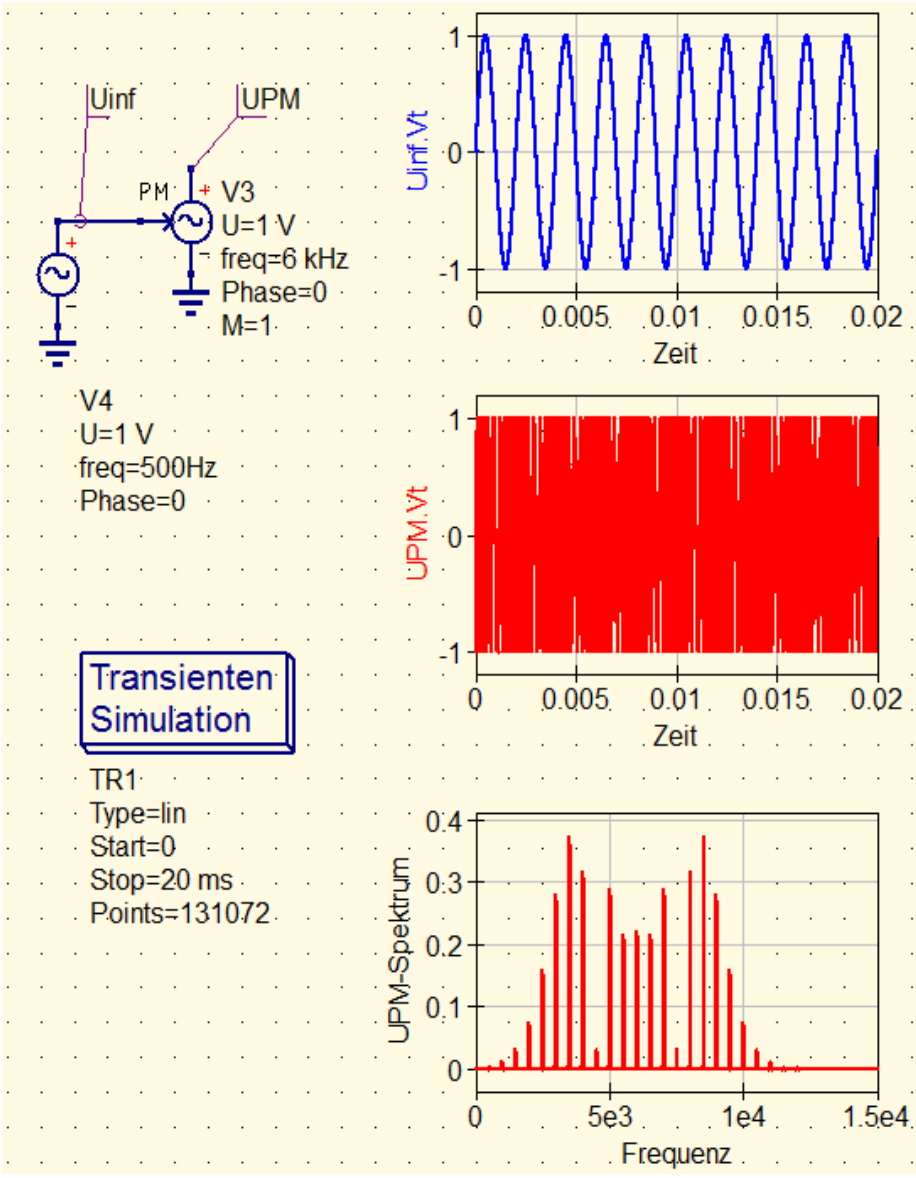
Wer sich die Diagramme genau anschaut, erkennt noch eine weitere wichtige Sache:

Zwischen der Informationsspannung zur Steuerung der Phasenverschiebung und dem zugehörigen Verlauf der Augenblicks-Frequenz besteht eine Phasenverschiebung von 90 Grad!
Das muss so sein, denn irgendwo im Fachbuch steht der Satz:

Die Kreisfrequenz ist die erste Ableitung der Phase nach der Zeit (ω(t)= dφ / dt)....

….und damit haben wir bereits die enge Verwandtschaft von Phasen- und Frequenzmodulation!

17.10.3. Das Frequenzspektrum bei Frequenz- und Phasenmodulation



Nehmen wir uns nochmals das eben simulierte Beispiel vor und lassen uns zusätzlich das Frequenzspektrum anzeigen. Allerdings sollten wir bei einer Informationsfrequenz von f_{inf} = 500 Hz mit einer höheren Frequenz-Auflösung von 50 Hz arbeiten und deshalb die Simulationszeit auf t = 1 / 50Hz = 20 ms erhöhen. Dabei sollten 131072 Punkte simuliert werden.
Die erforderlichen Eigenschaften des neuen Diagramms lauten

time2freq(UPM.Vt)

und wir wollen den Frequenzbereich von 0....15000 Hz darstellen.

Wenn man das Spektrum genau ansieht, dann sieht man:

Es besteht aus **vielen Linien, die alle den Abstand der Informations-frequenz von 500 Hz aufweisen.**

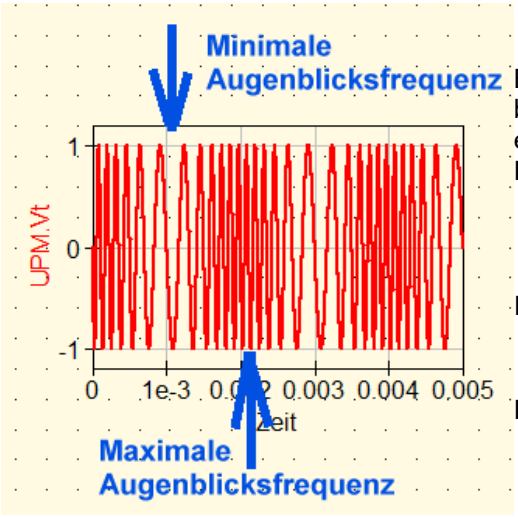
Sie gruppieren sich symmetrisch um den **Wert von 6 kHz (= unmodulierte Mittenfrequenz)** und ihre Amplituden folgen einer **Bessel-Funktion.**

Je größer der Modulationsindex „m“ (= und damit der **Phasenhub „m*2π“ = maximale Phasenvor- oder Nacheilung gegen die „Ruhelage“ der Phase**) wird, **desto mehr Linien kommen dazu und die belegte Bandbreite steigt entsprechend. Und zusätzlich ändert sich die Amplitudenverteilung nach den zum neuen Modulationsindex gehörenden Werten der Bessel-Funktion.**

17.10.4. Einige wichtige Informationen zur Phasen- und Frequenzmodulation

Es gilt:
Phasen- und Frequenzmodulation lassen sich weder vom Signalverlauf, noch vom Spektrum her unterscheiden, wenn die Information nur aus einem einzigen sinusförmigen Signal besteht!

Ist das nicht der Fall, dann muss man einige weitere Zusammenhänge kennen:



Bei der Modulation mit nur einer sinusförmigen Information beobachtet man im zeitlichen Signalverlauf eine maximale und eine minimale „Augenblicks-Frequenz“. Diese (symmetrische) Frequenzabweichung von der Mitte heißt

„Frequenzhub ΔF“

Ihr Wert beträgt z. B. bei einem UKW-Radiosender:

ΔF = +-75 kHz

FM-Funkgeräte arbeiten meist mit einem Hub von etwa +-2 kHz.

Lässt man jedoch bei einer FM-Radiostation, die gerade Musik mit **Tönen zwischen 20 Hz und 15 kHz** ausstrahlt, den Frequenzhub konstant, so beobachtet man einen unangenehmen Effekt:

Mit steigender Informationsfrequenz ist für den gewünschten Frequenzhub immer weniger „Phasenhub“ (= Abweichung von der „Ruhephase“) erforderlich ….ist irgendwie logisch, den mit steigender Frequenz werden bestimmte Phasenwerte immer schneller erreicht und durchlaufen! Es gilt hier:

Phasenhub = (Frequenzhub) / (Informationsfrequenz)

Weniger Phasenhub bedeutet aber eine zunehmende Empfindlichkeit gegen Rausch- und Störspannungen – damit wird mit steigender Informationsfrequenz der Signal / Rauschabstand schlechter!

Man hilft sich da mit einer „**Pre-Emphase**“ im FM-Sender: **die höheren Informationsfrequenzen modulieren den Sender stärker als die tiefen und erzeugen so einen größeren Phasenhub** (= Hochpass-Verhalten für die modulierende Informationsspannung).
Im FM-Empfänger muss man natürlich diesen veränderten Informations-Frequenzgang durch einen passenden Tiefpass korrigieren. Diese Schaltung heißt „**De-Emphase**“.

Jetzt noch die Antwort auf die Frage: „Welche Bandbreite belegt etwa ein solches FM-Spektrum?“
Da gibt es eine hübsche Faustformel und die lautet:

FM-Kanalbreite = 2 * (Frequenzhub + maximale Informationsfrequenz)

Beispiel für einen UKW-FM-Stereo-Radiosender mit einem Frequenzhub von 75 kHz und einer höchsten auftretenden Informationsfrequenz von 57 kHz (= RDS-Signal):

b = 2 * (75 kHz + 57kHz) = 264 kHz.

Und nochmals zum Abspeichern im Gedächtnis:
Bei der **Frequenzmodulation** hält man den **Frequenzhub konstant**. Nach der obigen Formel **MUSS dann der Modulationsindex und damit der maximale Phasenhub mit steigender Informationsfrequenz ABNEHMEN. Das verschlechtert leider den Signal-Rauschabstand bei hohen Tönen…**

Bei der **Phasenmodulation** hält man dagegen den **Modulationsindex = Phasenhub** **konstant**. Nach obiger Formel wird sich dann der **Frequenzhub mit steigender Informationsfrequenz vergrößern** und das erhöht wiederum die belegte Bandbreite – aber durch den konstanten Phasenhub erhalten wir im gesamten Informations-Frequenzbereich denselben Signal-Rauschabstand und dieselbe Übertragungsqualität!
==========

17.11. FSK = Frequency Shift Keying = vom Phasenmodulator zum Frequenzmodulator

Hier handelt es sich um eine digitale Betriebsart (manchmal als Frequenzumtastung bezeichnet), die den beiden Zuständen „log 0“ und „log 1“ zwei unterschiedliche Frequenzen zuordnet. Ihre Wurzeln liegen schon weit zurück, nämlich in der damaligen Fernschreibtechnik. Das einzige, was davon übrig geblieben ist, sind die beiden Bezeichnungen „**mark**“ für die höhere Frequenz und „**space**“ für die tiefere Frequenz.

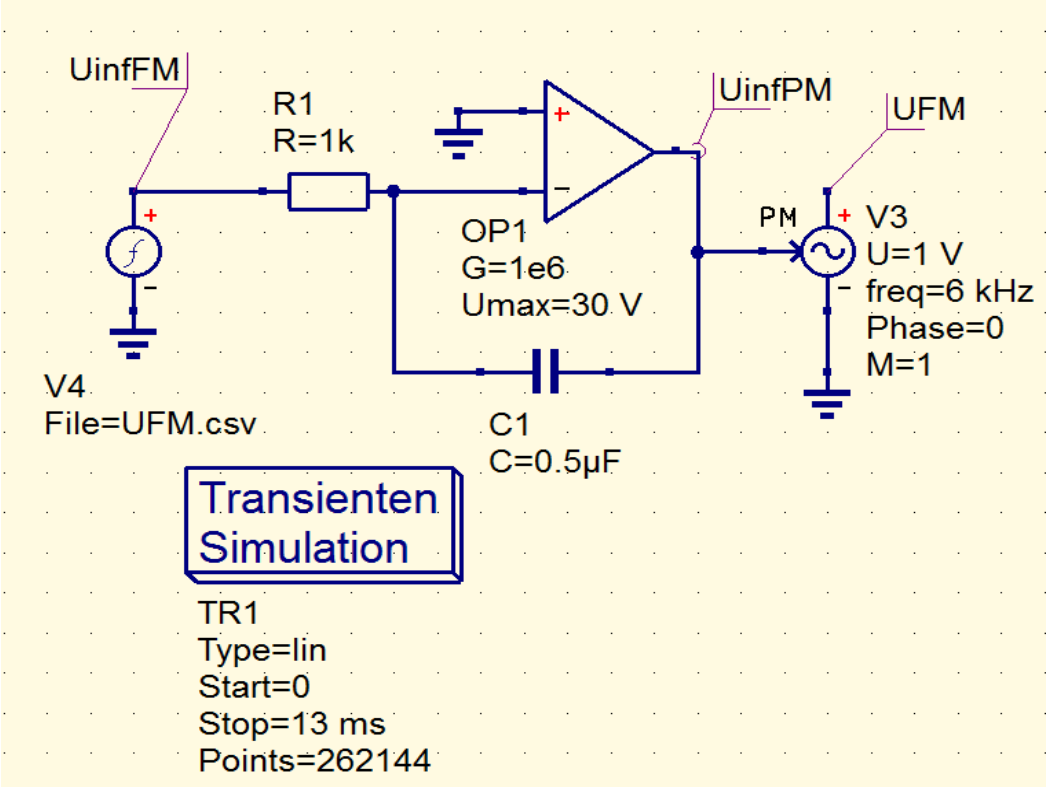
Wir besitzen in qucsstudio nur einen Phasenmodulator und müssen daraus einen Frequenzmodulator basteln. Das ist nicht allzu schwer, denn es galt der Satz:

Die Kreisfrequenz ist die erste Ableitung der Phase nach der Zeit
(ω(t)= dφ / dt)....

Wenn wir nun über die Phase die Frequenz steuern wollen, dann müssen wir diesen Satz umkehren:

Die Phase erhalten wir, wenn wir die Frequenz über der Zeit integrieren.

