

Labor für Nachrichtentechnik an der Dualen
Hochschule Baden Württemberg in
Friedrichshafen

=====

Schaltungs- Simulation mit QucsStudio

Version 2.5.7.

Auszug aus der „**Tutorial-Version 1.7
für Neu-Einsteiger**“ (...das komplette
Tutorial mit ca. 220 Seiten findet sich
zum Download in meiner Homepage)

22. Februar 2019

Autor:

Gunthard Kraus, DG8GB, Oberstudienrat i. R.

Gast-Dozent an der Dualen Hochschule Baden-Württemberg in Friedrichshafen

Email: mail@gunthard-kraus.de

Homepage: www.gunthard-kraus.de

1. Einführung

Zunächst etwas zur Verwirrung:

Es gibt „Qucs“ und „Qucs-Studio“.

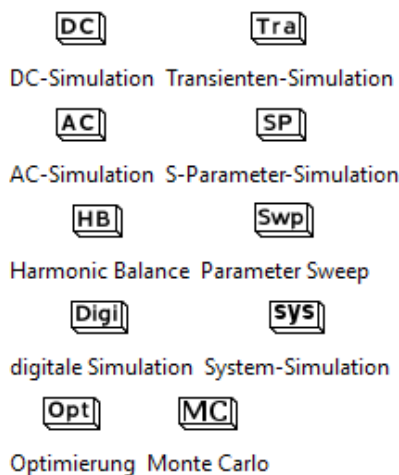
Deshalb nehmen wir uns zuerst „Qucs“ vor.

„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 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 Erbgemeinschaften: man konnte sich nicht mehr einigen und deshalb hat der Chef-Entwickler (Michael Margraf, DD6UM) irgendwann beschlossen, seinen eigenen Weg zu gehen. Er nahm zwar das Konzept mit, schrieb aber die Software neu, entwickelte weiter und nannte sie 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.....

Wichtig:

qucsstudio erfordert keine Windows-Installation oder Registrierung! Sie brauchen das Programm (im Augenblick: Version 2.5.7.) nur herunterzuladen und irgendwo (USB-Stick, Studentenverzeichnis) zu entpacken. Dann können Sie es sofort über „qucs.exe“ starten.

2. Beschaffung und Installation von QucsStudio

2.1. Installation daheim oder auf dem Notebook

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



QucsStudio

ein freier und leistungsstarker Schaltungssimulator

Herunterladen

QucsStudio gibt es zur Zeit nur für Windows®. Es läuft ohne Installation. Beim Entpacken der Archivdatei entsteht das Verzeichnis *QucsStudio*. Ausgeführt wird es durch Starten von *QucsStudio/bin/qucs.exe*.

QucsStudio-2.5.7.zip herunterladen (59 MByte)

english

Inhalt

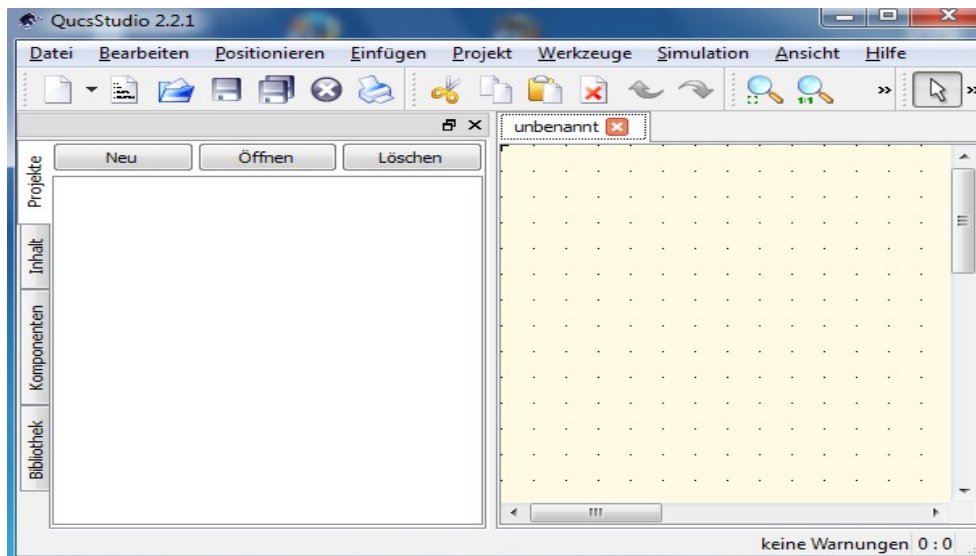
- Über
- Neuigkeiten
- Unterstützung
- Bildschirmfotos

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 wiederholten Programm-Abstürzen in Serie (...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.

2.2. Betrieb im Rechnernetz der Dualen Hochschule / Raum 029

Da gibt es eine Mail an alle Leute, die dort mit qucsstudio arbeiten wollen:

Bitte starten Sie das Programm mit der auf dem Desktop hinterlegten Batchdatei „qucs.bat“, damit verweist der Pfad zum Arbeitsverzeichnis direkt auf M:\ QucsProjekte .

Gruß

Raphaela Ulrich

3. Erstes Projekt: RC-Tiefpass

3.1. Simulation in der Frequency Domain: die Übertragungsfunktion

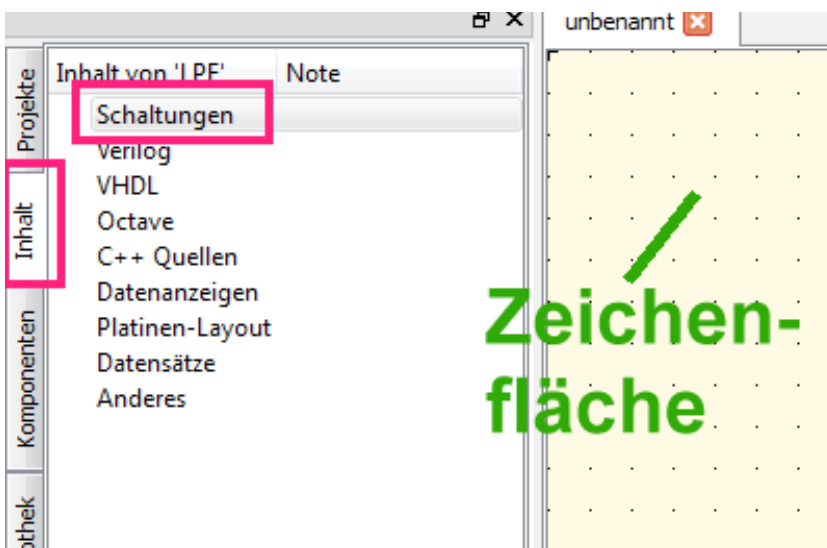
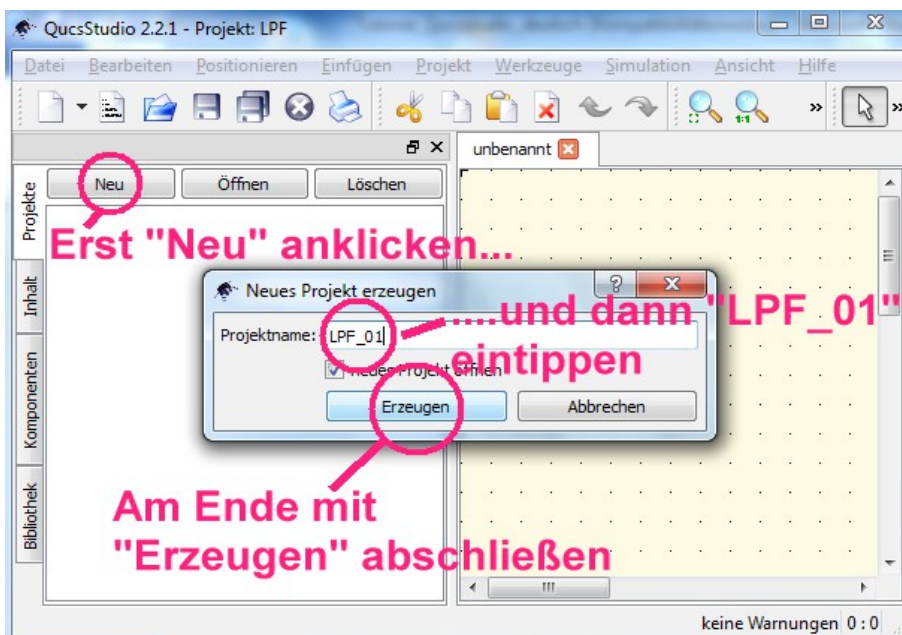
3.1.1. Lineare und logarithmische Darstellung des Simulationsergebnisses

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.



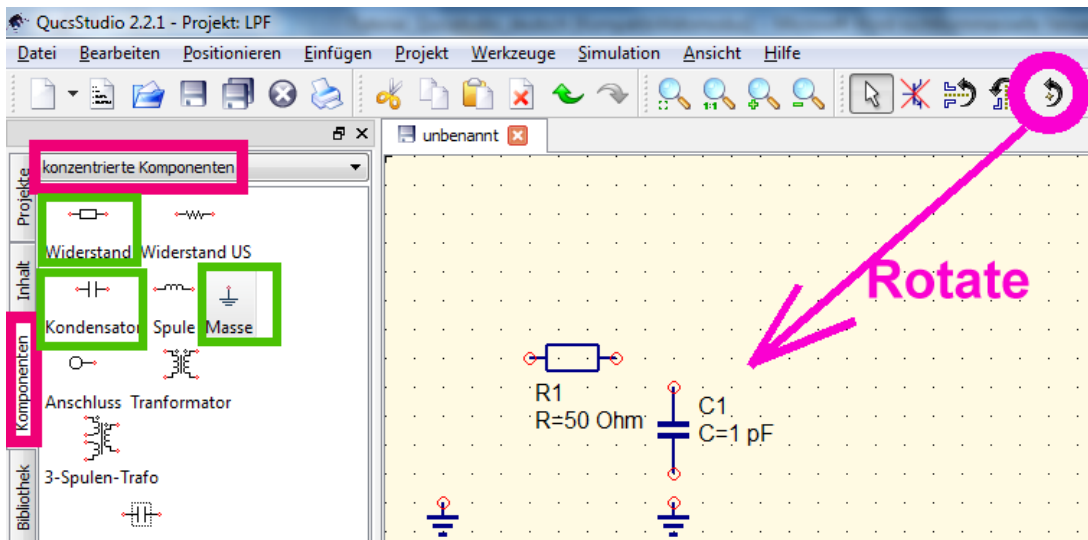
Noch ein kurzer Kontrollblick: Wurde automatisch auf „Inhalt“ umgeschaltet und die Option „**Schaltungen**“ gewählt? **Dann kann man diese neue Schaltung speichern, z.B. unter dem Namen**

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).