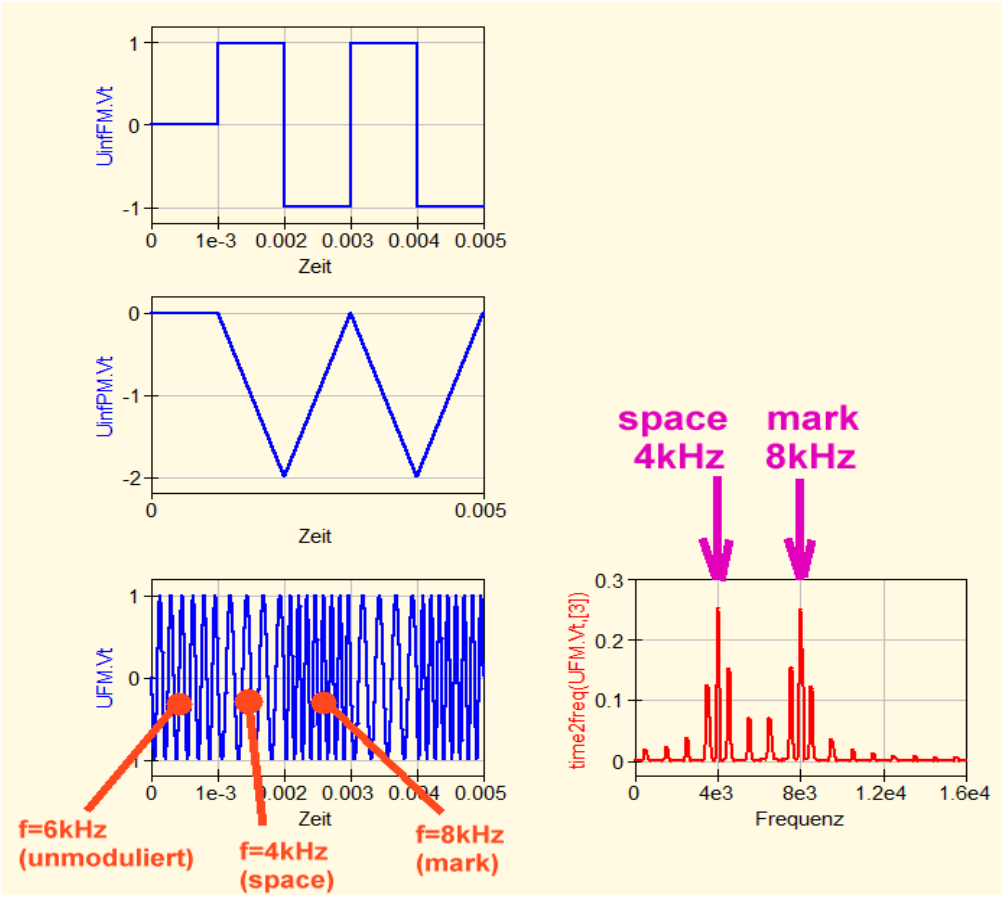
Also schalten wir vor den Steuereingang des Phasenmodulators einen Integrator und realisieren ihn über eine einfache OPV-Schaltung:



```
0           0
1e-3        0
1.001e-3    1
2e-3        1
2.002e-3    -1
3e-3        -1
3.002e-3    1
4e-3        1
4.002e-3    -1
5e-3        -1
5.002e-3    1
6e-3        1
6.002e-3    -1
7e-3        -1
7.002e-3    1
8e-3        1
8.002e-3    -1
9e-3        -1
9.002e-3    1
10e-3       1
10.002e-3   -1
11e-3       -1
11.002e-3   1
12e-3       1
12.002e-3   -1
13e-3       -1
13.002e-3   1
14e-3       1
14.001e-3   0
20e-3       0
```

Den Verlauf der Informationsspannung $U_{\mathbf{infFM}}$ erzeugen wir wieder über ein „PWL-Signal“, also mit einer „Dateibasierten Spannungsquelle“ und schreiben die erforderlichen Wertepärchen in diese Datei „UFM.csv“.
Sie wird im Projektordner gespeichert.

Die ersten beiden Bilder zeigen die Spannung vor und hinter dem Integrator.



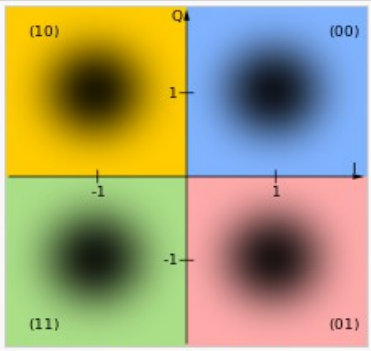
Hinter dem Phasenmodulator steht nun das geforderte FSK-Signal zur Verfügung. Und auch das Spektrum passt dazu:

Es weist je ein Maximum an Linien um die „mark“-Frequenz (= 8 kHz) und die „space“-Frequenz (= 8 kHz) auf.

Aufgabe:
Der **Integrator hat ja die Phase umgekehrt** und so erhalten wir für die Spannung mit dem Wert „+1V“ die tiefe Frequenz (= space).
Bitte korrigieren Sie das, indem Sie **hinter den Integrator noch eine Umkehrstufe schalten!**

17.12. Einige Beispiele zur QAM

17.12.1. Erzeugung einer 4-QAM (= 4-PSK)



Konstellationsdiagramm einer 4-QAM. Die zulässigen Entscheiderbereiche sind mit verschiedenen Farben hinterlegt. Die Empfangssymbole sind grafisch als eine Dichtefunktion mit unterschiedlich starken Grauwerten dargestellt.

Das hatten wir ja schobeim Einstieg in das Modulationsthema und wir wollen versuchen, so einen QAM-Modulator in der Simulation zum Laufen zu bringen.

Dazu brauchen wir aber nochmals den Phasenmodulator aus dem „technical handbook for qucs“ (...gibt es in der qucsstudio-Homepage) und analysieren die Formel zur Beschreibung des Verhaltens:

$$\phi + 2\pi \cdot M \cdot V_3(t))$$

„**A“ ist die Zeigerlänge,** vom Mittelpunkt aus gemessen. Sie bleibt in diesen vier Fällen konstant.

Wenn wir uns den Klammerinhalt ansehen, dann ist der Rest recht einfach:

Der erste Ausdruck „**ώt“ legt die Frequenz fest,** mit der wir arbeiten. Nehmen wir wieder **f = 6 kHz.**

Den Start-Phasenwinkel **ϕ setzen wir zu Null**.

Beim dritten Ausdruck verwenden wir **M =1** und steuern die Phasenlage der Spannung $V_3(t)$ über folgende Werte:

Für „45 Grad“: **$V_3(t) = 0{,}125$**

Für „135 Grad“: **$V_3(t) = 0{,}375$**

Für „225 Grad“: **$V_3(t) = 0{,}625$**

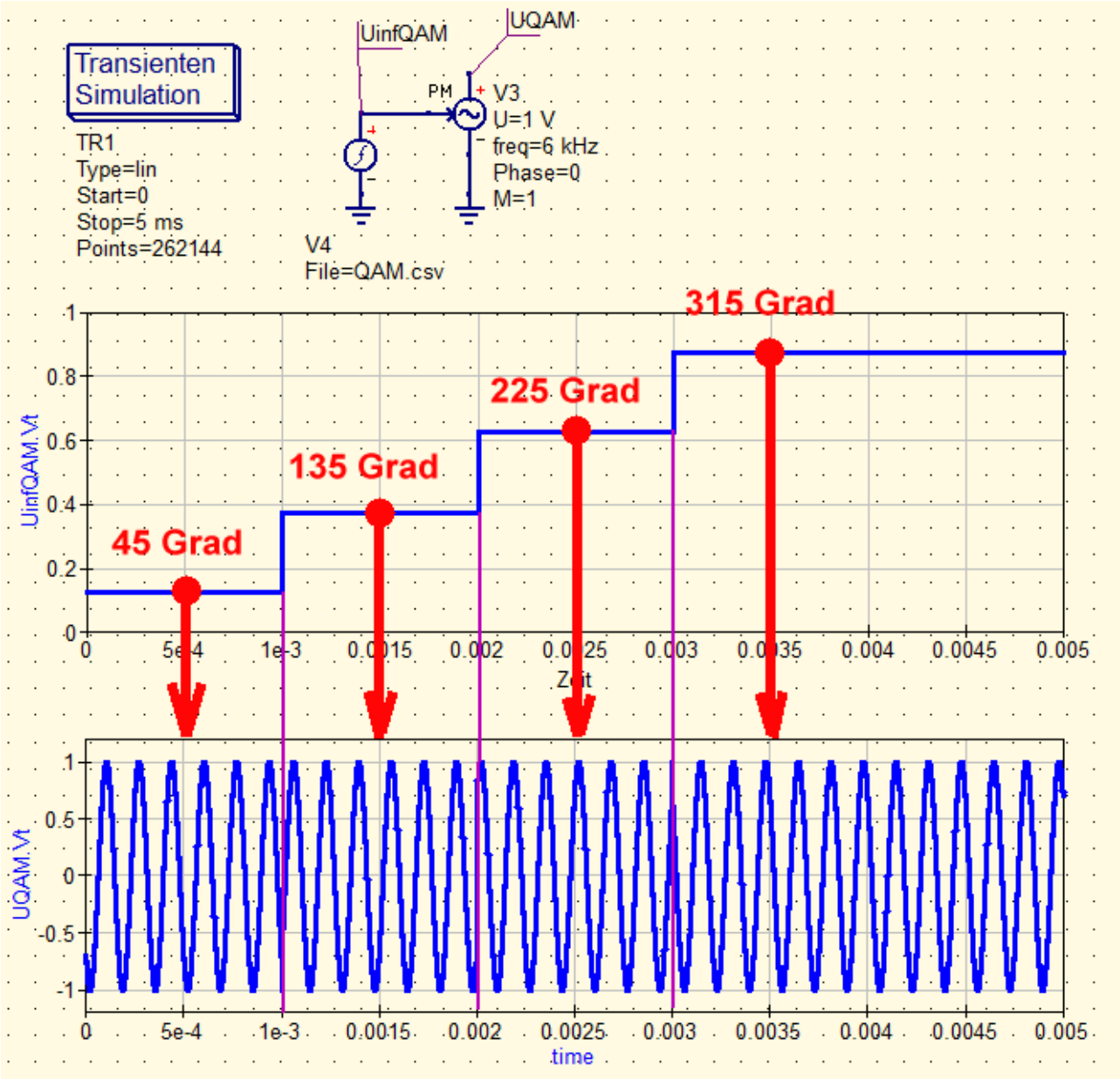
Für „315 Grad“: **$V_3(t) = 0{,}875$**

(Diese Werte werden ja in der Formel mit „2π*M“ multipliziert und „2π*1“ ergibt den vollen Kreisumfang mit 360 Grad...)

```
0               0.125
1e-3            0.125
1.001e-3        0.375
2e-3            0.375
2.001e-3        0.625
3e-3            0.625
3.001e-3        0.875
4e-3            0.875
```

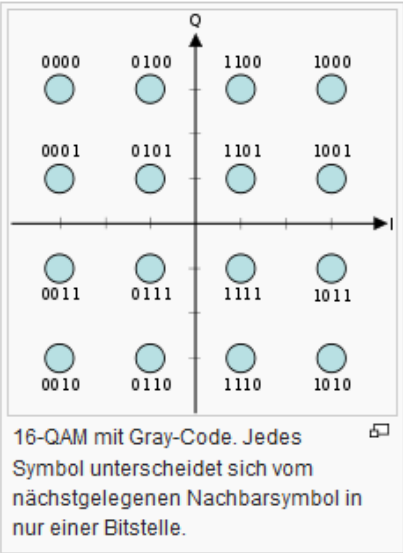
Damit schreiben wir folgende **„QAM.csv“** - Datei zur Ansteuerung des Phasenmodulators:

Das erzeugt den folgenden Verlauf von U_{QAM} mit den zugehörigen Phasenzuständen:



(Ein Tipp:
Um die Diagramme so weit in die Breite zu ziehen, markiert man sie erst und zerrt dann an einer der vier Ecken. Dort ist ein kleines Quadrat als „Öse für den Zughaken“ sichtbar)

17.12.2. Die 16-QAM



16-QAM mit Gray-Code. Jedes Symbol unterscheidet sich vom nächstgelegenen Nachbarsymbol in nur einer Bitstelle.

Auch auf sie haben wir schon bei der Einleitung hingewiesen. Allerdings steigt nun der technische Aufwand deutlich, wenn wir die Formel zur Beschreibung unseres Phasenmodulators ansehen:

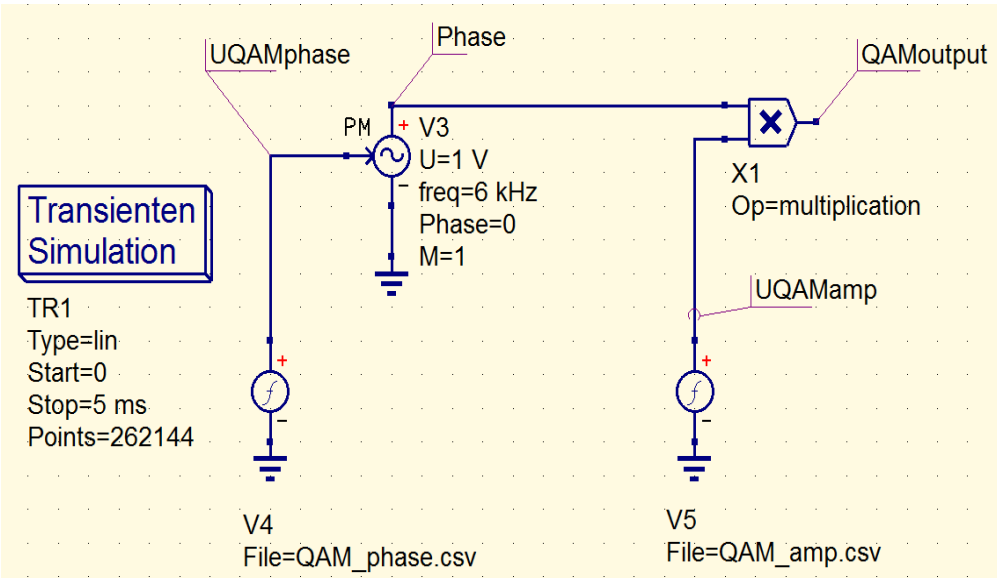
$$V_1(t) - V_2(t) = A \cdot \sin(\omega \cdot t + \phi + 2\pi \cdot M \cdot V_3(t))$$

Wir müssen nämlich bei jedem Punkt, den wir übertragen wollen, die Zeigerlänge über die Amplitude „A“ vor der Klammer

UND ZUSÄTZLICH

die zugehörige Phase über die Amplitude von V3(t) in der Klammer einstellen.

So würde eine Simulation aussehen:



Jetzt muss man natürlich zwei „*.csv“-Dateien schreiben und im Projekt abspeichern. Eine steuert die Phasen-Einstellung, die andere die Amplituden-Einstellung.
Dann könnte man diese Idee an einigen Punkten des Diagramms testen. Aber dazu benötigt man erst mal den genauen Amplituden- und Phasenwert jedes einzelnen Punktes aus dem obigen Diagramm....

Übungsaufgabe:
Übertragen Sie nacheinander die drei Bitkombinationen **„0000“, „0011“, „1110“ und 1010“** mit 16-QAM entsprechend dem obigen Diagramm. Die Trägerfrequenz sei wieder f = 6 kHz, die Trägeramplitude U = 1 V = 100%. Verwenden Sie dasselbe Zeitschema wie im vorigen Beispiel (4-QAM). Für die vier Bit-Kombinationen gelten folgende Zeiger:

0000 = 100% Zeigerlänge und ein Winkel von 135 Grad.

0011 = 74% Zeigerlänge und ein Winkel von 200 Grad.

11100 = 74% Zeigerlänge und ein Winkel von 290 Grad.

1010 = 100% Zeigerlänge und ein Winkel von 315 Grad.

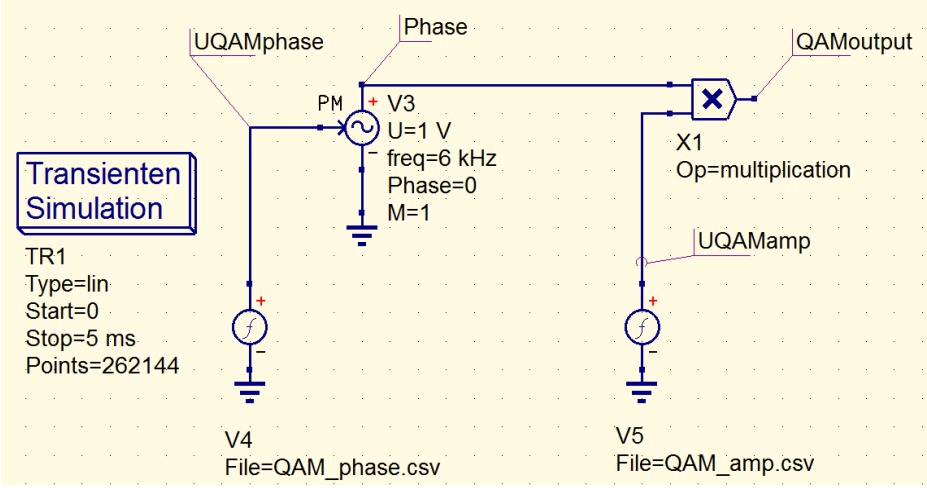
Lösung:

Das ist die Datei „**QAM_amp.csv**“...

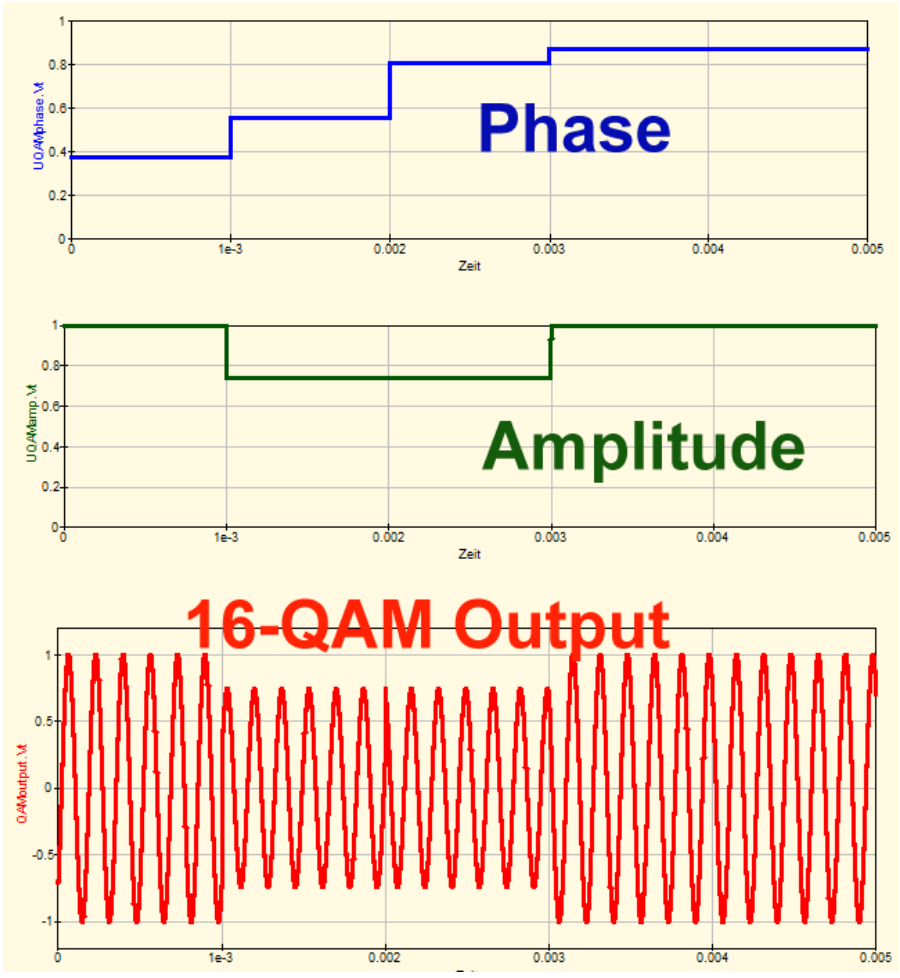
0	1
1e-3	1
1.001e-3	0.74
2e-3	0.74
2.001e-3	0.74
3e-3	0.74
3.001e-3	1
4e-3	1

und das die Datei „**QAM_phase.csv**“

0	0.375
1e-3	0.375
1.001e-3	0.5555
2e-3	0.5555
2.001e-3	0.8055
3e-3	0.8055
3.001e-3	0.875
4e-3	0.875



Das ist nochmals die erforderliche Simulationsschaltung....



...und das ist das korrekte Ergebnis.

17.12.3. Demodulation von QAM-Signalen

Man ahnt nun schon den Weg:

Das QAM-Signal speist parallel zwei getrennte Produktdetektoren.

Jeder Produktdetektor benötigt zusätzlich ein Signal mit der Trägerfrequenz, aber diese beiden Schaltspannungen müssen sich exakt um 90 Grad in der Phase unterscheiden und mit den Phasenlagen der im Sender verwendeten IQ-Trägersignale überein stimmen.

So bildet man die beiden Achsen „I“ und „Q“ nach und braucht dann nur noch zu multiplizieren. Auch eine passende Filterung der beiden demodulierten IQ-Signale muss folgen.

18. Rauschen

18.1. Erste Bekanntschaft mit Rauschsignalen

Empfängt man eine Nachricht ohne jegliche Information dann ist darin alles zufällig und ungeordnet und ergibt keinen Sinn oder ermöglicht keine Voraussagen. Jeder kennt das als „Rauschen“ und das klingt wirklich wie das Rauschen eines Wasserfalls oder einer Wasserleitung. Jede Antenne empfängt ein solches Signal aus der Atmosphäre oder aus dem Weltraum, jeder elektrische Widerstand oder Leiter erzeugt von sich aus eine solche Spannung durch die „Wärmebewegung der Leitungselektronen“, sofern wir nicht bis zum absoluten Nullpunkt abkühlen. Man drehe einfach mal sein UKW-Radio oder sein TV-Gerät auf einen Leerkanal zwischen zwei Stationen, dann weiß man, was gemeint ist. Aber Vorsicht: bei verschlüsselten Sendungen klingt das oft genau so wie Rauschen...und doch steckt Information drin. Das ist allerdings Absicht, um eventuelle Mithörer zu täuschen! Auch „künstlich erzeugtes Rauschen“ (= pseudo random noise) ist oft nie ganz perfekt.

Für „ideales Rauschen“ = „Weißes Rauschen = Johnson Noise“ gilt aber: Seine Leistungsdichte für jedes Hertz an Bandbreite ist über der Frequenz konstant und das ist typisch für die in Ohm’schen Widerständen entstehenden Rauschspannungen.

(Sobald man hier diesen Leistungsdichte-Verlauf z. B. durch Filter verändert, spricht man vorn „Rosa Rauschen“). Doch nun zur Frage:

18.2. „Rauschen“ -- woher kommt das?

Das lässt sich schnell und präzise beantworten: in jedem elektrischen Widerstand werden Elektronen bewegt, wenn Strom fließt. Sobald aber Wärme mit im Spiel ist (…..und das ist automatisch oberhalb des absoluten Nullpunktes stets der Fall…), werden diese Teilchen immer unruhiger und nehmen nicht den geraden Weg von Minus nach Plus. Sie stoßen zusammen, prallen zurück, werden nach vorne oder zur Seite geschleudert…. der Strom schwankt also durch den Wärme-Einfluss dauernd und völlig unregelmäßig um kleinere und größere, aber winzige Beträge. Selbst wenn ***gar keine äußere Spannung angelegt ist***, merken wir dieses Gerangel unter den Ladungsträgern und messen an den Anschlüssen des Bauteils eine kleine „Rausch-Leerlaufspannung U_{NOISE}“. Sie lässt sich folgendermaßen berechnen:

$$U_{NOISE} = \sqrt{\frac{4 \cdot h \cdot f \cdot B \cdot R}{e^{\frac{hf}{kT}} - 1}}$$

mit
h = Planck’schem Wirkungsquantum
k = Boltzmann – Konstante = 1,38 x 10^{-23} J / Kelvin
T = absoluter Temperatur in Kelvin
B = betrachteter Bandbreite in Hz
f = Mittenfrequenz der betrachteten Bandbreite in Hz
R = Widerstandswert in Ω

Das sieht nun entsetzlich kompliziert und praxisfremd aus, aber man kann bis mindestens 100 GHz und Temperaturen bis hinunter zu 100 K problemlos die einfache (und bekannte) Näherungsformel verwenden:

$$U_{NOISE} = \sqrt{4 \bullet k \bullet T \bullet B \bullet R}$$

Stellt man diese Formel etwas um, dann sieht man plötzlich sofort, was los ist:

$$\frac{\left(\frac{U_{NOISE}}{2}\right)^2}{R} = k \bullet T \bullet B$$

Das ist schlicht und einfach eine Leistungsangabe! Also wird in jedem Widerstand -- unabhängig von seinem Widerstandswert! -- durch den **Einfluss der Wärme die gleiche Rausch-LEISTUNG entstehen.** Sie ergibt die in der vorherigen Formel angegebene Leerlaufspannung U_{NOISE}.

Sobald man also diesen Widerstand als Spannungsquelle mit Urspannung U_{NOISE} und (rauschfreiem) Innenwiderstand R auffasst, muss man an diese Quelle einen (ebenfalls rauschfreien) Lastwiderstand mit dem gleichen Wert R anschließen, um Leistungsanpassung zu erhalten. **Am Lastwiderstand liegt dann die halbe Urspannung U_{NOISE} und an den Lastwiderstand wird die verfügbare Rauschleistung „kTB" abgegeben**.

Die Rauschleistung steigt linear mit der absoluten Temperatur des Bauteils und der zur Verfügung gestellten Bandbreite (…die Spannung folgt dann natürlich mit der Wurzel aus der Leistung). Sie ist normalerweise unabhängig von der Frequenz und trägt in einem solchen Fall den Namen „**Weißes Rauschen".**

Ganz wichtig ist nun folgende Sache:
Fast immer arbeitet man bei Empfängern und Systemen mit „Pegeln" anstelle von Spannungswerten. Das ist ein logarithmisches Maß und damit kann man das **Multiplizieren** (wenn z. B. Stufen in Reihe geschaltet sind und ihre Verstärkungen multipliziert werden müssen…) durch eine **Addition** ersetzen. Am bekanntesten dürfte hier die Einheit „**dBm**" sein. Dabei wird nicht mit Spannung, sondern mit Leistung gearbeitet und jeder vorhandene Leistungswert wird ins Verhältnis zum Bezugswert

P_0 = 1 Milliwatt am Systemwiderstand

gesetzt. Das ergibt den Leistungspegel:

$$10 \bullet \log\left(\frac{\text{Leistungswert}}{\text{1mW}}\right) \; in\ dBm$$

Betrachtet man nun die obige Sache mit der **Rauschleistung „kTB" etwas genauer,** dann lässt sich eine interessante Vereinfachung einführen:

Die Rauschleistungsdichte „kT" stellt die Leistung in jedem „Hertz an Bandbreite" dar und man muss sie mit der geltenden Bandbreite multiplizieren, um die gesamte produzierte Rauschleistung zu erhalten.

Geht man nun auf die Pegelrechnung über, dann sollte man folgendes wissen und gut abspeichern:
Jeder Widerstand produziert damit bei Raumtemperatur (T_0 = 300 K) einen **internen Rauschpegel** und damit eine interne Rauschleistungsdichte von etwa -168 dBm. Korrekt sind es

-167,8 dBm pro Hz Bandbreite

Dadurch kann er an einen **rauschfreien** Lastwiderstand bei **Leistungsanpassung einen um 6dB niedrigeren Leistungspegel** von etwa

-173,8 dBm pro Hertz

abgeben.
Bei größeren Bandbreiten als 1 Hz ist der Rest dann ganz einfach:

Tatsächlicher maximal abgebbarer Rauschpegel in dBm =
-173,8 dBm + 10log(Bandbreite in Hz)

Aufgabe 1:
Wie groß sind **IM LEERLAUF** der Rauschpegel und die Rauschspannung an einem Widerstand von 50 Ohm und 300 Kelvin, wenn mit einer Bandbreite von 1 Hz gearbeitet wird? **Bestätigen Sie diese Ergebnisse durch eine qucsstudio-Simulation.**

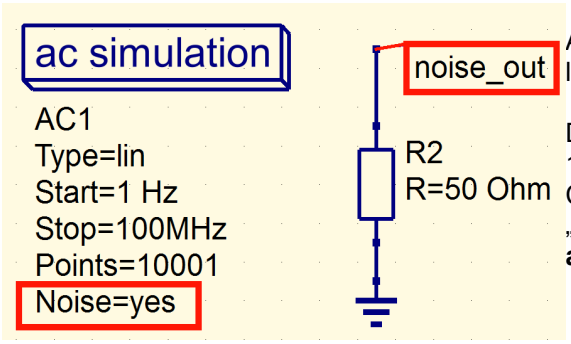
Lösung:
Der Rauschpegel = **-167,8 dBm** (Siehe oben) entspricht einer Leistung von

P = (1mW) * 10exp(Pegel /10) = 1,66 *10exp(-20) Watt

Daraus wird schließlich mit einem „Quellwiderstand von 50 Ohm“ eine Leerlaufspannung von

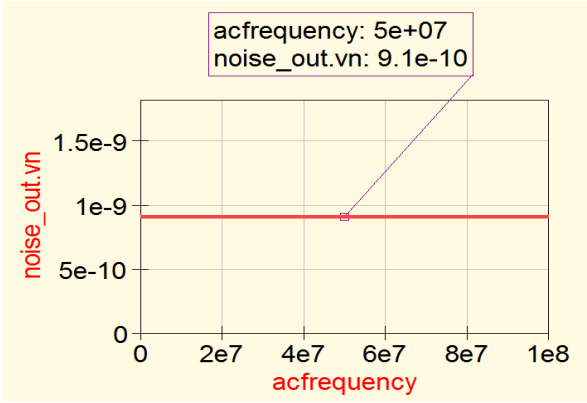
U = sqrt(P * 50Ohm) = 0,91 nV

Und so sieht die Simulation aus:



An einem 50Ω – Widerstand, der an einem Ende an Masse liegt, wird der Label „**noise_out**“ angebracht.