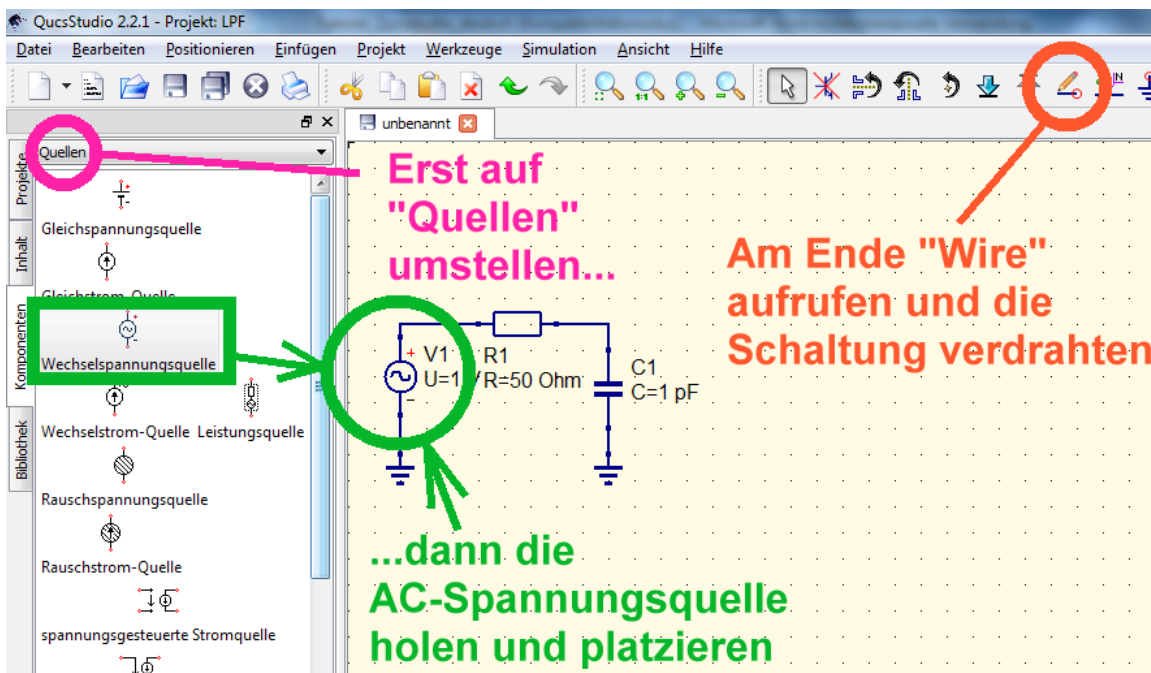
2. Schritt:

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:

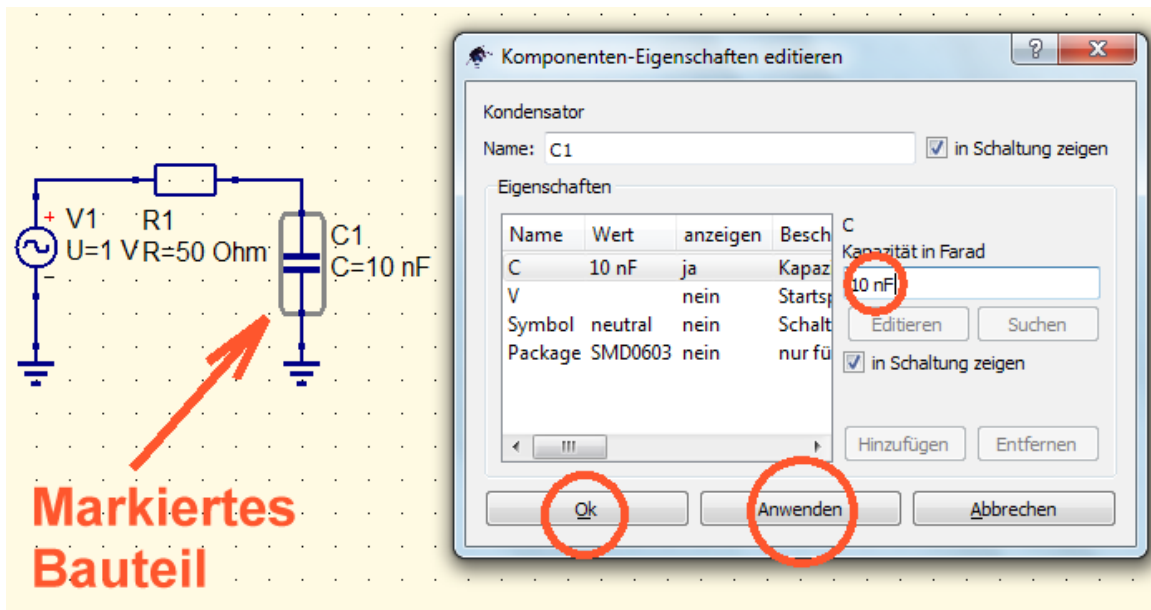


Nachdem das Auswahlmenü auf „Quellen“ umgestellt ist, kann eine **AC-Spannungsquelle** (= „Wechselspannungsquelle“) geholt und im Schaltbild platziert werden. Den Abschluss bildet die Verdrahtung mit einem „Wire“ oder „Draht“ durch einen Mausklick am Anfang und einem weiteren Mausklick nach dem Verlegen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

4. Schritt:

Am Kondensator wollen wir uns nun ansehen, wie ein Bauteilwert geändert wird.

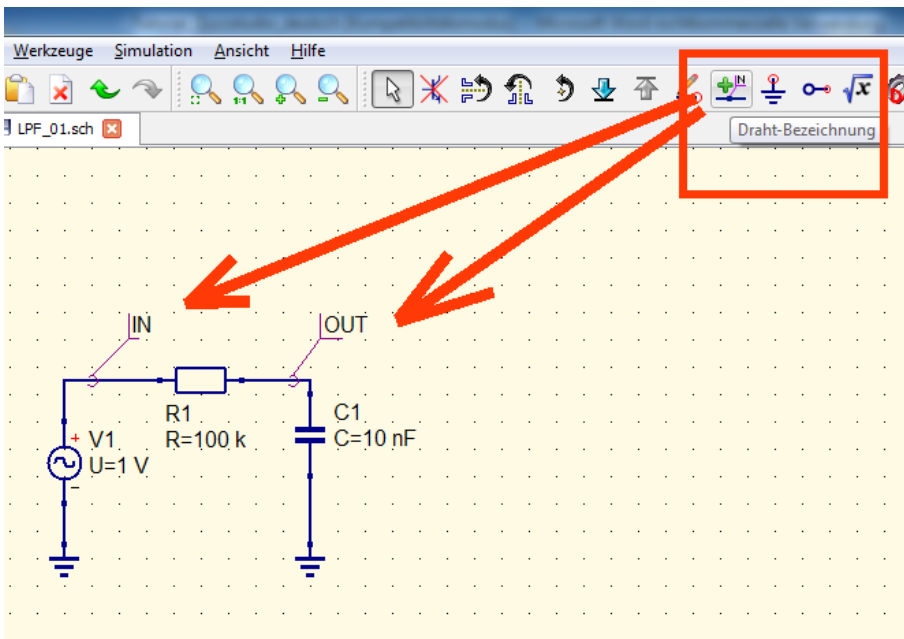
Dazu wird erst mit der linken Maustaste zweimal schnell nacheinander auf das Schaltzeichen geklickt -- oder erst mit der rechten Maustaste auf das Symbol geklickt und dann „Eigenschaften editieren“ gewählt. So wird das „**Property Menu**“ geöffnet. Dort wird im entsprechenden Fenster der Wert von 10 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 Widerstand. 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.



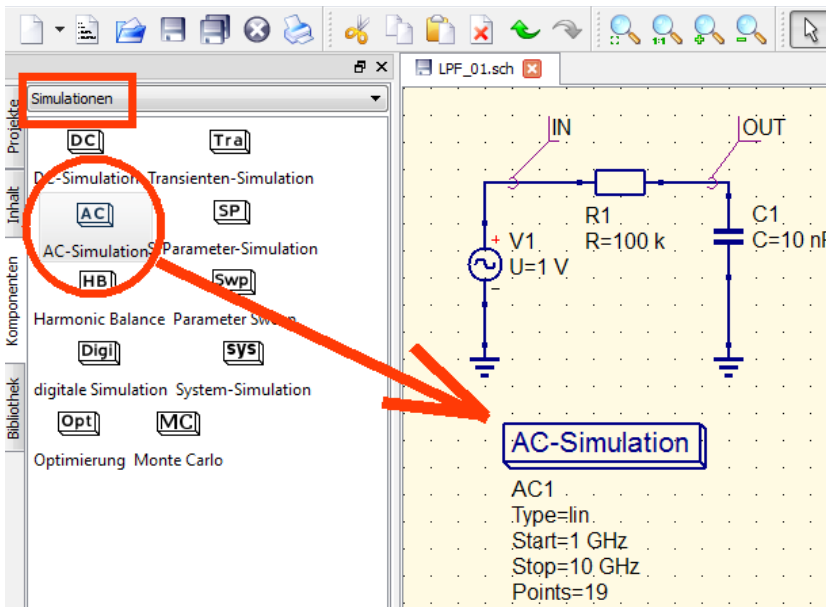
Jetzt fehlen noch „**Labels**“, also passende verständliche Bezeichnungen für wichtige Punkte in der Schaltung. **Hier sind es der Eingang (= IN) und der Ausgang (= OUT).**

Dazu gibt es einen hübschen Button „**Draht-Bezeichnung**“ 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“.

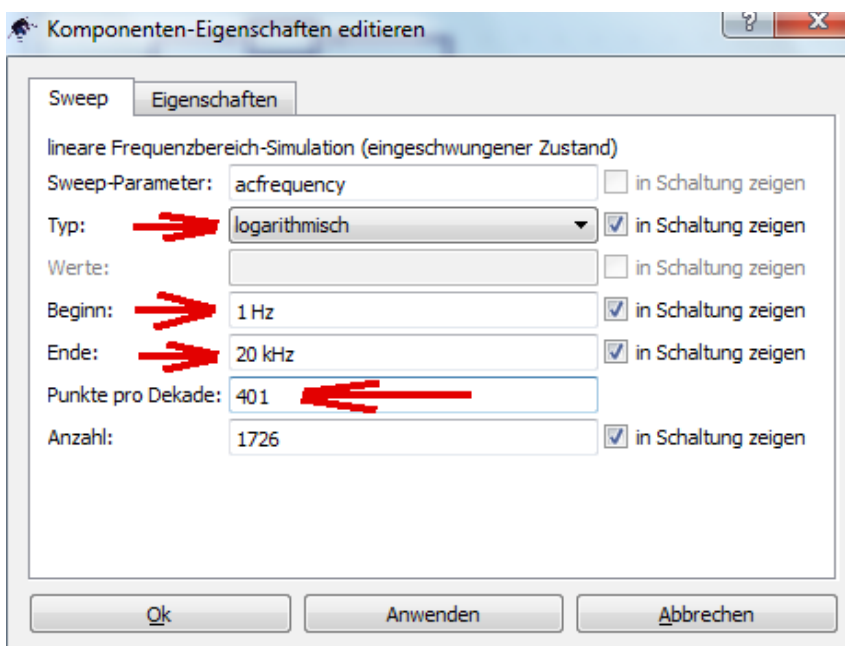
Wichtig: Ohne Labels kann man sich keine simulierten Ergebnisse zeigen lassen!



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 einen rechten Mausklick auf dieses Symbol „AC-Simulation“, gefolgt von „Eigenschaften editieren“ kommen wir an das Eigenschaftens-Menü für die **Sweep – Vorgaben** heran.



Bitte eintragen:

Logarithmisch

Start: 1 Hz

Ende: 20 kHz

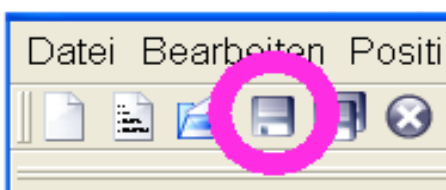
401 Punkte pro Dekade

(Die Punktezahl wird dann automatisch berechnet)

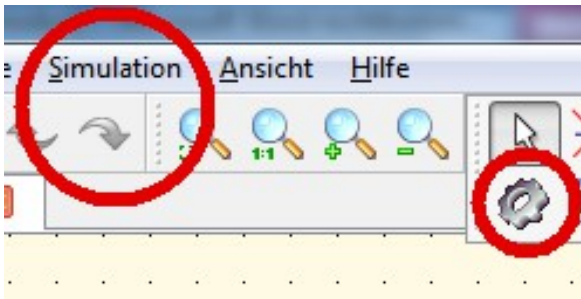
7. Schritt:

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.