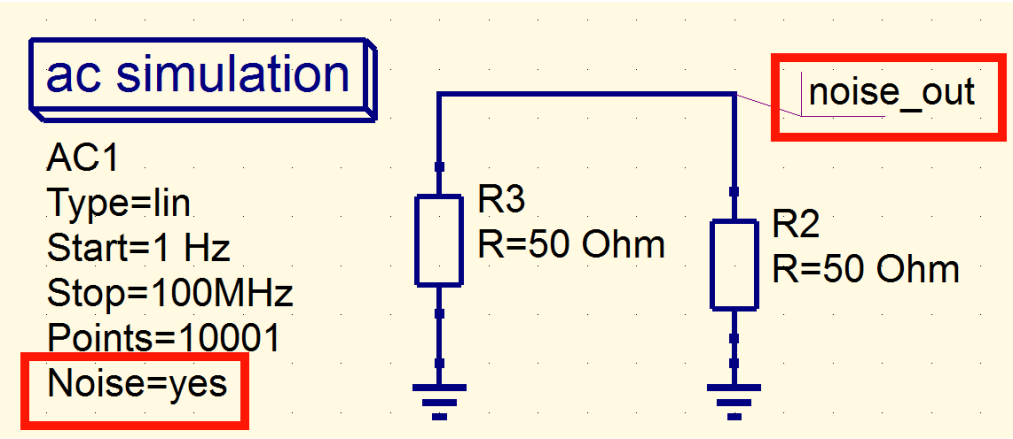
Dann wird eine AC – Simulation von 1 Hz bis 100 MHz mit 10001 Punkten vorbereitet. Allerdings muss man nach „Edit Component Properties“ darin noch auf die zweite Karteikarte „**Properties**“ wechseln, um die **Rauschsimulation zu aktivieren.**



Hier haben wir die erwarteten 0,91 NanoVolt, mit dem Marker bei 50 MHz abgelesen.

==========

Aufgabe 2:
Schließen Sie diesen Widerstand von 50Ω mit einem weiteren Widerstand mit gleichen Wert ab und bestimmen Sie dann den Rauschpegel **„noise_out“** an der Verbindungsstelle.



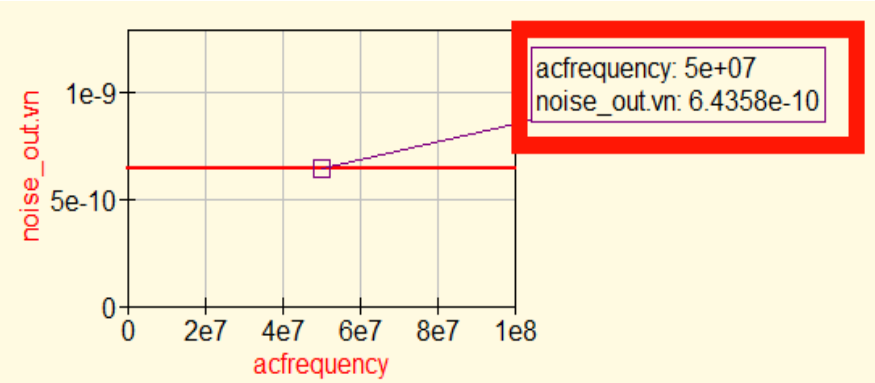
Analysieren und begründen Sie das Ergebnis.

So geht das Programm vor:

Es betrachtet zunächst den linken rauschenden Widerstand R3 als eine Quelle mit der Rausch-Urspannung von 910 pV und dem rauschfreien Innenwiderstand R3. Diese Quelle wird mit dem rauschfreien Widerstand R2 = 50 Ω belastet und das ergibt 0,455 nV = 455 pV pro Wurzel aus 1 Hz beim Punkt „**noise_out“**.
Nun wiederholt es diese Rechnung für den Widerstand R2 als Rauschquelle und das führt ebenfalls zu 455 pV pro Wurzel aus 1Hz bei „out“. Am Ende addiert man diese beiden Teilsignale nach dem „Überlagerungssatz“.

Allerdings hat man es dabei mit „nicht korrelierten Signalen“ zu tun und darf deshalb nur die Effektivwerte geometrisch addieren:

$$U_{gesamt} = \sqrt{[(0{,}455nV)^2 + (0{,}455)^2]} = 0{,}635nV$$



Genau das zeigt uns die Simulation…

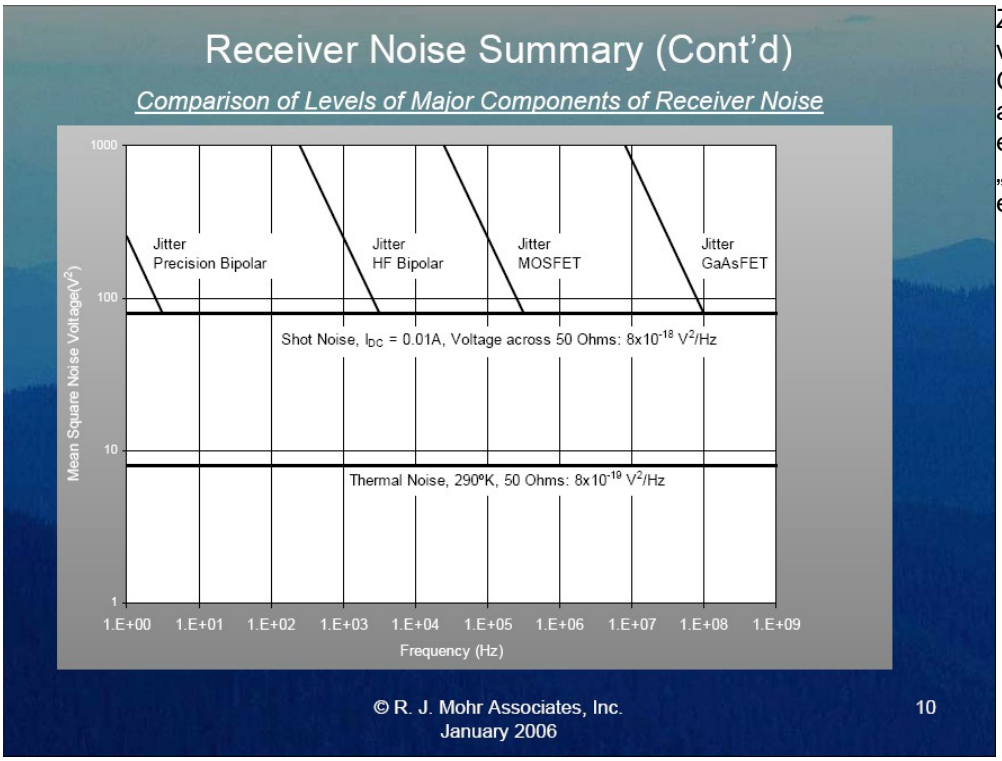
==========

18.3. Weitere Rauschquellen

In jedem aktiven Bauteil (Röhre, Bipolar-Transistor, Sperrschicht-FET, MOSFET, HEMT usw.) haben wir es außer mit dem thermischen Rauschen noch mit zwei zusätzlichen Rauscharten zu tun:

a) ***Shot-Noise*** (= Schrot-Rauschen) tritt bei Vakuumdioden und P-N-Übergängen durch Ungleichförmigkeiten des Stromflusses beim Durchqueren der Potential-Unterschiede auf. Es handelt sich hierbei um breitbandiges, weißes Rauschen.

b) ***Flicker-Noise = Jitter Noise*** oder „**1 / f – Noise**“ (= Funkel-Rauschen) entsteht durch Verunreinigungen oder Defekten im Kristallaufbau. Sie führen zu kurzen impulsförmigen Schwankungen beim Stromfluss -- und zu einem Impuls gehört eben ein Spektrum, dessen Leistungsdichte mit steigender Frequenz abnimmt. Man definiert hier wie bei einem Tiefpass eine „Corner Frequency“ (= Grenzfrequenz) und es ist interessant zu sehen, wie sich die aktiven Bauteile grundsätzlich voneinander unterscheiden. ***Das untenstehende Bild*** gibt hier einen guten Eindruck und die zugehörige Quelle (Mohr Associates) sollte man sich unbedingt aus dem Internet holen, denn sie stellt eine sehr präzise, aber kompakte und gleichzeitig gut verständliche Einführung dar.



Zusätzlich arbeitete man viele Jahre lang mit Gasentladungsröhren als Rauschquellen, die ein sehr breitbandiges „Plasma-Noise“ erzeugten.

==========

18.4. Weißes Rauschen zur Messung einer Übertragungs -- Funktion

18.4.1.. Zur Erinnerung: die S-Parameter

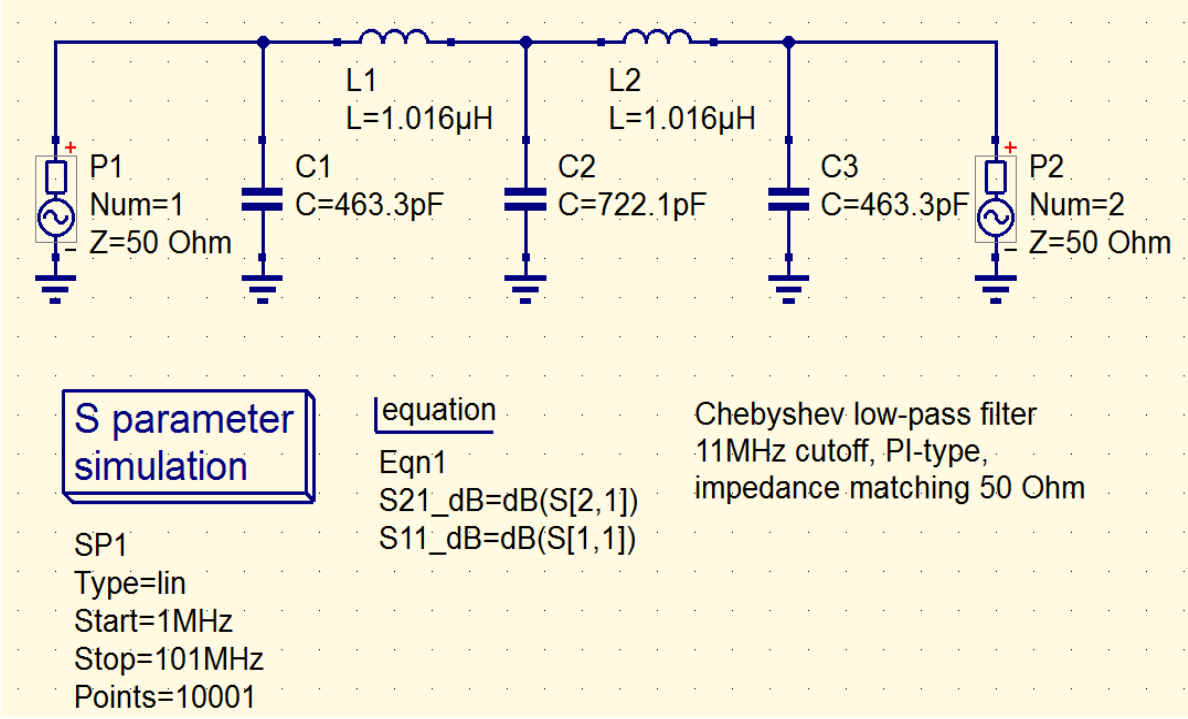
In der Übertragungstechnik: werden dauernd Filter, Verstärker, Weichen, Koppler usw. benötigt, entwickelt und produziert. Da möchte man doch stets wissen, was hinten herauskommt, wenn man vorne etwas reinschickt.
Nehmen wir als Beispiel einen **Tiefpass mit der Grenzfrequenz von 11 MHz** (…denn den benötigt man, um Signale oberhalb der oft bei RF-Empfängern benützten „Zwischenfrequenz“ von 10,7 MHz zu unterdrücken).
Bei allen diesen Bausteinen der Übertragungstechnik arbeitet man am Eingang und Ausgang mit demselben Systemwiderstand (meist: 50Ω) und beschreibt die Eigenschaften durch die „**S-Parameter“** (= Streuparameter = scattering parameters). **Man versucht also stets, den Idealfall der Leistungsanpassung zu erreichen und die S-Parameter zeigen, wie gut das glückt.**
Das Prinzip ist ganz einfach:

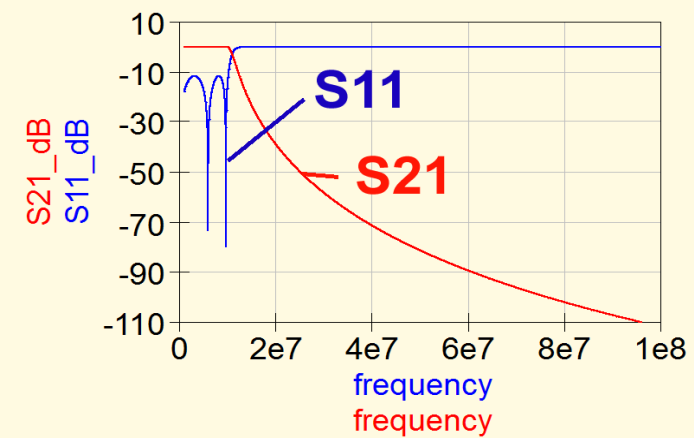
Man schickt auf einem 50 Ω-Kabel das Ausgangssignal eines Generators mit 50 Ω Innenwiderstand zum Prüfling. Es hat die Amplitude „Halbe Urspannung“ (wegen Ri = Ra) und löst bei der Ankunft am Prüfling **zwei Reaktionen** aus:

a) Weicht der Eingangswiderstand von 50 Ω ab, dann gibt es ein **Echo** und diese „nicht abgebbare Energie“ läuft auf dem Kabel zum Generator zurück. Sie wird durch die **Input Reflection S11** ausgedrückt.

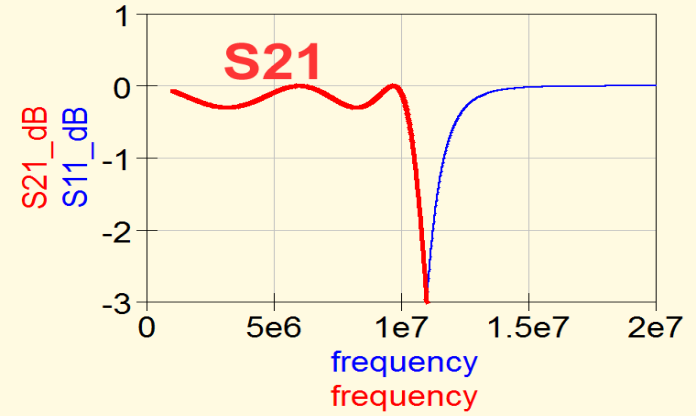
b) Der Rest der ankommenden Energie tritt in den Baustein ein und erzeugt dort ein Ausgangssignal, das größer oder kleiner sein kann. **Diese Reaktion wird durch die „Forward Transmission S21“ beschrieben und sie ist die übliche Methode zur Angabe einer Übertragungsfunktion in der Nachrichtentechnik.**

Das Ganze wird dann auf der Ausgangsseite wiederholt und liefert die beiden restlichen Parameter S22 und S12 bei einem „Twoport“.
Das ist die mit qucsstudio erstellte Schaltung des Tiefpasses, wie sie für die S-Parameter-Simulation vorbereitet wird.





Hier ist das Simulationsergebnis von S11 und S21 im Frequenzbereich von 1 kHz bis 100 MHz zu sehen.



Die „Tschebyschef-Wellen von S21“ tauchen erst nach kräftigem Zoomen auf.

Doch nun zur Frage:

Wie bestimmt man in der Praxis den Parameter S21 als „Forward Transmission“?

Da stehen prinzipiell drei Möglichkeiten zur Verfügung:

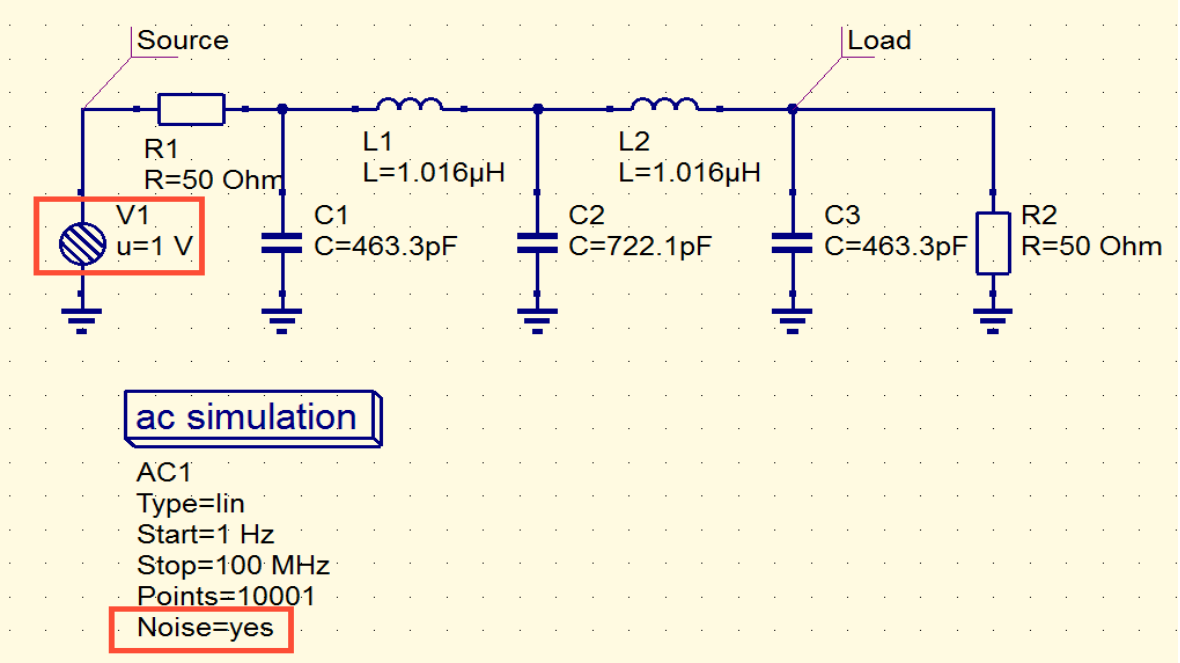
a) Mit der **S-Parameter-Simulation bzw. mit dem Vektoriellen Network Analyzer**, bei dem die Frequenz des speisenden Generators von der tiefsten bis zur höchsten Frequenz langsam **„gesweept“** wird. Das entspricht unserem simulierten Beispiel.

b) Mit einem „**Dirac-Impuls**“ am Eingang, der bekanntlich kein Linienspektrum besitzt, sondern den vollständigen Frequenzbereich lückenlos mit Energie belegt (…folglich ist die Spektrale Leistungsdichte über der Frequenz konstant). Die praktische Umsetzung ist allerdings sehr aufwendig. Nicht nur wegen der hohen Anforderungen an die nötigen speziellen Messgeneratoren und den Messaufbau, sondern auch wegen der Messung und Auswertung der „Impulsantwort“ in der Time Domain. Aus ihr muss nämlich anschließend über die Fast Fourier Transformation (FFT) der Frequenzgang der Übertragungsfunktion **berechnet** werden.

c) Da **Weißes Rauschen** (ähnlich wie der Dirac-Impuls) ebenfalls ein Spektrum mit konstanter Leistungsdichte aufweist, bietet sich diese Methode an, wenn der Eingang einer Schaltung durch eine zu hohe Dirac-Impuls-Amplitude gefährdet ist. Am Ausgang des Messobjektes muss dann mit einem Spektrum-Analyzer die Spektrale Leistungsdichte des Rauschsignal-Outputs an jeder Stelle des vorgesehenen Frequenzbereiches gemessen werden.

18.4.2. Ermittlung von S21 mit weißem Rauschen in der Simulation

Wir wollen uns nun mit der Methode „c)“ etwas ausführlicher beschäftigen, denn auch sie eignet sich hervorragend zur Simulation. Wir verwenden dazu die vorige Schaltung und wandeln sie folgendermaßen ab:

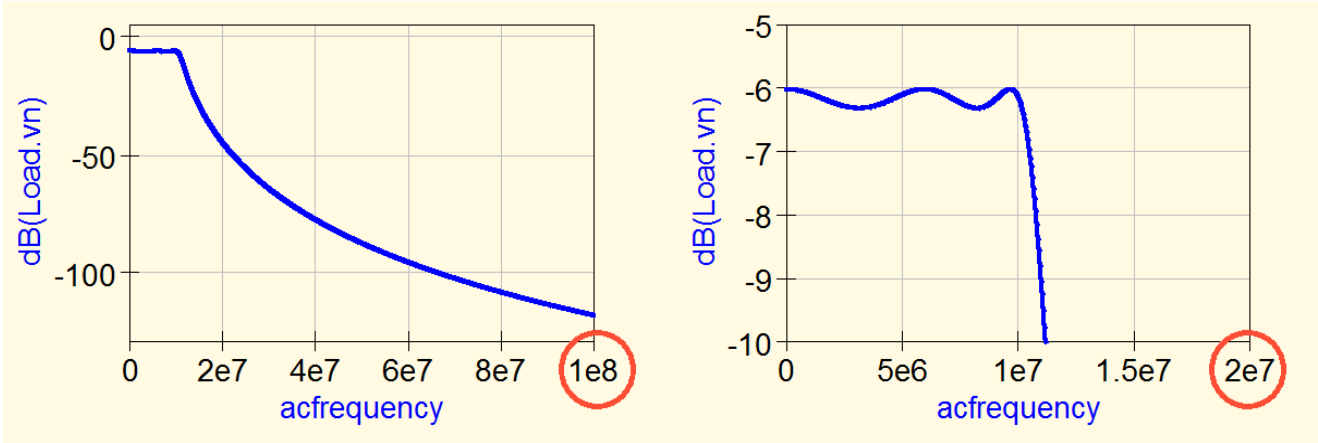


a) Am Ausgang bringen wir ein **Label „Load“, direkt an der Rauschspannungsquelle ein Label „Source“** an.

b) Die Rauschspannungsquelle findet sich im „Components“ - Menü, ihre Urspannung wird auf „1 V / sqrt(1Hz)“ eingestellt. **Dieser hohe Spannungswert ist sowohl in der Simulation wie auch in der Praxis wichtig, damit das Eigenrauschen der in der Schaltung vorhandenen Widerstände keine Rolle spielt!**

c) Die AC-Simulation soll von 1 Hz bis 100 MHz mit 10001 Punkten arbeiten. Dazu wird die Rauschsimulation freigegeben.

Dann wird simuliert und in einem kartesischen Diagramm „dB(Load.Vt)“ dargestellt



Im linken Bild sehen wir S21 den Bereich von Null Hz bis 100 MHz mit einem Dynamikumfang von ca. 150 dB (...das ist die „Weitab-Selektion“). Das rechte Bild reicht nur bis 20 MHz und soll die Tschebyschef-Wellen im Durchlass - Bereich zeigen.

Das sieht ja schon recht schön aus, aber als Bezugswert bei der dB-Berechnung diente „1 V“.

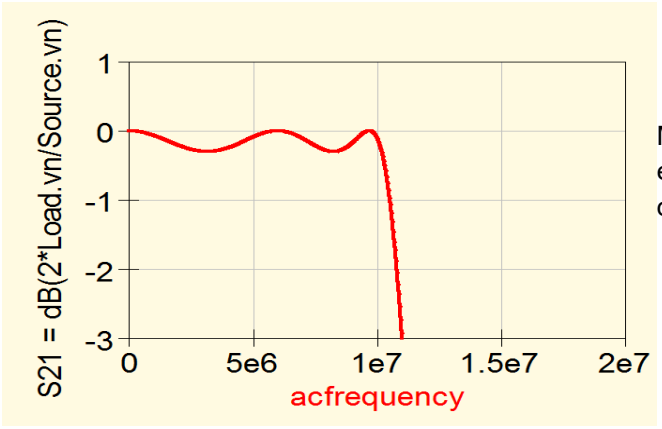
Wollen wir dagegen den korrekten Parameter „S21“ darstellen, dann müssen wir uns an die Definition und Formel halten:

S21 = (Spannung U_{Load} am Widerstand R2) / Hinlaufende Welle = U_{Load} / (halbe Urspannung)

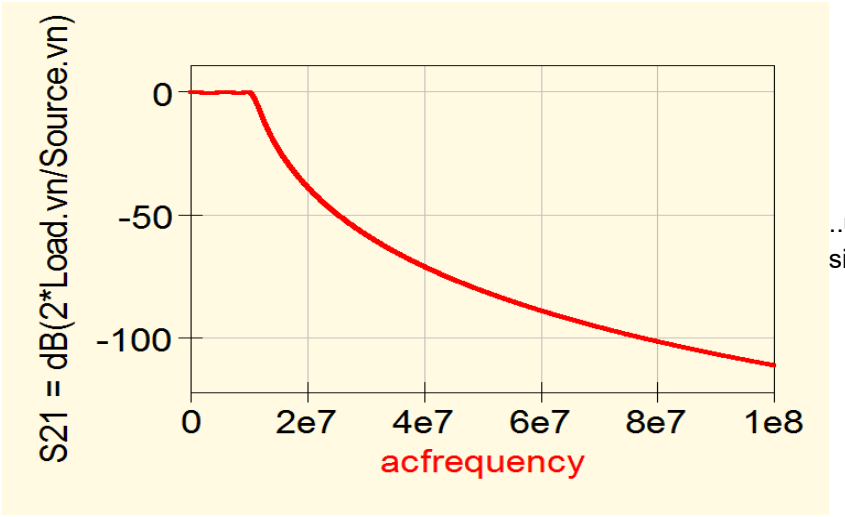
...und das Ganze am Ende in dB....

Also gehen wir in unser kartesisches Diagramm und tragen oben links folgende „Graph Properties“ ein;

dB(2*Load.vn / Source.vn)



Mit einem Frequenzbereich von 0....20 MHz und einem Amplitudenbereich von +1....-3 dB sieht das dann so aus, wie wir wollen.....