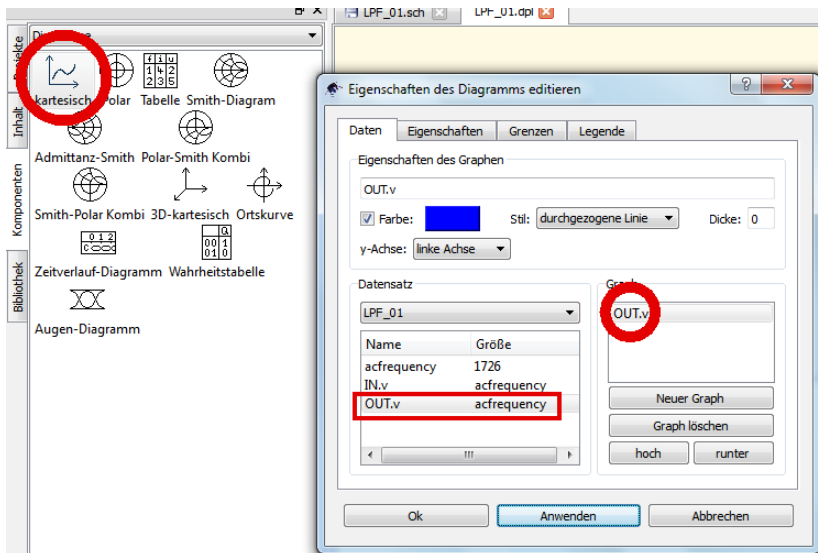
8. Schritt:

Nun wird simuliert!

Dazu sucht man **entweder** nach dem Button mit dem Zahnradchen **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

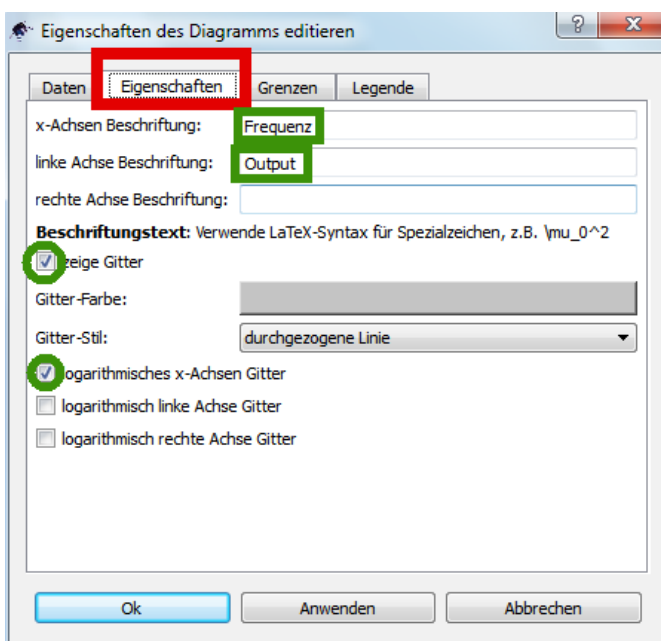
„**OUT.v**“

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** der Linie kann man jetzt ändern,

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



x-Achsen Beschriftung: „**Frequenz**“

linke Achse Beschriftung: „**Output**“

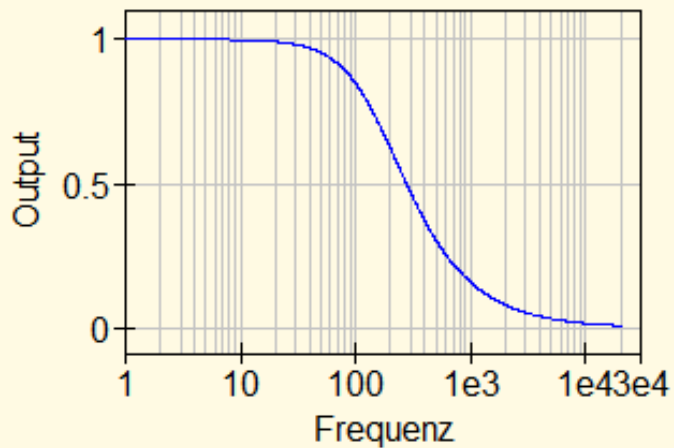
Dazu setzen wir Häkchen bei

„**zeige Gitter**“

und

logarithmisches x-Achsen-Gitter

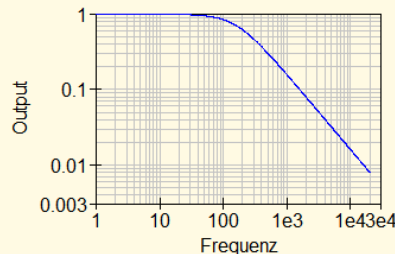
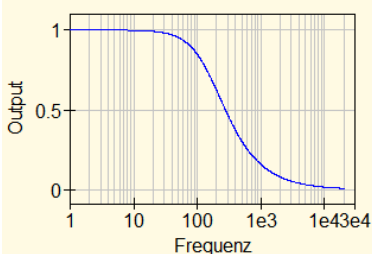
Die Spannung steigt: was passiert nun wohl nach „OK“?



Das wollten wir doch wissen!

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

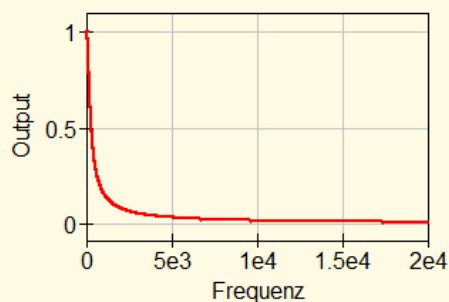
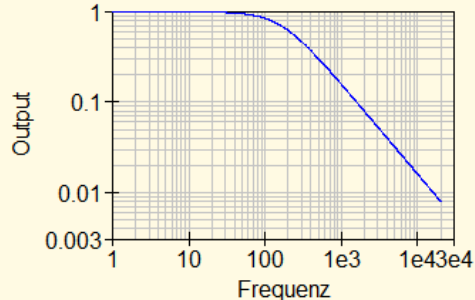
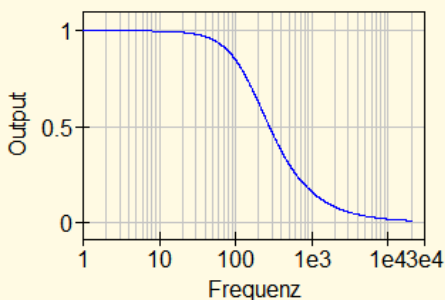
Wiederholen wir diese Prozedur und holen uns (erneut durch einen Mausklick auf „Kartesisch“ im Diagramm – Menü) ein weiteres Diagramm auf unseren Schirm. Diesmal sollen **beide Achsen logarithmisch geteilt** sein. Also füllen wir die „Daten“-Karte wie vorher aus, setzen aber auf der Karteikarte „Eigenschaften“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



So sollte die Zeichenfläche auf dem Bildschirm nun aussehen:

Zum Abschluss des Kapitels wollen wir noch (auf der Karteikarte „Eigenschaften“) **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

$$P_{out} / P_{in} \text{ in dB} = 10 * \log(P_{out} / P_{in})$$

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

$$U_{out} / U_{in} \text{ in dB} = 20 * \log(U_{out} / U_{in})$$

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

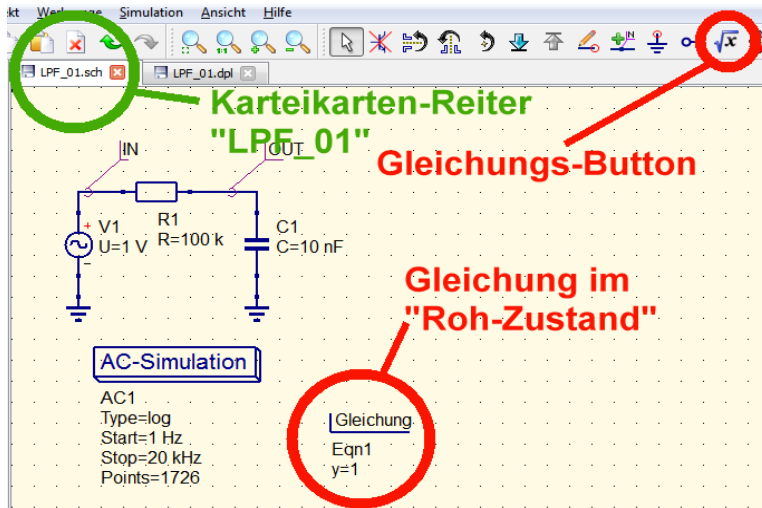
usw.

3.1.3. „dB“- und Phasen-Darstellung beim Ergebnis

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

Möglichkeit 1:

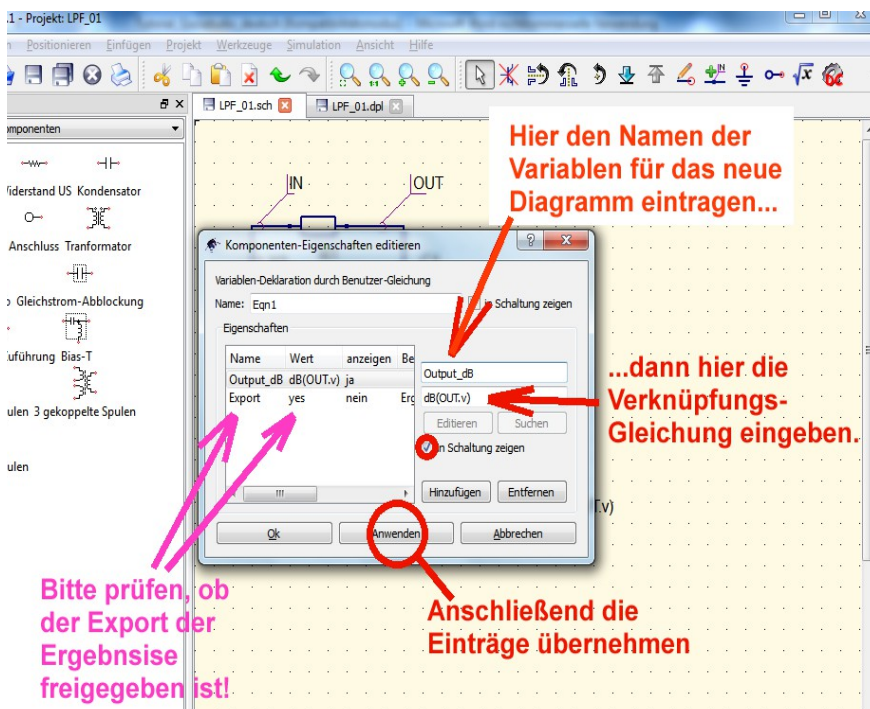
Hierfür müssen wir zum ersten Mal bereits **vor** der Simulation eine Gleichung bereitstellen, damit dieses Ergebnis gleich nach der Simulation für die Anzeige zur Verfügung steht!



Also kehren wir zu unserem Schaltbild durch Anklicken des zugehörigen

„Karteikarten-Reiters LPF_01.sch“

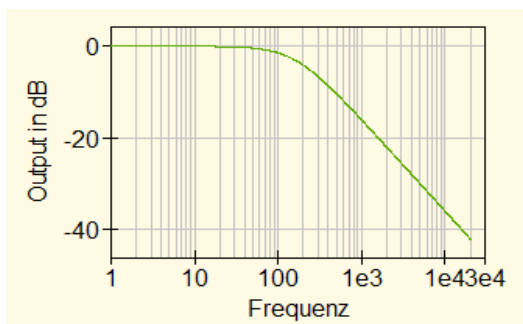
zurück und drücken auf den **Gleichungs-Button** in der Menüleiste. Dann hängt die „**Gleichung im Rohzustand**“ am Cursor und kann im Schaltbild abgesetzt werden. Auf das Wort „Gleichung“ wird nun **doppelt geklickt**, um die erforderlichen Einträge im Menü vorzunehmen.



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 Menü „**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 weiteres kartesisches Diagramm und seine Eigenschaften. Darin wird **zum Beispiel die Ausgangsspannung „OUT.Vt“** ausgewählt und markiert.

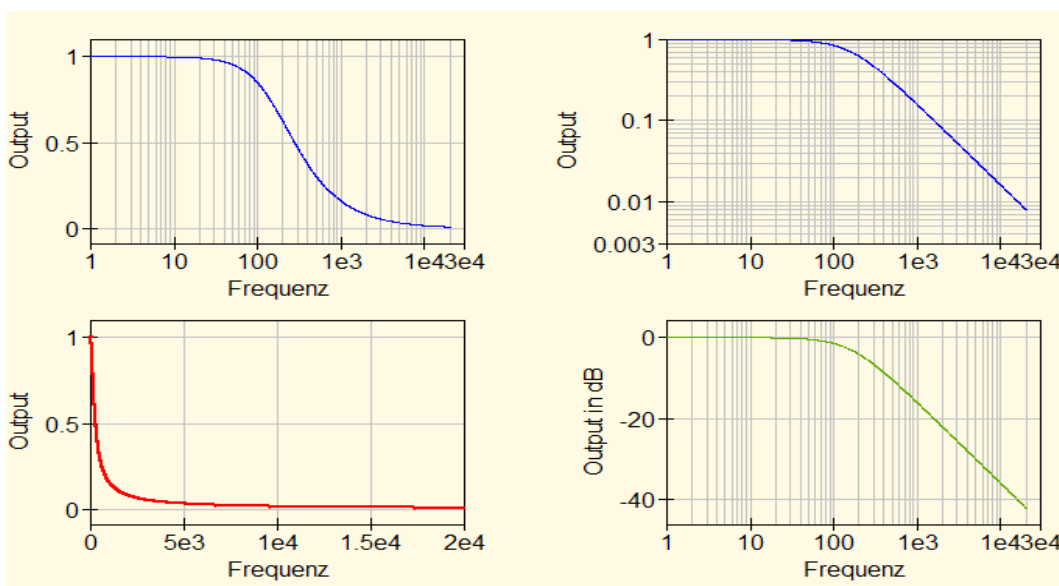
2. Schritt:

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“**.

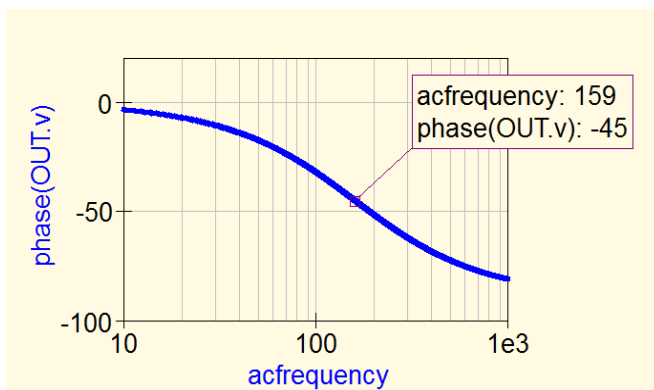
In **diese obere Zeile** stellt man ganz links den Cursor und ändert den Eintrag in

dB(OUT.Vt)

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



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. Simulation in der Time Domain

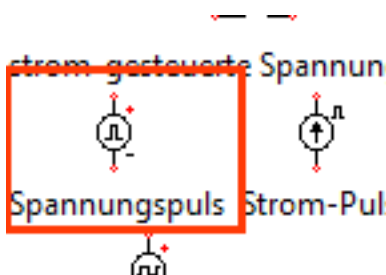
3.2.1. Impulsantwort des RC-Tiefpasses

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:

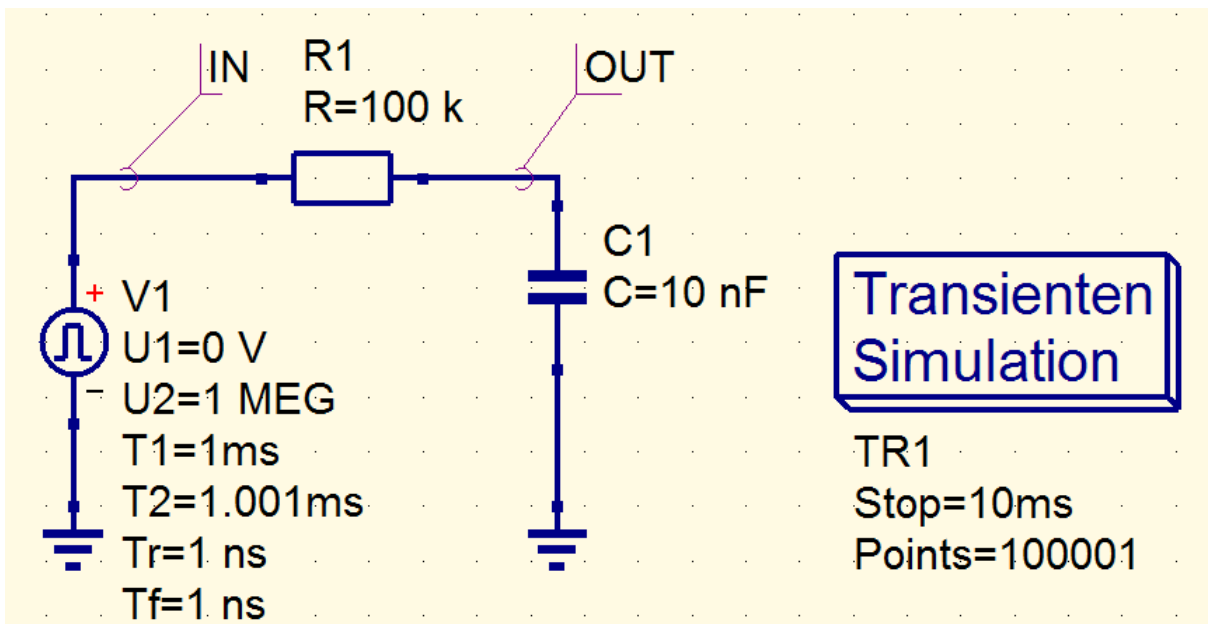
Macht man die Pulslänge wenigstens um den Faktor 100....1000 kleiner als die Zeitkonstante des untersuchten Systems und erhöht dafür um denselben Faktor die Amplitude, dann bleibt die Fläche gleich. Aber man erhält dieselben Ergebnisse wie bei der Speisung mit dem echten Dirac-Impuls.

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.

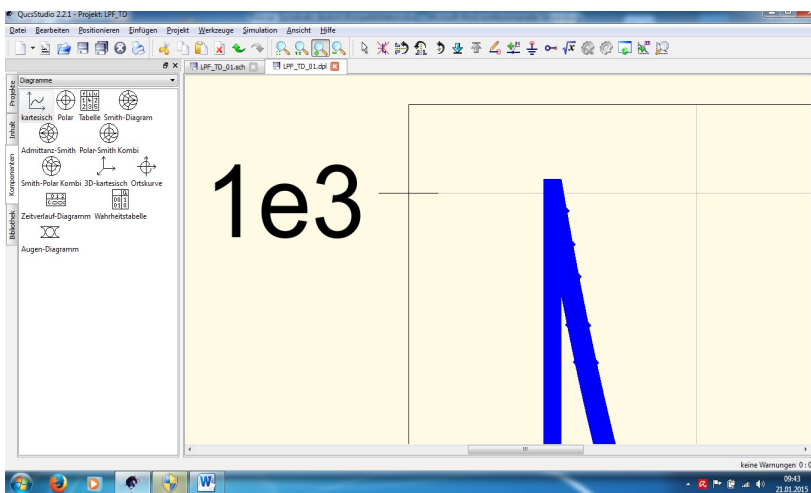
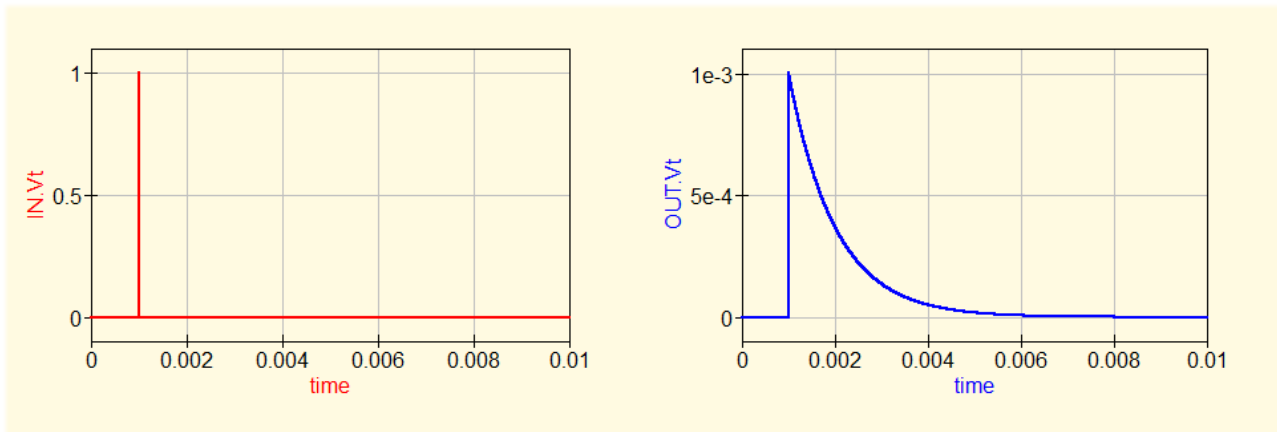


Wir brauchen nun eine **Puls-Spannungsquelle** und die finden wir in der Kartei „**Komponenten / Quellen**“. Wir arbeiten mit einer **Pulsamplitude von 1 MegaVolt** und einer **Pulslänge von 1 Mikrosekunde**. Die **Startverzögerung** sei 1 Millisekunde, **Anstiegs- und Abfallzeit** je 1 Nanosekunde.

Dann holen wir uns aus der Kartei „**Simulationen**“ die „**Transienten-Simulation**“ und setzen sie im Schaltbild ab. **Eingestellt wird eine Simulationszeit von 10 Millisekunden, die Schrittweite sei 100 Nanosekunden**. 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:



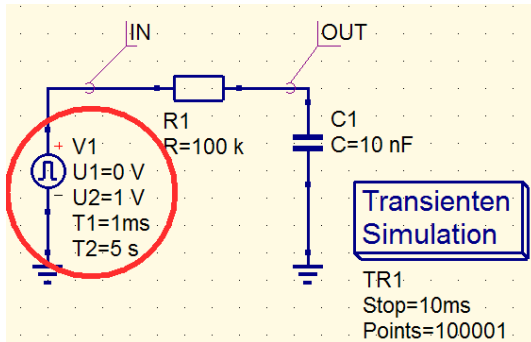
Selbstverständlich ist es nun möglich, jedes beliebige Detail des Kurvenverlaufs über den Zoom heraus zu holen.

Das zeigt dieses Beispiel.