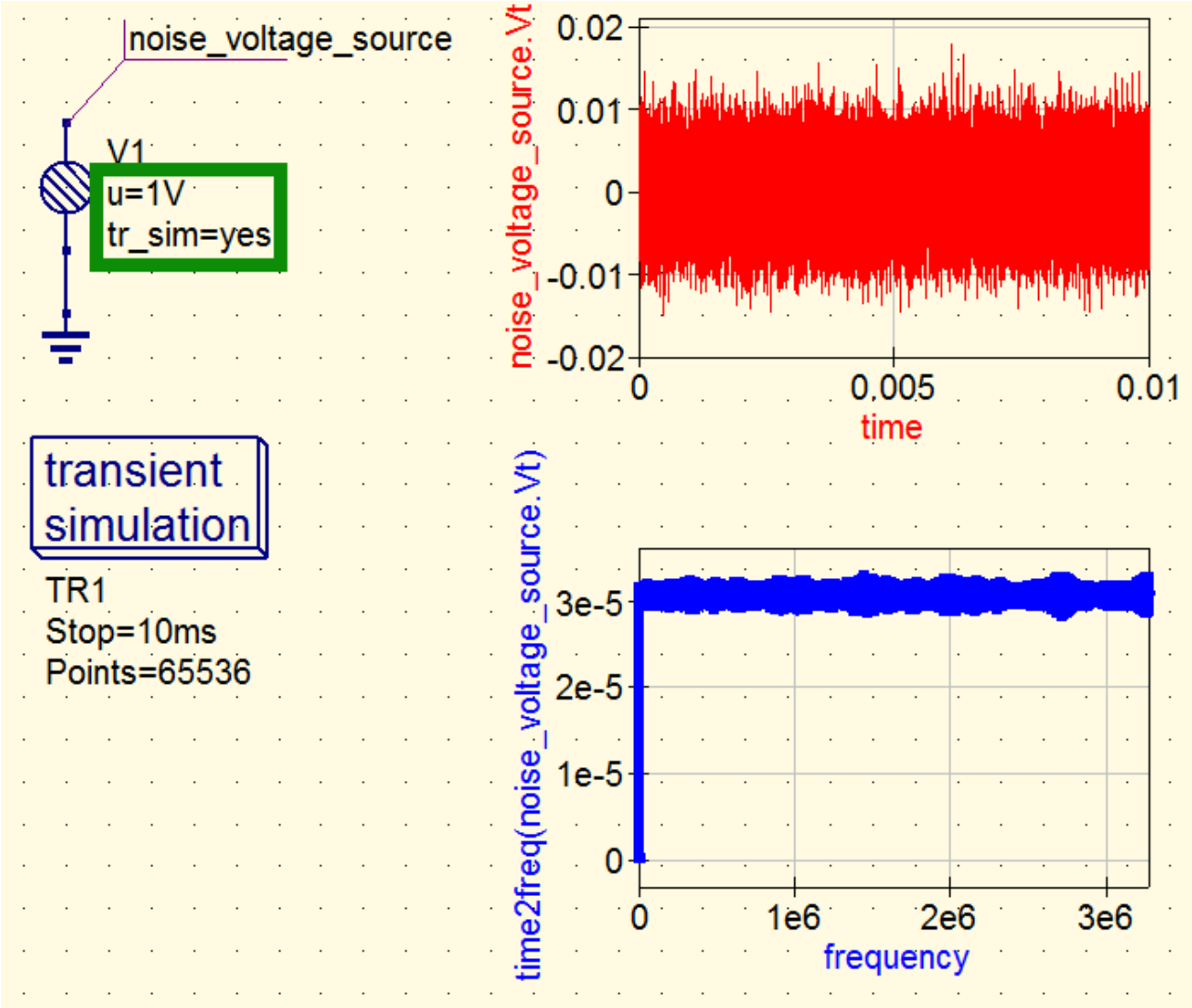
..und auch die S21-Weitab-Selektion sieht aus wie im Datenblatt.

18.5. Erzeugung eines Rauschsignals im Zeitbereich mit qucsstudio

Das interessiert direkt den Schaltungsentwickler, denn so etwas ist oft sehr praktisch. Man kann dadurch z. B. unregelmäßige Schwankungen einer Versorgungsspannung nachbilden, bei AD-Wandlern die Linearität verbessern, die „Bit-Error-Rate“ eines Systems untersuchen usw.

Da gibt es im Bauteilvorrat (Unterordner = Quellen) eine hübsche Rauschspannungsquelle für solche Simulationen im Zeitbereich. Mit ihrer lässt sich „Weisses Rauschen“ erzeugen (...und ggf. über entsprechende Koeffizienten im Eigenschaften-Menü auch „Rosa Rauschen“).

Allerdings stellen wir zuerst bei der Quelle die **„Spannung pro Wurzel aus 1 Hz“ auf einen Wert von 1V** ein. Außerdem muss man noch die **Simulation des Rauschens bei Simulationen im Zeitbereich (= Transient Simulation) freigeben**.



Das unten rechts abgebildete Spektrum zeigt sehr schön die zum „**Weißen Rauschen“ gehörende konstante Spannung über der Frequenz, die zu einer konstanten Leistungsdichte gehört.**

Wer möchte, der kann den Verlauf der spektralen Leistungsdichte durch Quadrieren der Spannung erhalten.

Ein Tipp:
Wer für andere Zwecke die **komplette Folge der simulierten Werte als Textfile** benötigt, der gehe so vor:
Wir simulieren nochmals für die Zeit von 0....10 ms mit einer Auflösung von 10001 Punkten. Dann erhalten wir Zeitschritte im Abstand von einer Mikrosekunde.

time	noise_voltage_source.Vt
0	-0.00728
1e-6	0.0157
2e-6	0.00441
3e-6	0.0144
4e-6	0.000403
5e-6	-0.00793
6e-6	-0.0108
7e-6	-0.0108
8e-6	0.00597
9e-6	-0.0183
1e-5	-0.00326
1.1e-5	-0.00997
1.2e-5	-0.000999
1.3e-5	0.0152
1.4e-5	0.00676
1.5e-5	0.0213

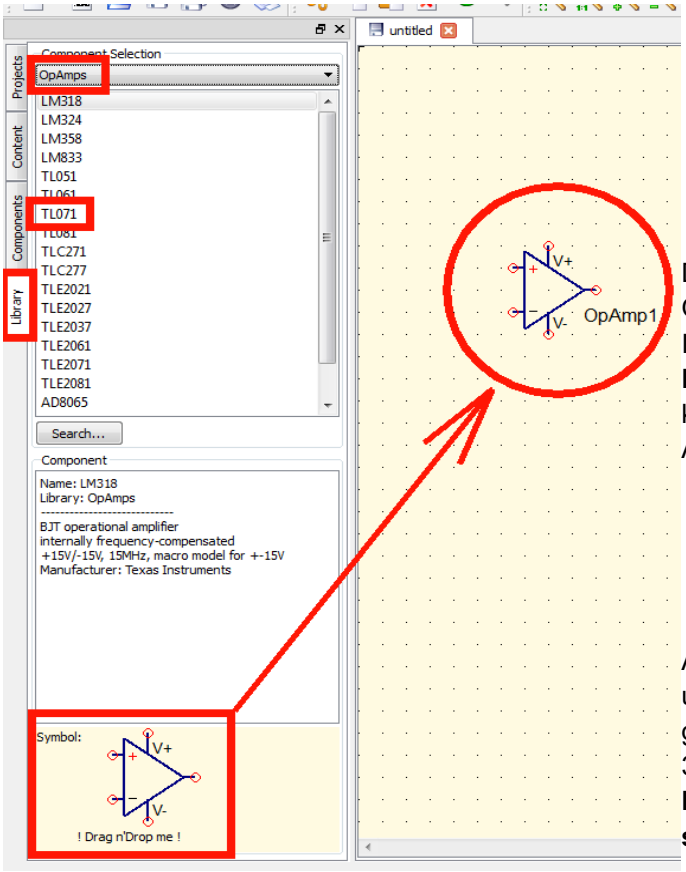
Nach der Simulation wird aus dem Diagramm-Angebot die Tabelle ausgewählt und in ihren Eigenschaften die Darstellung der Spannung „noise_voltage_source“ vorgegeben. Die anschließend auftauchende Tabelle markiert man und zieht mit der Maus „die obere Ecke nach rechts“ und falls nötig, die „Tabelle weiter nach unten“. Damit schafft man eine vernünftige Anzeige.

Soll dieser Verlauf in einer Datei gespeichert werden, dann klickt man zuerst direkt auf den Verlauf der Spannung innerhalb des Diagramms, um sie zu markieren. Anschließend sucht man oben in der Menüleiste nach „Project“ und klickt darin auf „Export to CSV“. Das Programm fragt dann nach dem Dateinamen und dem Speicherort und erledigt anschließend die Speicherung.

Übrigens:
Bei der praktischen Anwendung in einer Schaltung kann man nun eine „dateibasierte Spannungsquelle (mit diesem *.csv-File als Vorgabe) z. B. einer Betriebsspannung oder einem ansteuernden Eingangssignal in Reihe schalten und so eine schwankende Betriebsspannung oder ein verrauschtes Eingangssignal vortäuschen.
Natürlich darf man dann nicht vergessen, auch einen Innenwiderstand bei dieser Quelle vorzusehen…

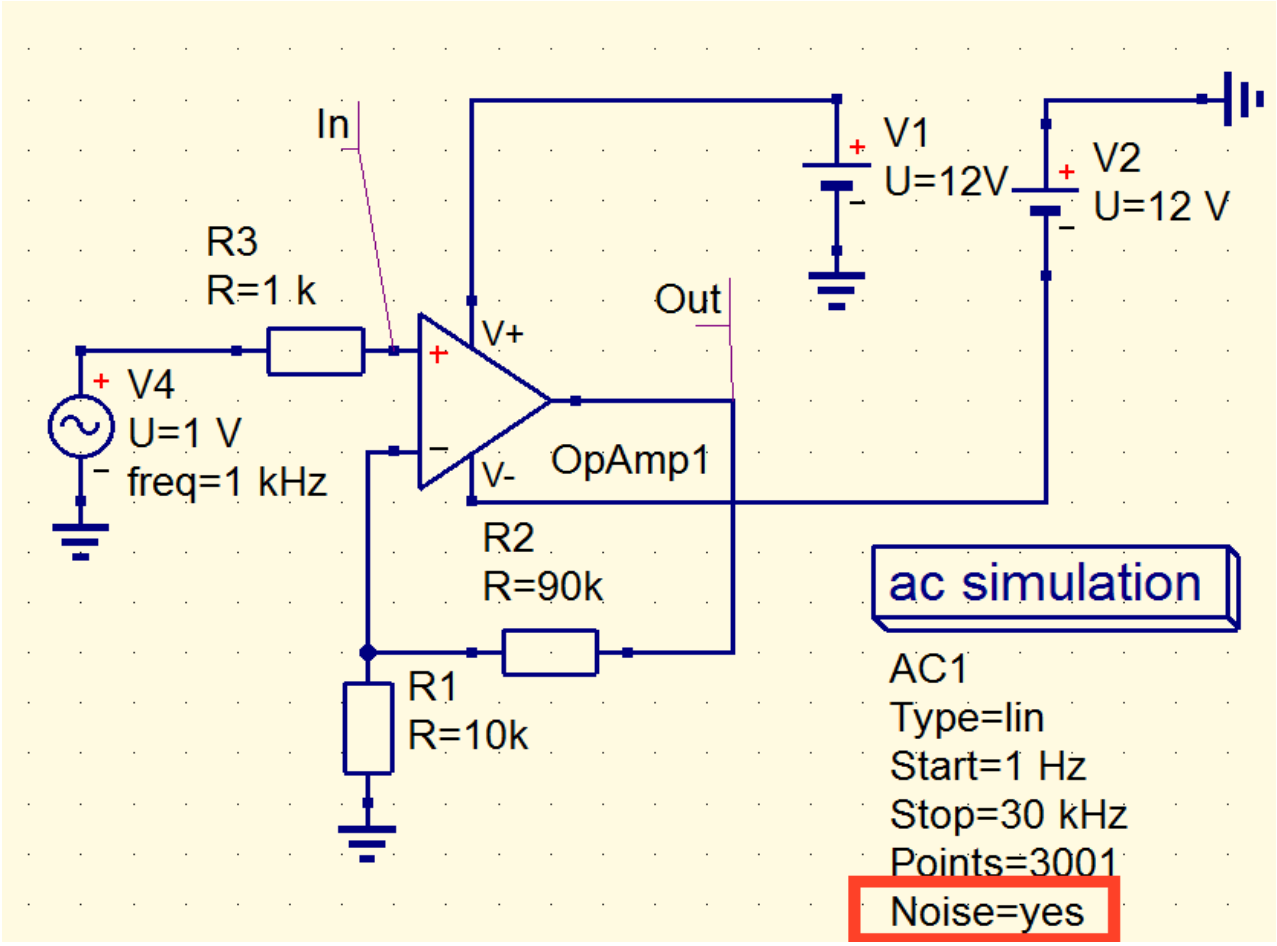
(Anmerkung:
Im Augenblick gibt es hier noch Probleme, denn Windows oder Excel fügt zwischen die berechneten Tabellenwerte immer einen Strichpunkt ein und das kann anschließend der qucsstudio-Editor nicht mehr lesen. Da kommt eine Fehlermeldung – aber der Programmautor kämpft mit dem Problem.....)
.

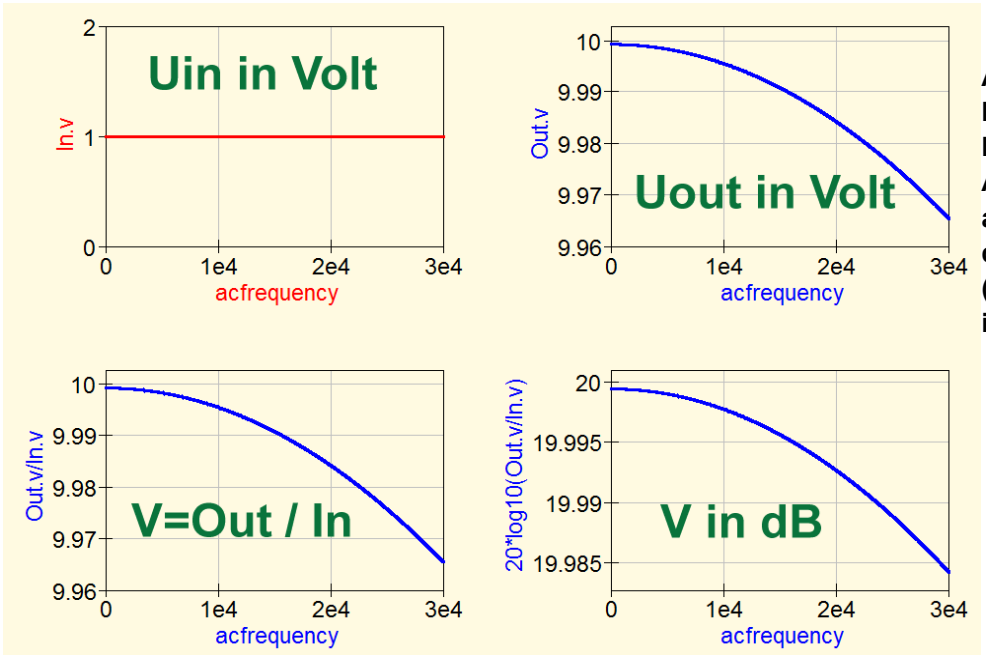
18.6. Analyse einer OPV-Schaltung (einschließlich Noise-Simulation)



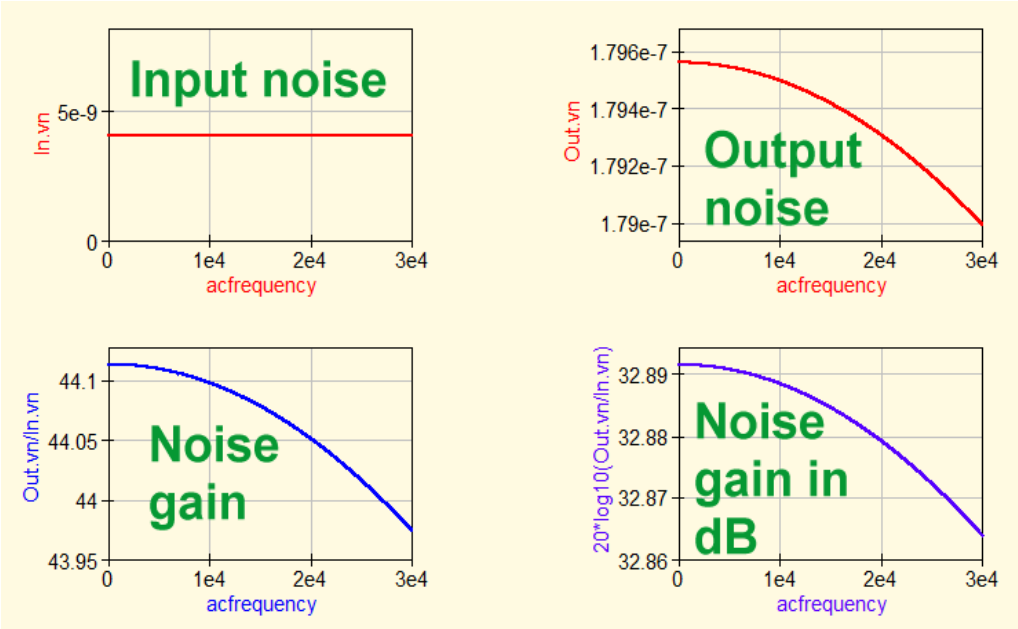
Dazu verwenden wir den bekannten rauscharmen Operationsverstärker TL071 von Texas Instruments. Er findet sich als Modell in der **Library** von qucsstudio unter **OpAmps** und wir können uns sein Symbol per „Drag'n Drop“ auf die Arbeitsfläche ziehen.

Anschließend wird folgende Schaltung (= nicht umkehrender Verstärker mit V = 10fach) gezeichnet) und ein linearer Sweep von 1 Hz bis 30 kHz programmiert. **Nicht vergessen: Bitte die Rauschsimulation (in den Properties der „ac simulation) freigeben!**





Als erstes Ergebnis lässt man sich die Eingang- und Ausgangsspannung anzeigen und daraus die Verstärkung (sowohl linear wie auch in dB) berechnen.



Nun wiederholt man dieses Spiel für die Eingangs- und die Ausgangs-Rauschspannung.

Da der OPV zusätzlich rauscht, wird die am Ausgang messbare Rauschspannung deutlich größer als (nach der Verstärkung allein) zu erwarten ist.

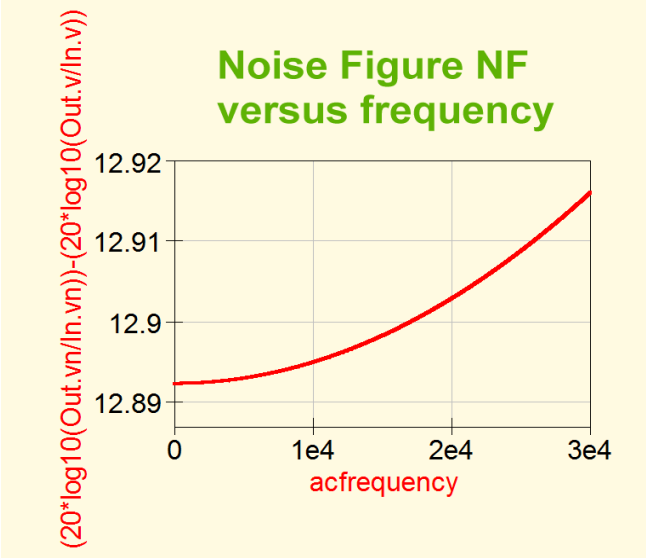
Wenn man nun die “Stufenverstärkung in dB” von der “Noise gain in dB” subtrahiert, erhält man die

Noise Figure NF des Verstärkers in dB

Der Signal / Rauschabstand am Eingang wird also genau um diese Noise figure NF am Ausgang verschlechtert!

Die nötige Formel für die “Graph Properties” des Diagramms lzur Darstellung von NF autet:

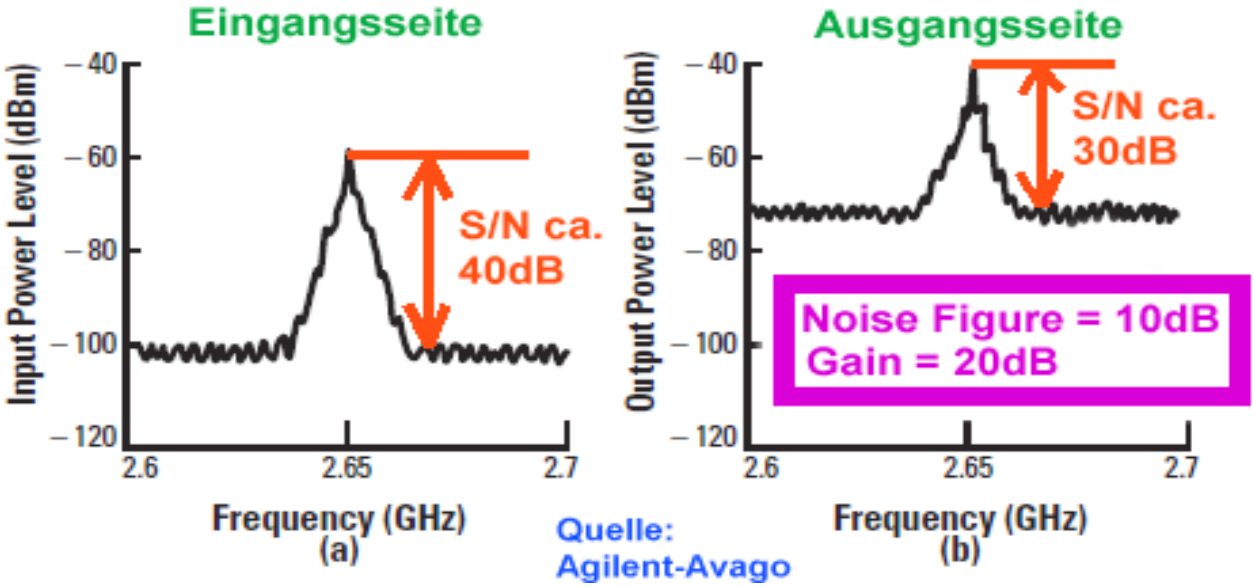
(20*log10(Out.vn/In.vn))-(20*log10(Out.v/In.v))



...und da haben wir NF in dB über der Frequenz.

Das kann man sich sehr schön an diesem Beispiel eines Verstärkers für 2,65 GHz klar machen:

Die Verstärkung beträgt 20 dB, der Signal-Rauschabstand am Eingang 40 dB.
Am Ausgang ist das Nutzsignal bei 2,65 GHz um 20 dB höher, aber der Ausgangs-Rauschpegel noch stärker angestiegen. Deshalb ist der Signal-Rauchabstand auf 30 dB gesunken und die Noise Figure beträgt NF = 10 dB.



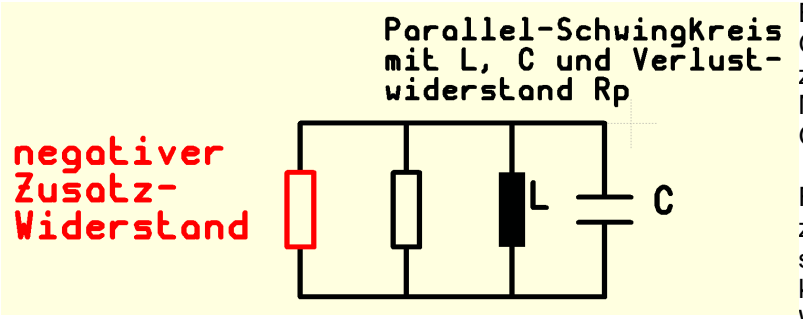
Hinweis:
Bei hohen Frequenzen arbeitet man beim Schaltungsentwurf nur mit **S-Parametern**. Die dazu vom Hersteller veröffentlichen Dateien enthalten heute meist auch die erforderlichen **Zusatzinformationen für die Berechnung der Noise Figure NF bzw. der „minimal möglichen Noise Figure NFmin.**

Das Kapitel 12 dieses qucsstudio-Tutorials enthält ein komplettes Entwicklungsbeispiel für einen solchen „Low Noise Amplifier (LNA). Bitte dort bei Bedarf die nötige Prozedur für die erforderliche Rauschanpassung nachlesen...

19. Untersuchung eines Sinus-Oszillators

19.1. Oszillator-Prinzip

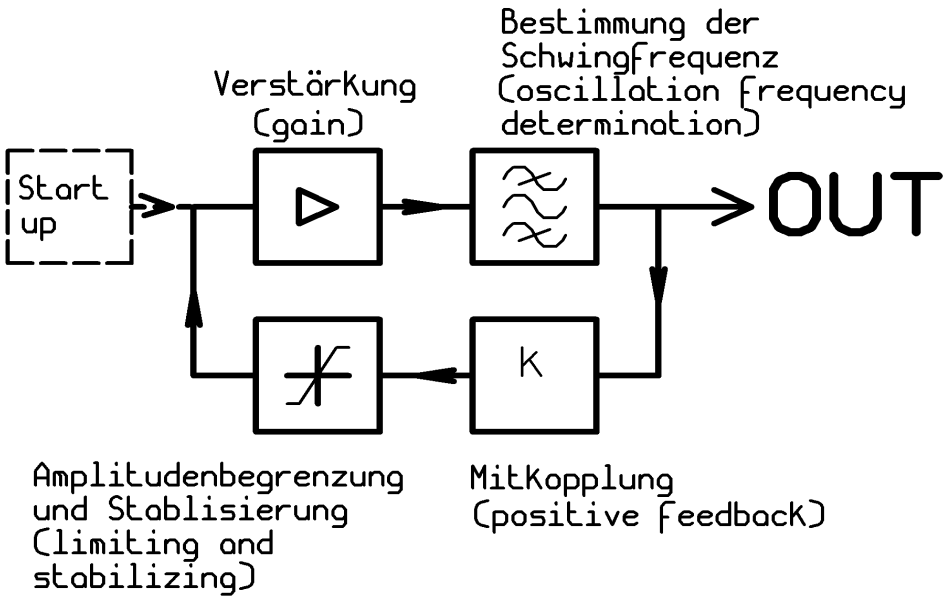
Ein Sinus-Oszillator soll – wie der Name sagt – eine sinusförmige Dauerschwingung mit nur einer Frequenz erzeugen.



Es gibt natürlich viele Oszillator-Grundschaltungen, aber wir wollen eine zuerst kurz erwähnen. Sie wird oft in der Mikrowellentechnik eingesetzt und die Grundidee ist einfach:

Führt man einem Schwingkreis Energie zu, dann entsteht eine Schwingung mit seiner Eigenfrequenz. Normalerweise klingt sie wegen der Verluste im Parallel-Widerstand schnell ab. Aber wenn man einen „**negativen Widerstand“ parallel schaltet** (...und dazu dienen passende Bauteile oder Schaltungen..), dann werden die Verluste kompensiert und bei „**Überkompensation“** bildet sich sogar eine **„anwachsende Dauerschwingung“** aus. Sie muss dann irgendwie in der Amplitude begrenzt werden.

Am häufigsten wird bei Frequenzen unter 1000 MHz jedoch diese Anordnung gewählt:



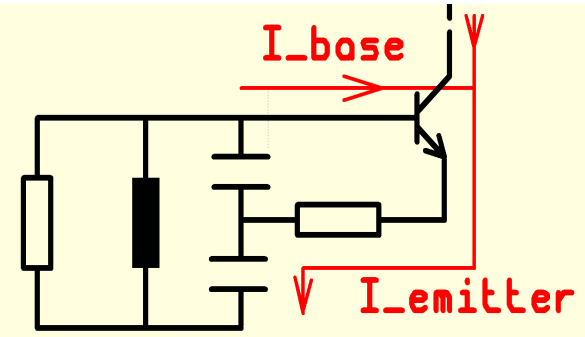
Das Prinzip ist leicht zu verstehen:
Die „Startup-Einheit“ führt dem folgenden Verstärker ein Signal mit der vorgesehenen Schwingfrequenz zu. Dieses Signal wird verstärkt und im nachgeschalteten Bandpass herausgefiltert (= „bandbegrenzt“).

Wichtig:
Dieses ausgesiebte Signal muß **gleichphasig** mit dem Eingangssignal sein und speist ein Mitkoppelnetzwerk, dem eine Amplitudenstabilisierung folgt.

Nun kommt der Anteil **„k*Uout“** **gleichphasig zurück zum Verstärkereingang** und unterstützt das Startup-Signal.
Ist die „Schwingbedingung „**k*Verstärkung >1**“ erfüllt, dann erhält man ein stetig anwachsendes Ausgangssignal. Das Startup-Signal ist nun überflüssig, **weil „die Schaltung nun selbst ihren Eingang mit der nötigen Energie versorgt“**. Diesem stetigen Anwachsen gebietet nun irgendwann die „**Amplitudenbegrenzung und Stabilisierung**“ Einhalt und **sorgt dadurch für eine konstante Amplitude der Ausgangsspannung.**

Die Sache mit der „Starteinrichtung“ ist recht einfach: dazu genügt das im Transistor des Verstärkers erzeugte Eigenrauschen, das am Verstärkerausgang gefiltert und zum Eingang zurückgeschickt wird. Darin sind auch Anteile mit der Schwingfrequenz enthalten, die dann verstärkt und erneut „im Kreis herum“ zum Eingang gelangen. Das führt zu einem exponentiellen Anwachsen der Ausgangsspannung bis zur Begrenzung.