3.2.2. Sprungantwort des Tiefpasses

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

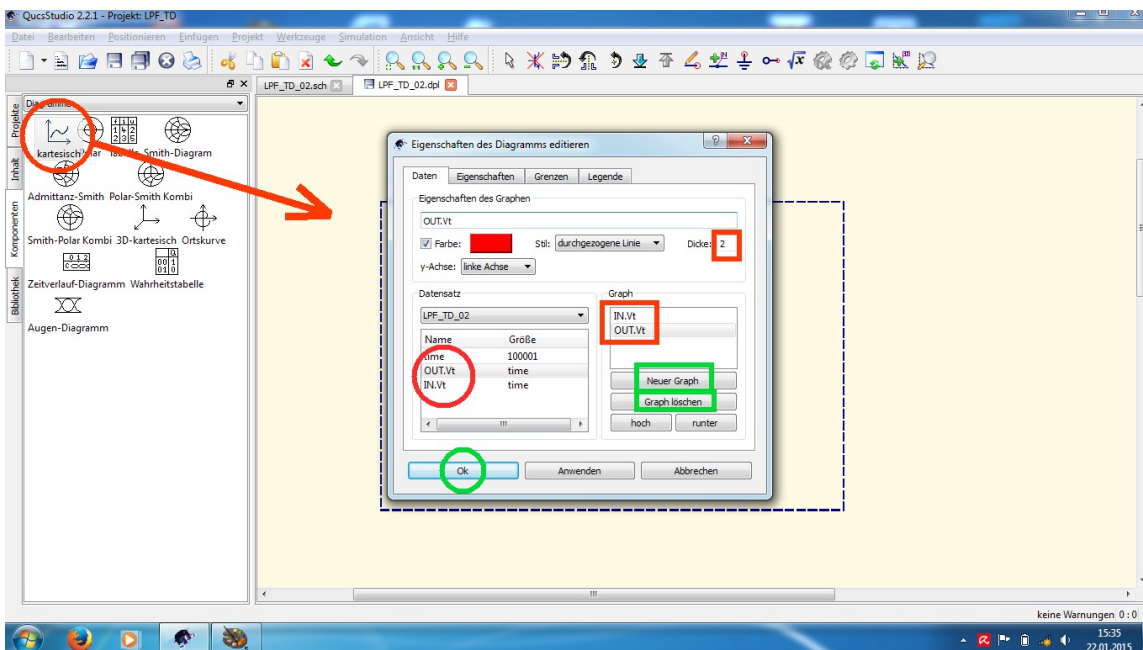
3.2.2.1. Lösung mit Puls-Quelle



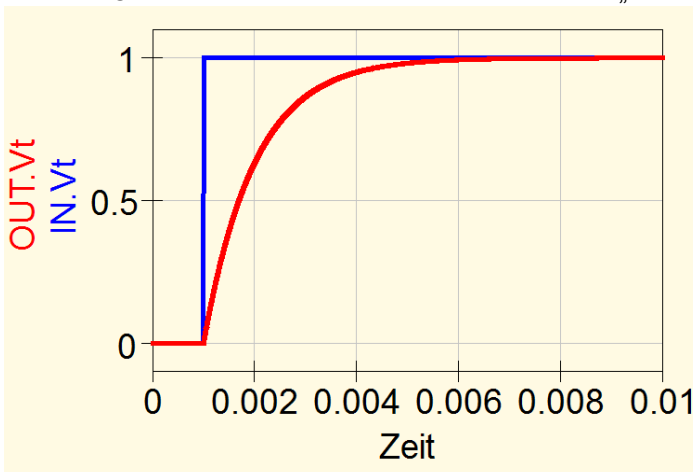
Wir benutzen die vorige Schaltung und ändern nur die **Programmierung der Pulsquelle**. Nach einer Zeitverzögerung von 1 Millisekunde startet der Puls (mit einer Amplitude von 1V) und dauert **5 Sekunden (= also viel länger als die Simulationszeit von 10 Millisekunden)**. Die Anstiegs- und Abfallzeit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

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:

- Der Aufruf dieser Quelle erfolgt wieder über „Komponenten / Quellen“. Aber die PWL-Quelle sucht man vergeblich, denn hier heißt sie „**dateibasierte Spannungsquelle**“.
- 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 $t = 1$ Millisekunde haben wir immer noch Null Volt, aber 1 Mikrosekunde später bereits 1 Volt. Das bleibt so bis zum Ende der Simulationszeit von 10 ms.

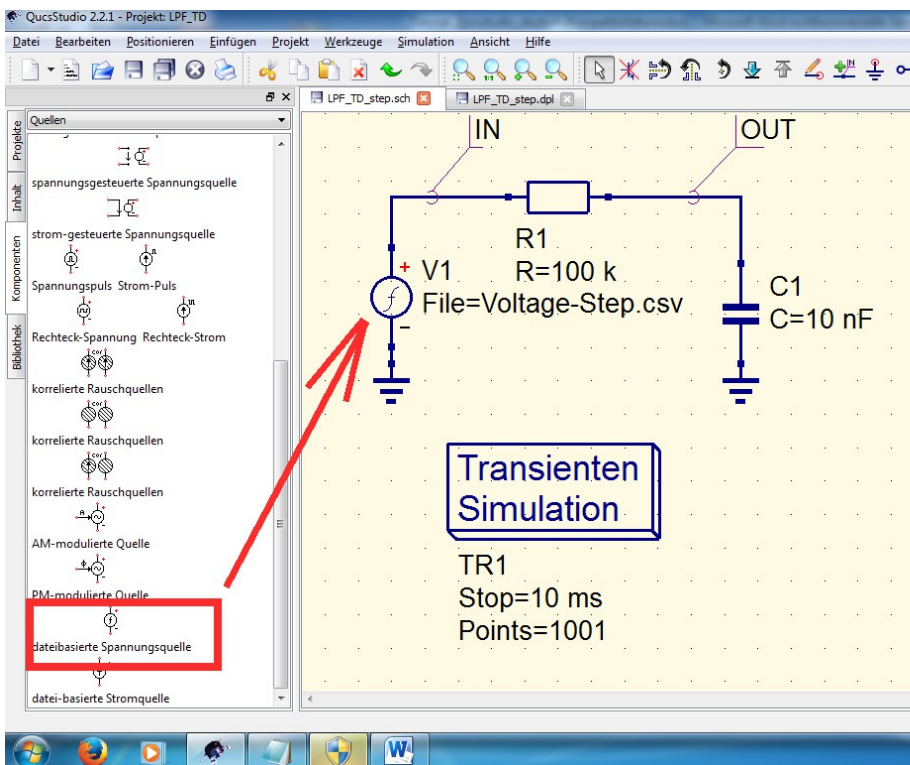
Vorsicht:

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

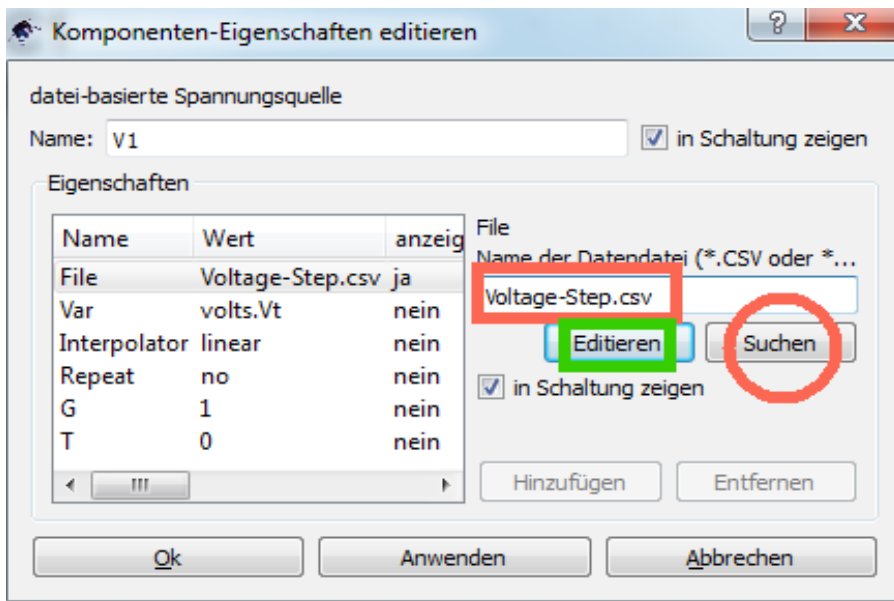
So sieht dann der Datei-Inhalt von „**Voltage-Step.csv**“ aus:

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.



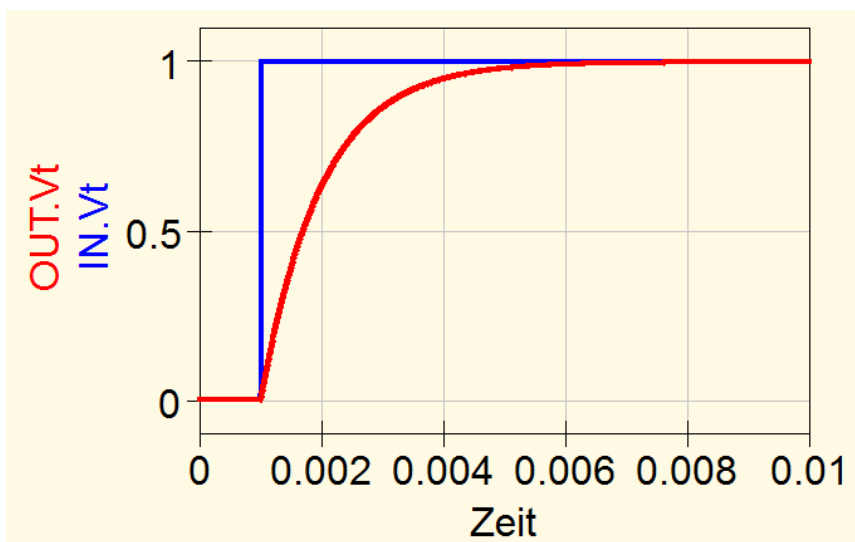
Zuerst stellen wir über „**Suchen**“ den Pfad zur gespeicherten „**Voltage-Step.csv**“-Datei her. Ihre Bezeichnung muss dann im rot markierten Feld auftauchen.

Falls etwas zu ändern ist, 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:

- IN.vt** wird aktiviert und erscheint im Graphen-Fenster.
- Die Liniendicke wird auf „**2**“ gestellt, die Farbe ist automatisch „blau“.
- Dann wird „New Graph“ angeklickt und der nochmals in der zweite Zeile auftauchende Graph „IN.vt“ gelöscht.
- Dafür wird nun „**OUT.vt**“ angewählt und dadurch übernommen.
- Die Liniendicke für „OUT.vt“ sollte ebenfalls noch auf „2“ stehen, aber die Linie sollte eine **rote Farbe** aufweisen.
- Auf der Karteikarte „Eigenschaften“ wird für die X-Achse die Bezeichnung „Zeit“ angegeben.
- Und mit „Anwenden“ und OK sollte sich folgendes Diagramm einstellen.



3.2.3. Periodische Signale am Eingang

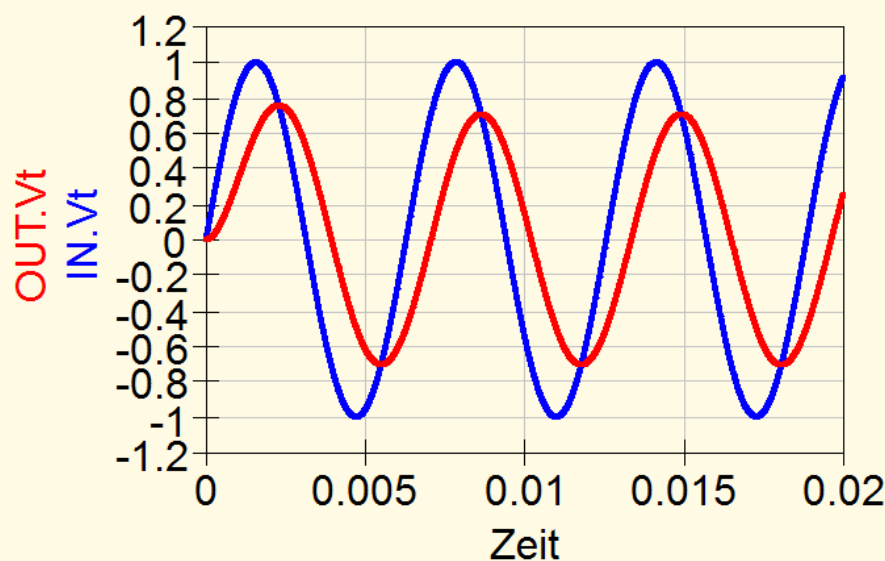
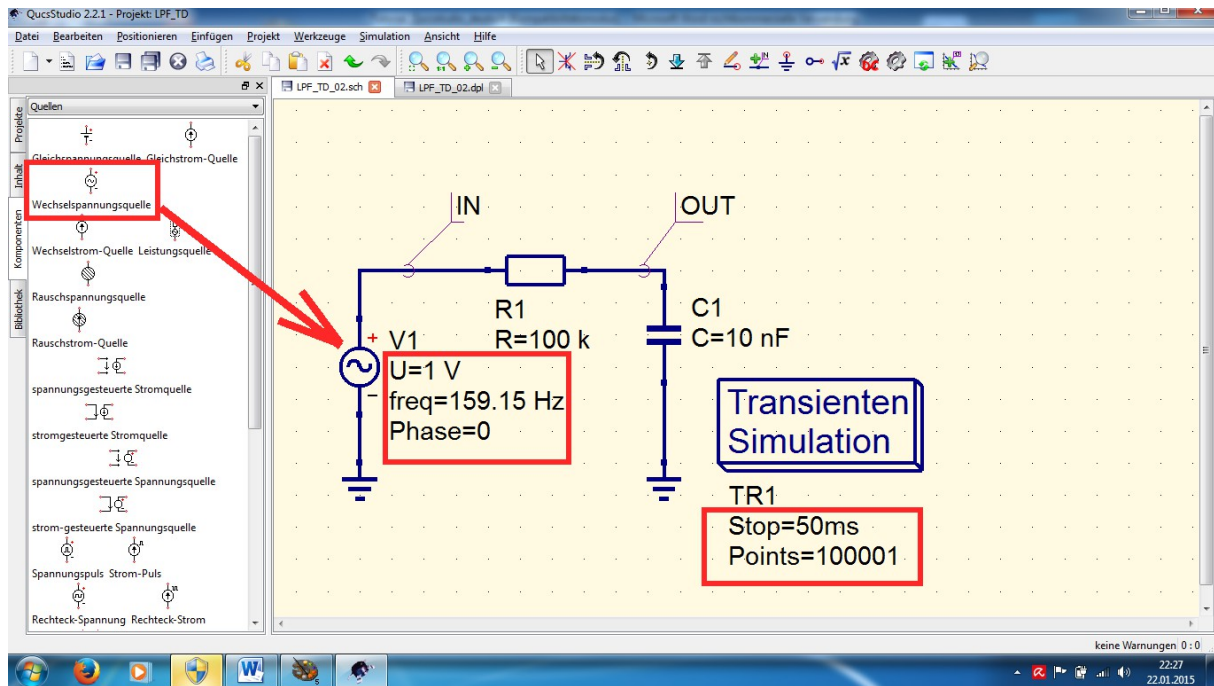
3.2.3.1. Speisung mit einer Sinusspannung

Wir nehmen exakt den Fall der Grenzfrequenz. Sie beträgt

$$f_g = 1 / 2\pi RC = 1 / 2 * \pi * 100k * 10nF = 159,155 \text{ Hz}$$

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...50 ms 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.

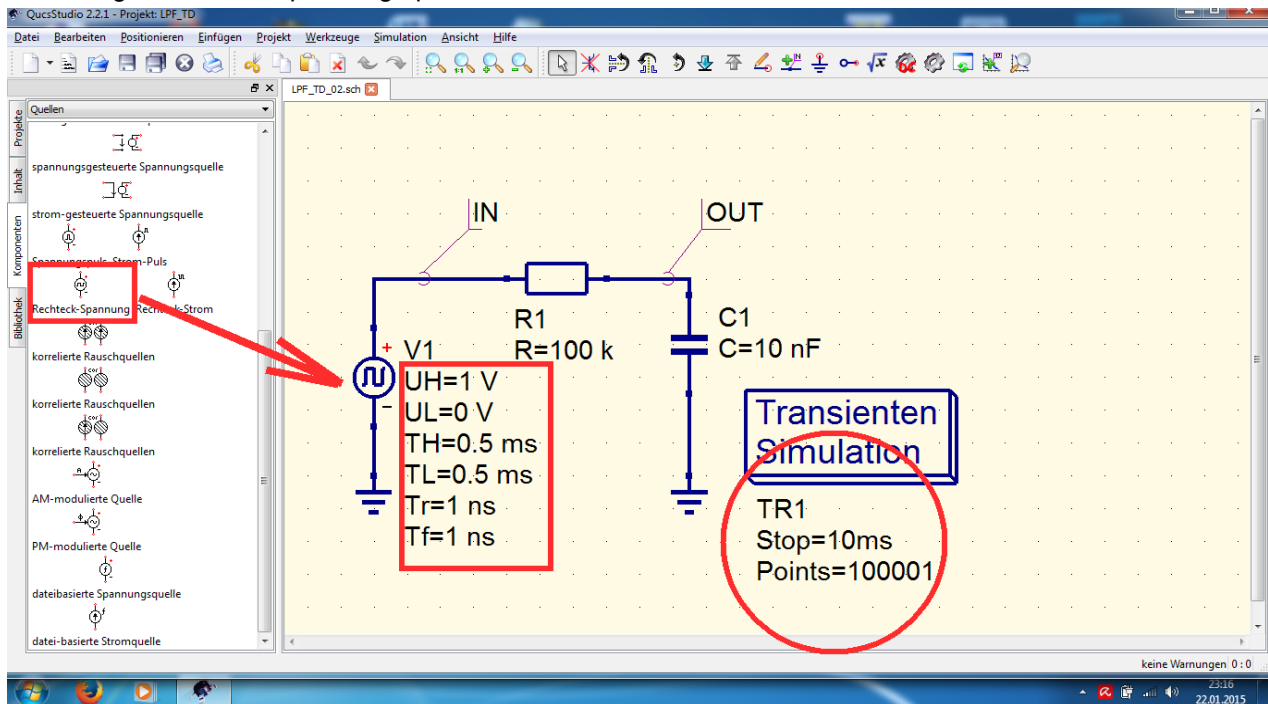
Allerdings erst, wenn man im Menü „Eigenschaften“ die x-Achsen-Kalibrierung auf „manuell“ umgeschaltet und eine Zeit von 0...20ms sowie eine Teilung („Schritt“) von 0,005s eingestellt hat.

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:

- Durch einen doppelten linken oder mit einem rechten Mausklick auf das Symbol der Quelle ihre Property Liste öffnen.
- Dann die gewünschte Größe durch Anklicken in der linken Menüliste markieren.
- Im rechten Fenster den neuen gewünschten Wert eintragen.
- „Anwenden“ drücken.
- Das Häkchen bei „in Schaltung zeigen“ setzen.
- Am Ende, wenn alles erledigt ist, „OK“ drücken.

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

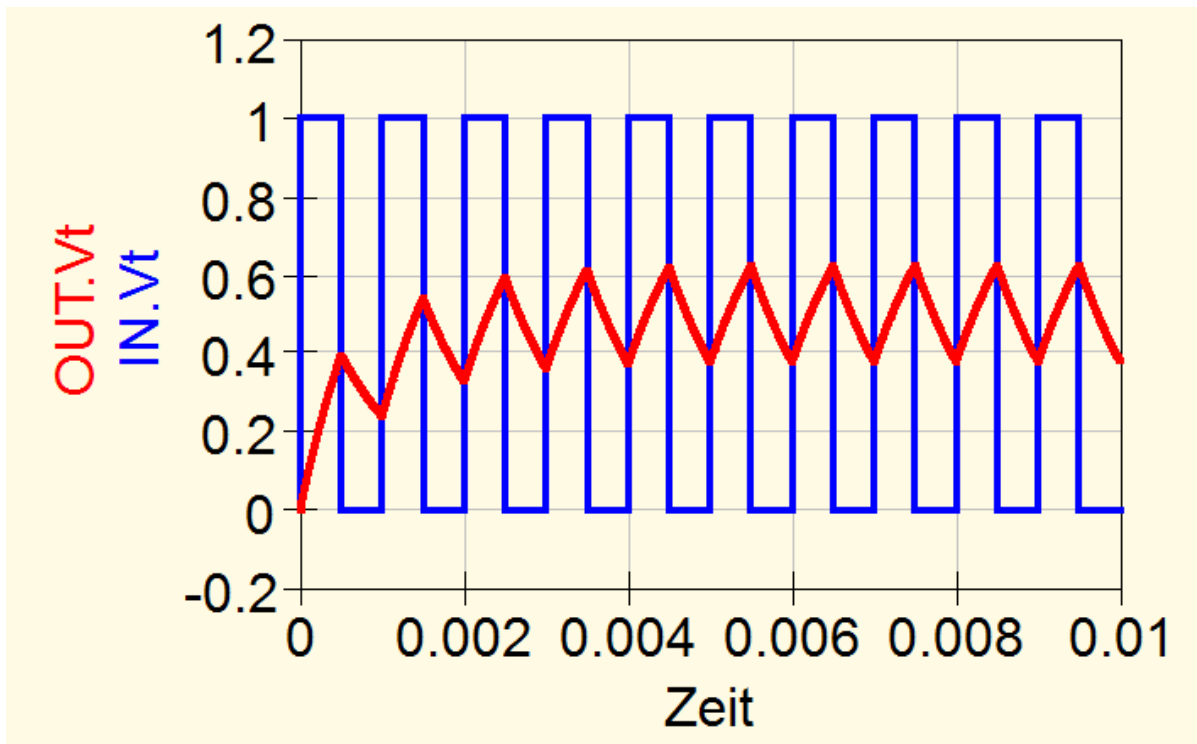
Wir sorgen erneut für eine Darstellung der beiden Spannungen in **zwei unterschiedlichen Farben mit einer Linienbreite von „2“**.

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:

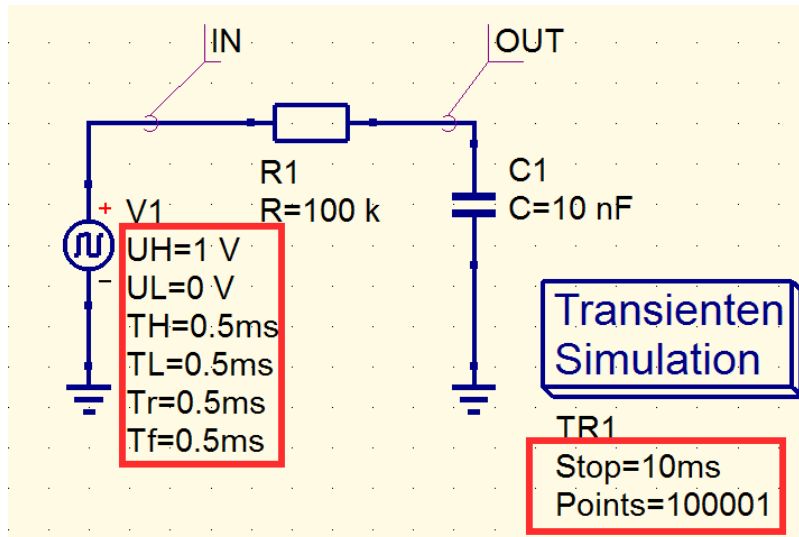


3.2.3.3. Speisung mit einem Dreieck-Signal

Das ist ganz einfach, denn wir nehmen die eben benutzte Schaltung mit dem Rechtecksignal ($f = 1 \text{ kHz}$) 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.

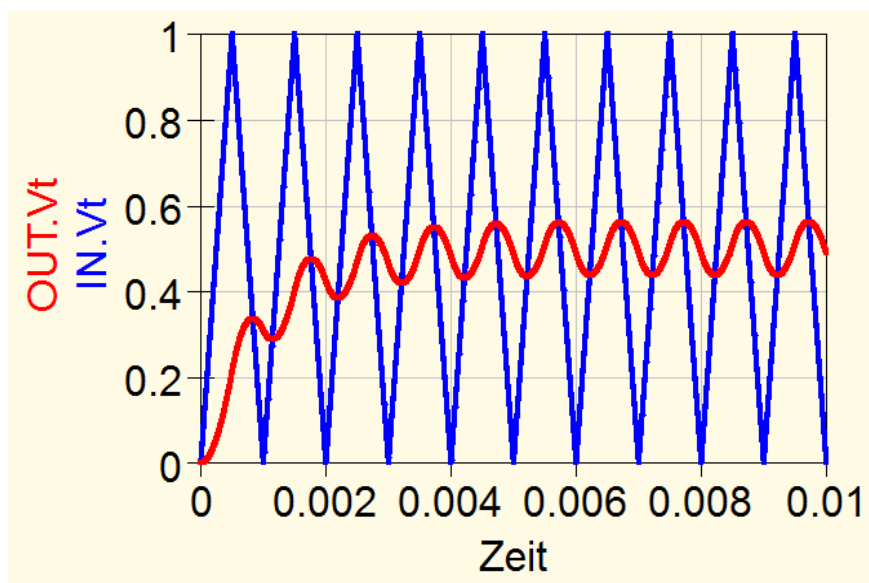
Wenn man nun noch weiß, **dass die Anstiegszeit immer als Teil der „High-Pegel-Zeit“ und die Abfallzeit immer als Teil der „Low-Pegel-Zeit“ gezählt wird**, ist die Sache noch einfacher:



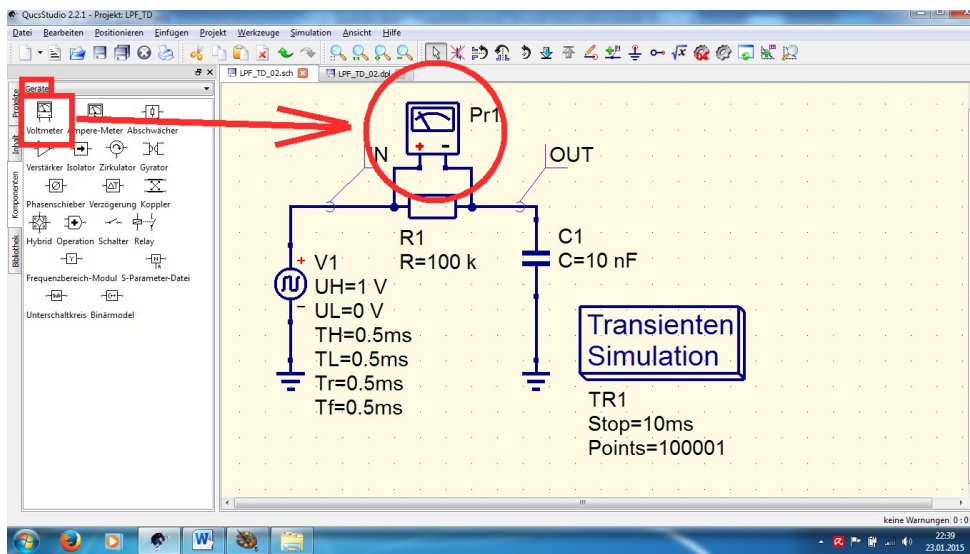
da braucht man nur alle 4 Zeitangaben gleich groß und 0,5 Millisekunden lang zu machen, um ein 1 kHz-Dreieck zu erhalten...

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.



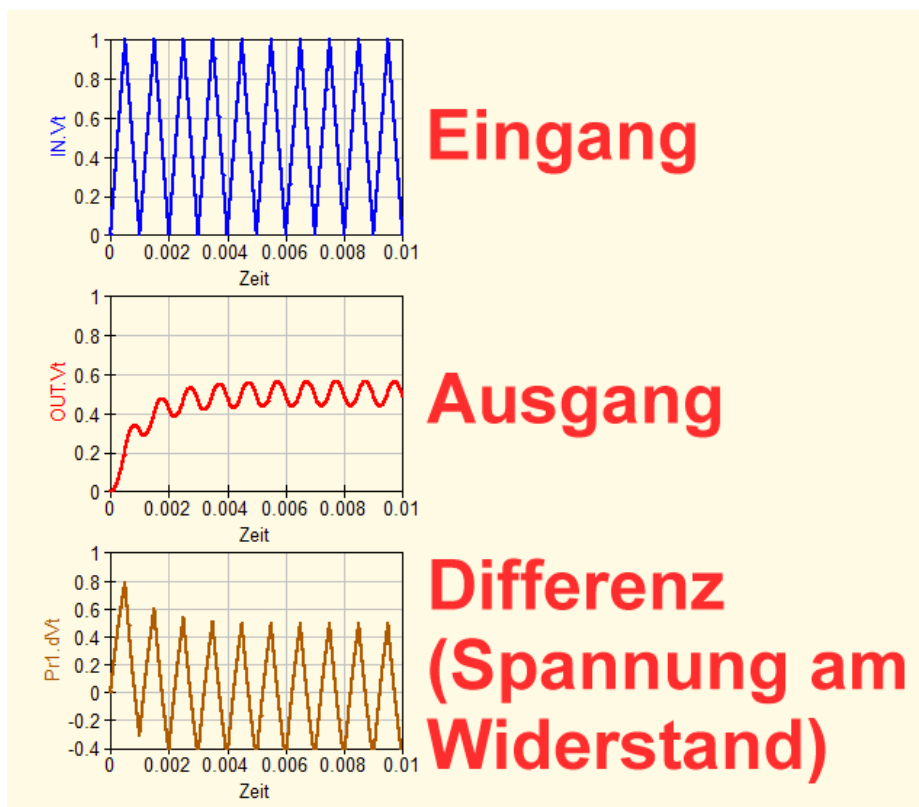
3.2.3.4. Differenz-Messungen



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

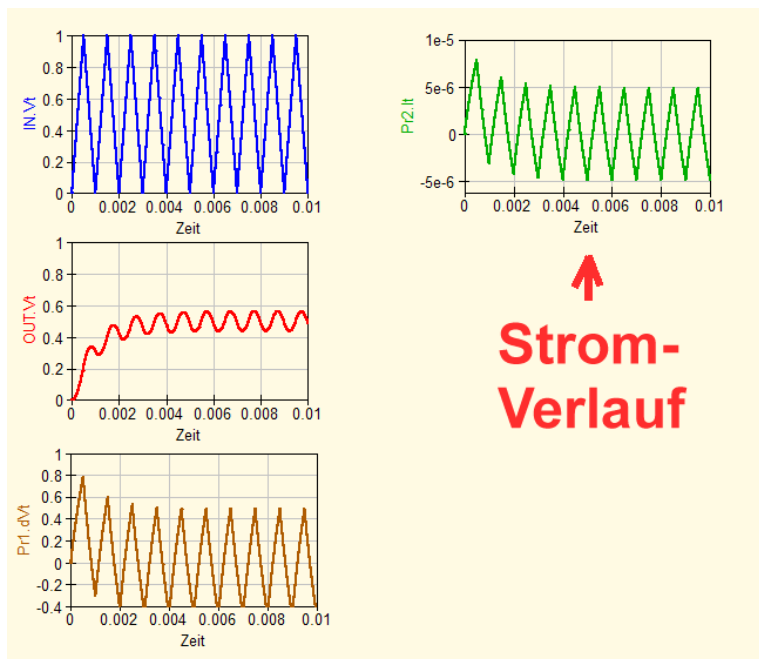
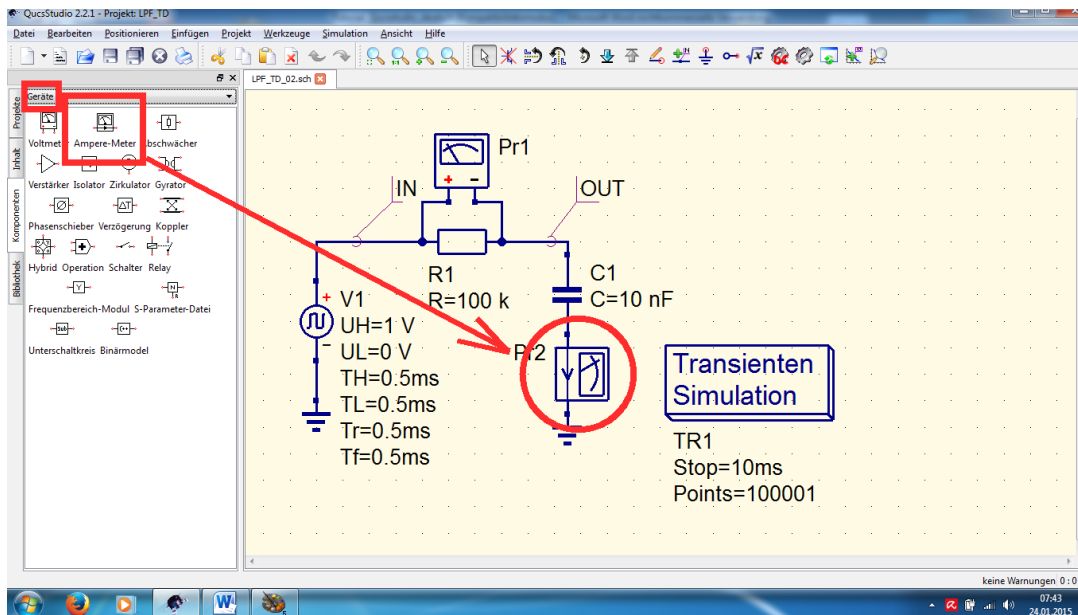
<CTRL +v>), so kann man die drei Spannungen einzeln (mit gleichen Diagramm-Eigenschaften) untereinander anordnen.



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.



„Grenzen“ und entfernt bei der linken Achse das Häkchen für „manuell“.

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

Am schnellsten geht die Ergebnisausgabe, wenn wir eines der linken Diagramm mehrfach kopieren (...am besten die Anzeige des Voltmeters).

Anschließend sind folgende Maßnahmen nötig:

- Die Eigenschaften-Liste wird durch einen rechten Mausklick auf dieses neue Diagramm geöffnet.
- 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.
- Dann wechselt man auf die Karte

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 = $1 / \text{Simulationszeit}$

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: Leakage).

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

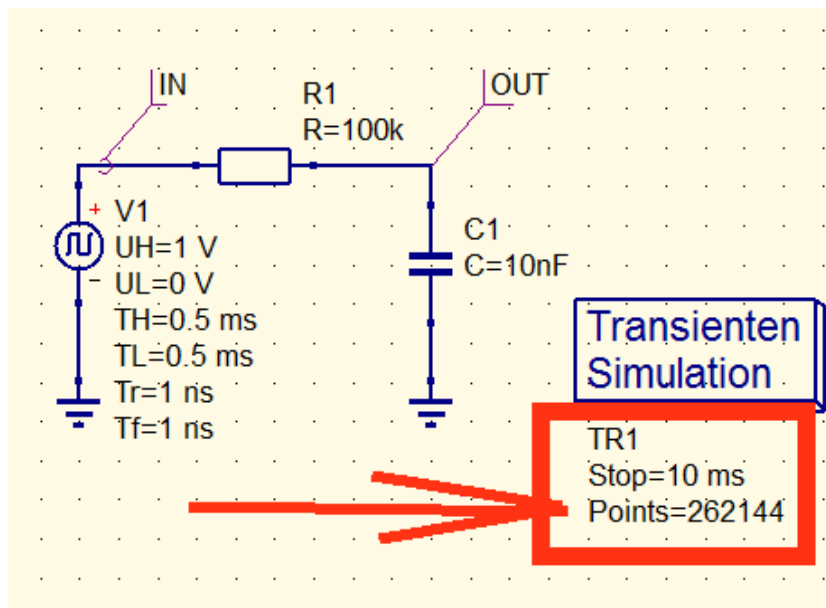
- a) Er soll so klein sein, dass die simulierte Kurve nicht „eckig“ aussieht.
- b) Die zugehörige „**minimale Samplefrequenz = $1 / \text{Timestep}$** “ muss **mindestens doppelt so hoch** sein wie die allerhöchste im simulierten Signal vorkommende Oberwellen-Frequenz. Erst dann ist die „**Shannon-Bedingung**“ erfüllt und man braucht nicht mit „Aliasing-Effekten“ zu rechnen (...die das Ergebnis meist 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**. Und immer muss man im Hinterkopf behalten:

Mehr Samples für die FFT geben bessere Amplitudenauflösung und höhere Grenzfrequenz, erhöhen aber entsprechend die Rechenzeit und die Größe der Ergebnisdatei!