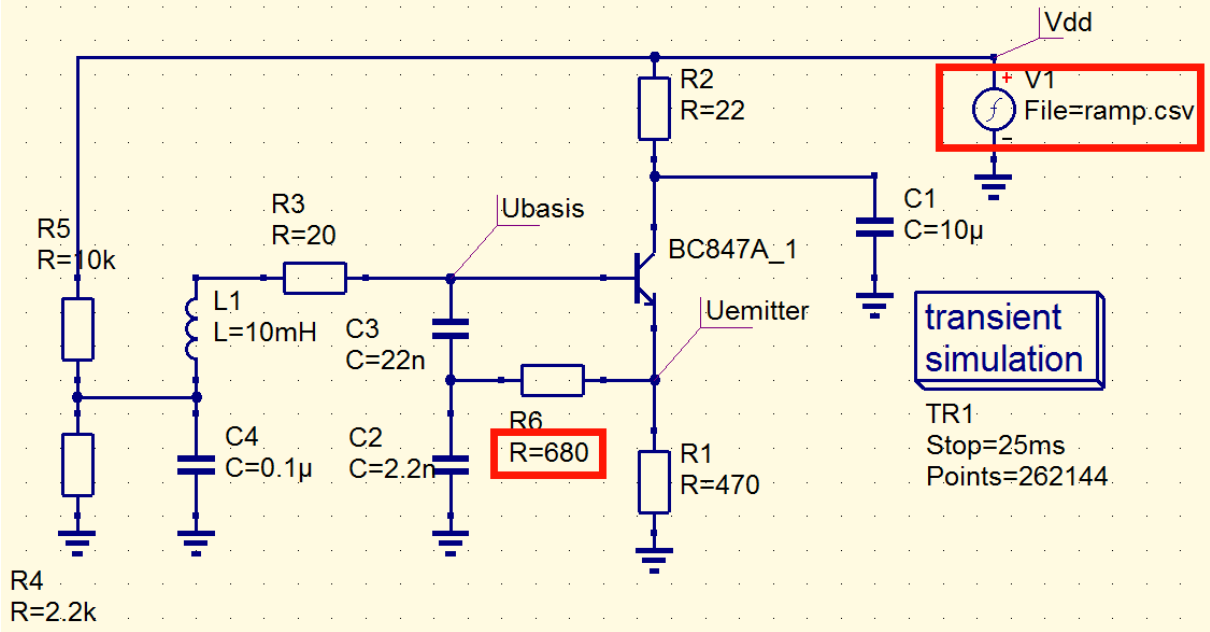
19.2. Simulierte Schaltung



Wir wollen eine **Sinus-Oszillatorschaltung (Colpitts - Oszillator mit npn - Transistor BC847C in Kollektorschaltung**) untersuchen. Hier die grundsätzliche Arbeitsweise:

Die Schwingkreisspannung liegt an der Basis des Transistors, der dadurch den Basisstrom "I_base" aufnimmt. Er stellt einen Stromverstärker dar und speist seinen (um die Stromverstärkung erhöhten) Emitterstrom in den Schwingkreis ein. Das deckt die Verluste. Der Mitkoppelgrad "k" wird über die Größe des Widerstandes zwischen Schwingkreis und Emitter eingestellt.

Und das ist die praktische Schaltung, erweitert um die Arbeitspunkt-Einstellung:

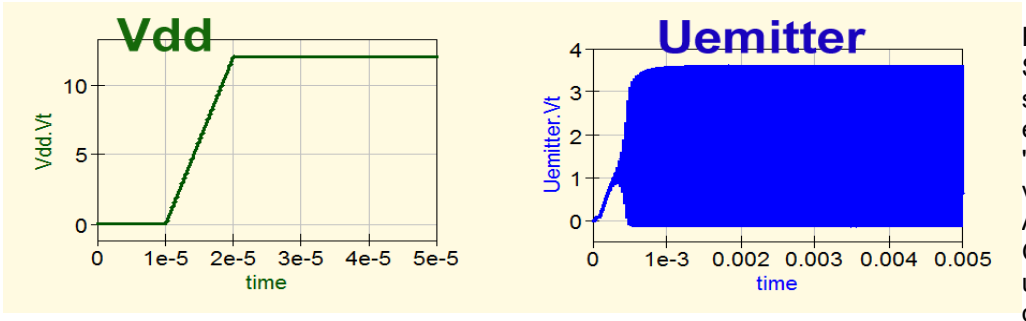


Da sieht man gleich (rot eingerahmt!) ein wichtiges Detail, denn die Simulation braucht als Startup-Signal einen **"Schubs"**, **damit die Schaltung zum Schwingen kommt.** Das wird so gelöst:

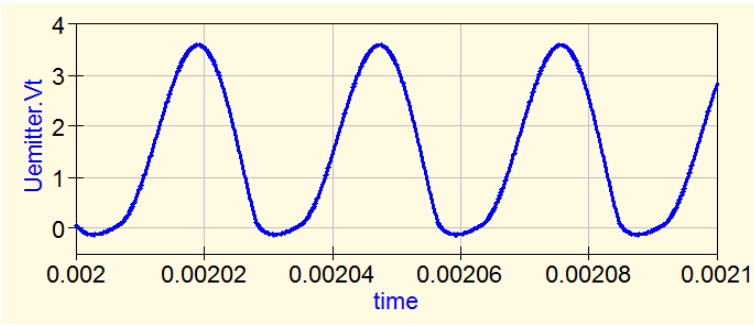
Die Versorgungsspannung Vdd springt beim Start nicht sofort auf +12 Volt, sondern wird durch eine "**dateibasierte Spannungsquelle**" (= PWL source) mit folgenden **Rampenverlauf** "hochgefahren". Er wird in der Datei "**ramp.csv**" abgelegt und sieht so aus:

```
0    0
10e-6  0
20e-6  12V
40e-6  12V
```

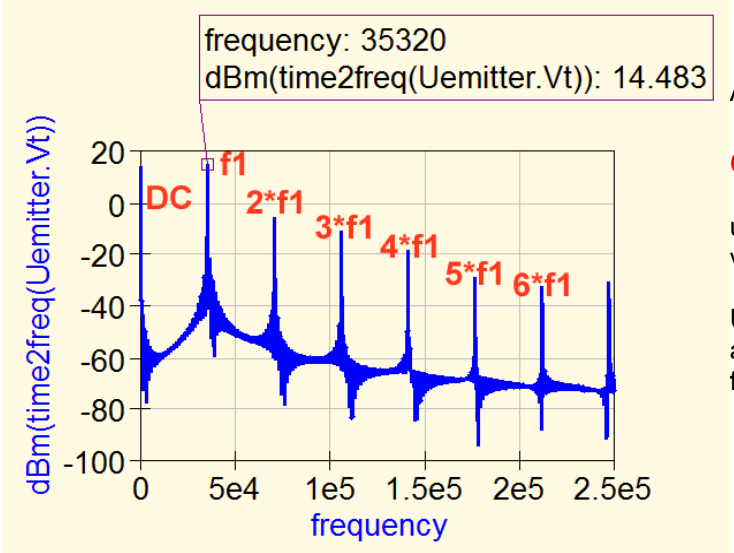
Man sieht auch, dass die Versorgungsspannung Vdd bis t = 10 µs auf Null bleibt, um dann in weiteren 10 µs linear bis auf +12V anzusteigen. Das ist der Endwert.
Jetzt wird simuliert und anschließend der Verlauf der Emitterspannung in den ersten 5 Millisekunden dargestellt:



Hier haben wir diese Spannung und sehen sehr schön das exponentielle "Anschwingen", gefolgt von der Begrenzung. Allerdings ist das Ganze etwas unsymmetrisch und das schauen wir uns näher an.



Da haben wir den Grund:
Die Mitkopplung ist viel zu stark und deshalb wird der Transistor weit in die Sättigung und Begrenzung getrieben. Die Folge ist eine stark einseitig verzerrte Kurvenform, die einen starken Oberwellenanteil befürchten läßt. Und die extreme Unsymmetrie läßt leider auch kräftige Amplituden bei den geradzahligen Harmonischen befürchten (...denn die fehlen vollständig bei perfekter Symmetrie..).



Also rufen wir die FFT auf mit

dBm(time2freq(Uemitter.Vt)

und bekommen die Bestätigung für das vermutete Spektrum.

Und durch den Einsatz des Markers sehen wir auch gleich die **Oszillator-Grundfrequenz mit f1 = 35,32 kHz.**

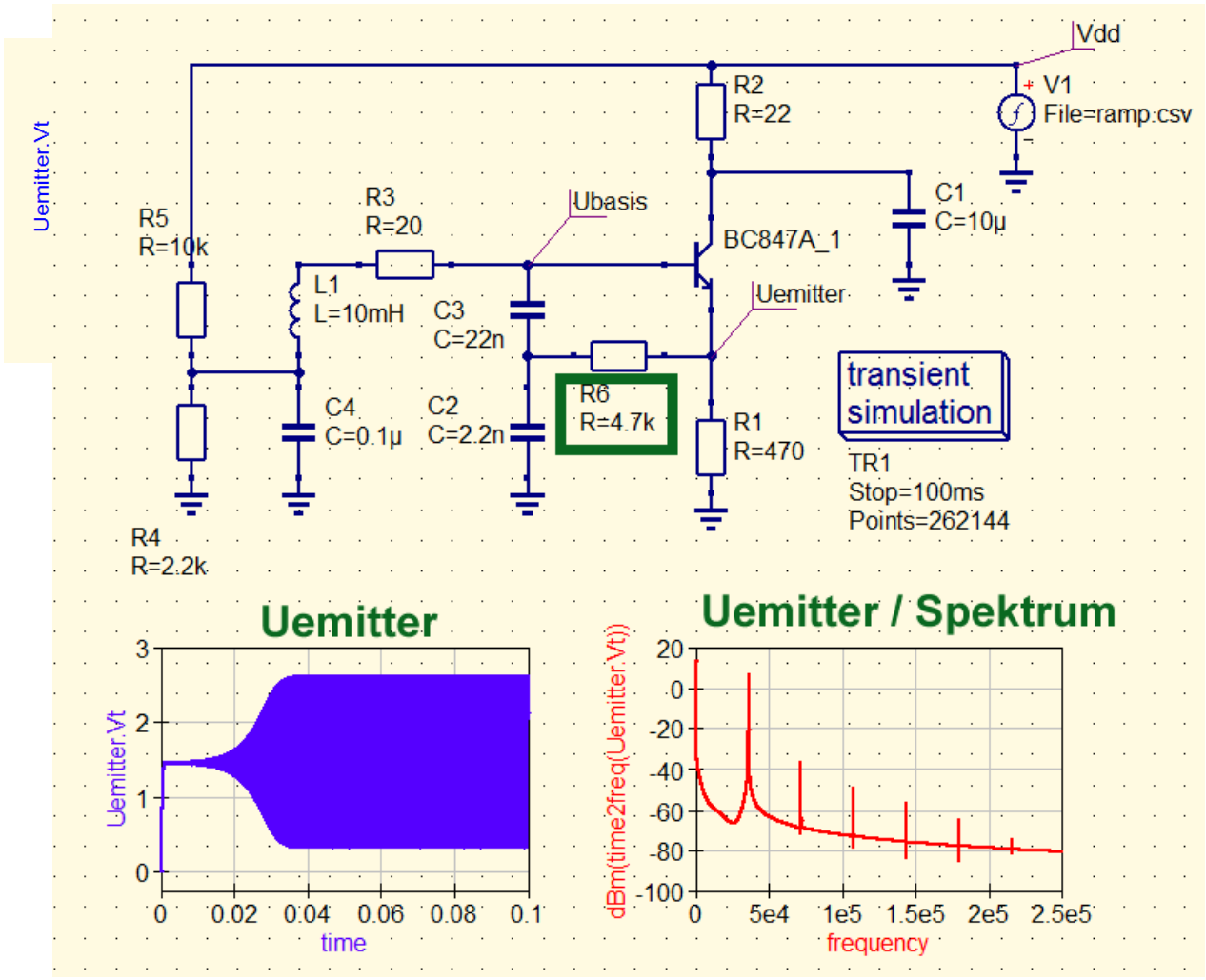
Übrigens:
Der im Schaltbild markierte Widerstand R6 / 680Ω stellt die Mitkopplung (=positive feed back) dar, denn über ihn fließt ein Teil des gleichphasigen Emitter-Wechselstromes in den Schwingkreis, deckt die Verluste und bewirkt so eine Dauerschwingung. Erhöht man seinen Wert, dann wird diese Mitkopplung schwächer. Dadurch wird zwar die Kurvenform des Ausgangssignals verbessert und der Anteil der Oberwellen nimmt ab. Aber das Anschwingen dauert nun immer länger und irgendwann reicht die Mitkopplung nicht mehr zum Ausgleich der Verluste. Dann geht überhaupt nicht mehr ....

Aufgabe:
Versuchen Sie, diesen Punkt durch wiederholtes Erhöhen von R4 und anschließender Simulation möglichst genau zu finden. Beobachten Sie dabei auch die Verbesserung der Kurvenform und die Veränderung des zugehörigen Oberwellenanteils über das simulierte Frequenzspektrum.

Beobachten Sie auch die Verlängerung des „Anschwingens“. Es dauert umso länger, je näher Sie diesem Punkt kommen – und bald läuft nichts mehr. Dieser Punkt ist bei mehr als 4,7k fast erreicht. Simulieren Sie deshalb von 0...100 Millisekunden.

Prüfen Sie diese Information nach. Bestimmen Sie am Ende das Spektrum über die FFT.

Lösung:



Man sieht, dass bei R6 = 4,7 kΩ bereits deutlich mehr als 25 Millisekunden bis zum Anstieg auf die maximale Ausgangsspannung nötig sind.

Auch im Spektrum erkennt man nun deutlich die Amplitudenreduktion bei den Oberwellen. Bei der fünffachen Grundfrequenz haben wir jetzt ca. 80 dB Dämpfung erreicht.
(Vorhin waren es 40 dB).

Und der zeitliche Verlauf ist nun vorschriftsmäßig sinusförmig