3.3.2. Ein praktisches Beispiel

Dazu benutzen wir unseren nun schon gut bekannten Tiefpass, an den eine Pulsspannung mit $f = 1000 \text{ Hz}$ und $U_H = 1 \text{ V}$ bzw. $U_L = 0 \text{ V}$ 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

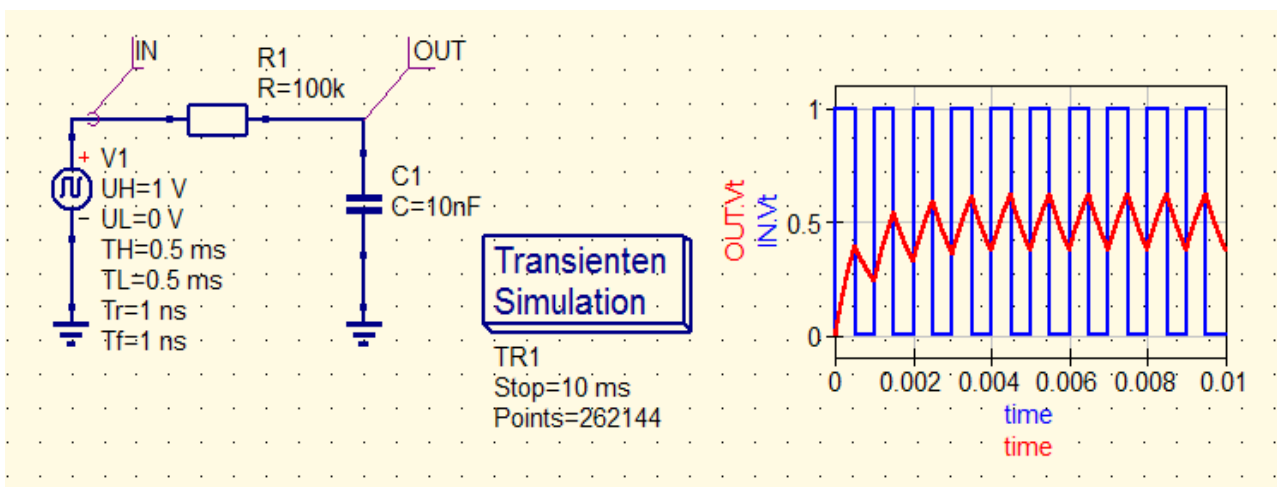
Außerdem kennen wir nun schon gleich die Frequenzauflösung des Spektrums. Sie beträgt

$1 / 10 \text{ ms} = 100 \text{ Hz}$

Nach der Simulation wählen wir wieder in bekannter Weise unser

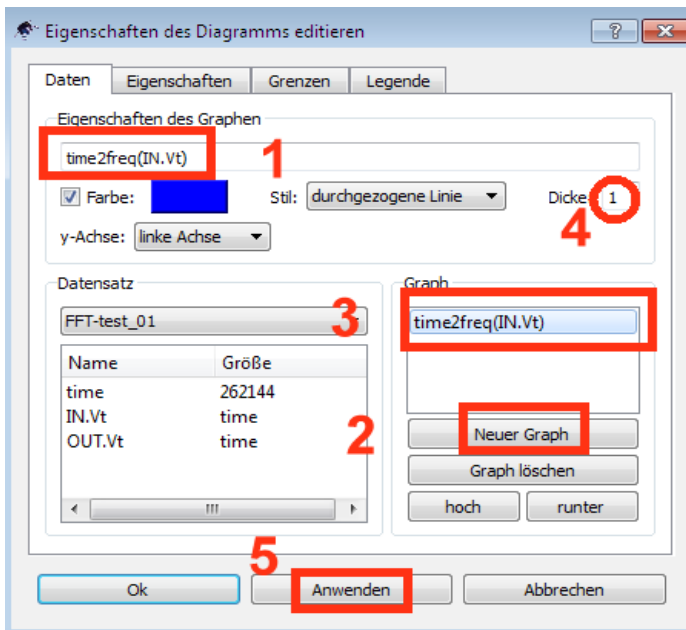
kartesisches Diagramm und lassen uns beide Spannungen für die Zeit von $t = 0 \dots 10 \text{ ms}$ darstellen. Aber jetzt machen wir uns das Leben leichter:

Wir kopieren das Ergebnis in die Zwischenablage (<CTRL + 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 Zahnrad klicken und simulieren, schaltet qucsstudio zwar automatisch auf die Ergebnisseite. Aber die Schaltplan-Seite 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!**



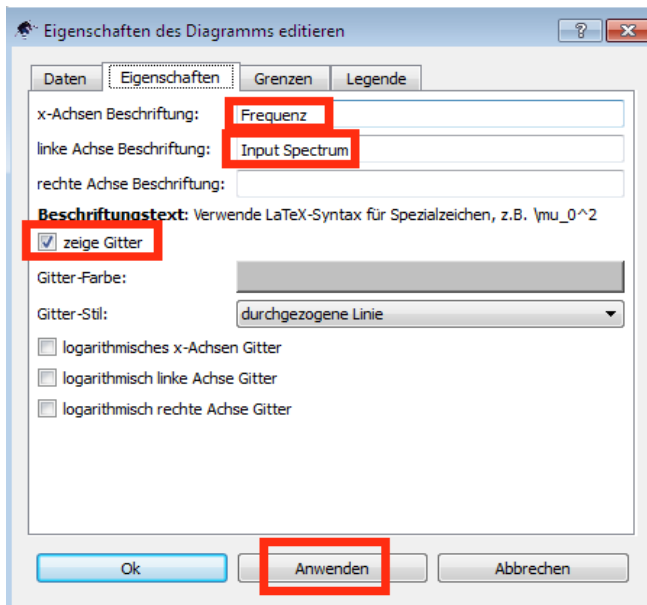
1:
In die Zeile „**Eigenschaften des Graphen**“
geben wir die FFT-Aufforderung ein:

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**“.



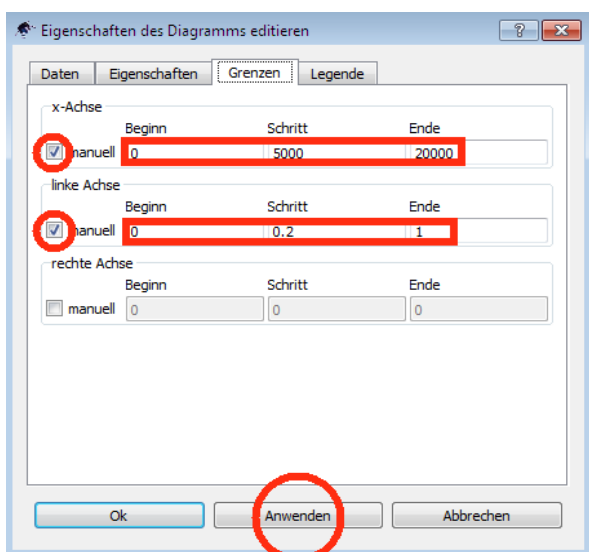
Dann wechseln wir zur Karteikarte „Eigenschaften“.

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.

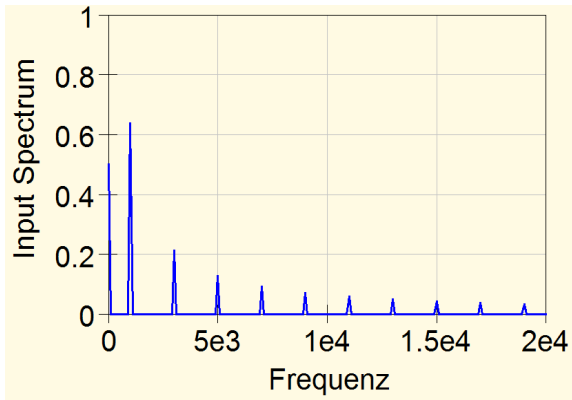


Auf der Karte „**Grenzen**“ geht es weiter.

Beide Achsen werden auf „manuell“ umgestellt und
folgende Bereiche vorgesehen:

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

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:

☒ zeige Gitter

Gitter-Farbe:

Gitter-Stil: durchgez.

☐ logarithmisches x-Achsen Gitter

☒ logarithmisch linke Achse Gitter

☐ logarithmisch rechte Achse Gitter

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

Daten Eigenschaften Grenzen Legende

x-Achse

	Beginn	Schritt	Ende
☒ manuell	0	5000	20000

linke Achse

	Beginn	Schritt	Ende
☒ manuell	1e-05	0.2	1

rechte Achse

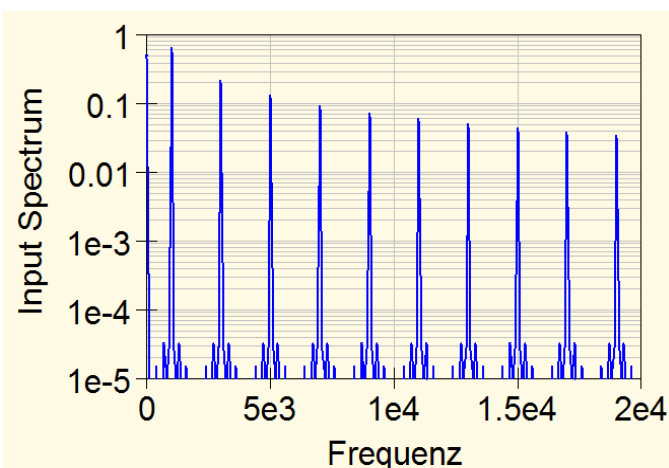
	Beginn	Schritt	Ende
☐ manuell			

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.



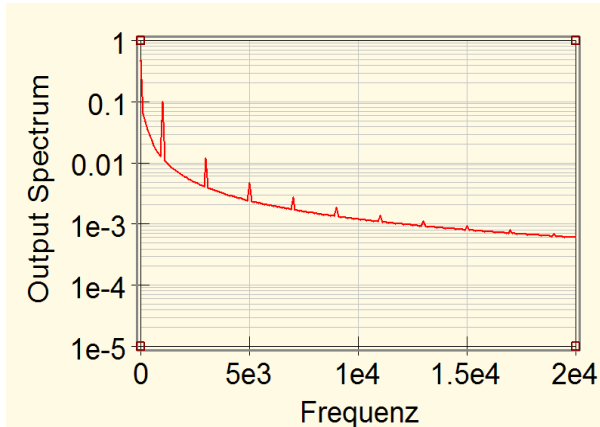
Na also....

Ürigens: 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 Linienbreite einstellen (...Breite „2“ ist optimal). Mit „**Anwenden**“ wird das übernommen und mit OK schließt sich das Menü wieder.

Die nun folgende Aufgabe

„Stellen Sie 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 braucht im neuen Diagramm bei den Graph-Eigenschaften (graph properties) nun die Anweisung

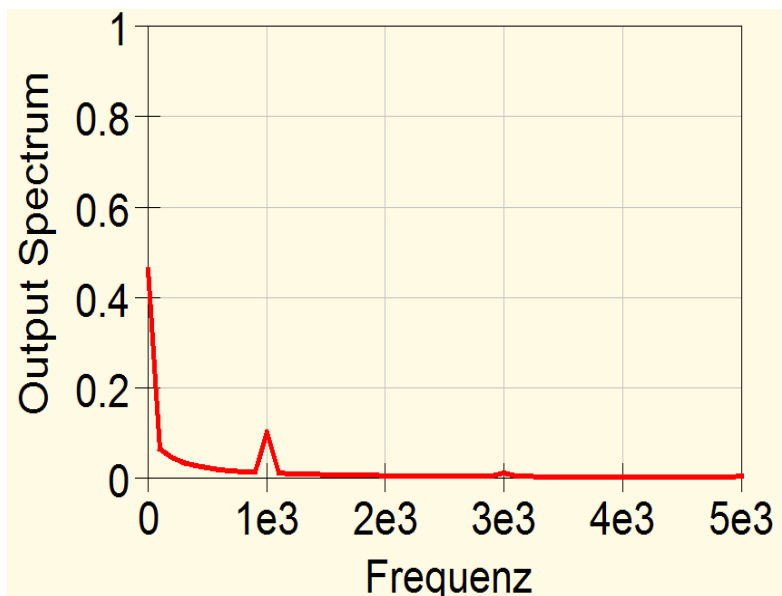
time2freq(OUT.Vt)

Dann darf man nicht vergessen, unter „**Eigenschaften**“ die neue Achsenbezeichnung „**Output Spectrum**“ sowie bei der waagrechten Achse „**Frequenz**“ vorzusehen.

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

Zusatzaufgabe:

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



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

Ihre Breite wird durch die Simulationszeit festgelegt und bei $t_{sim} = 10 \text{ ms}$ bekommen wir eben nur eine Auflösung von $b = 1 / t_{sim} = 100 \text{ Hz}$.

Wer also 10 Hz Auflösung möchte, muss

$$t = 1 / 10 \text{ Hz} = 100 \text{ ms}$$

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:

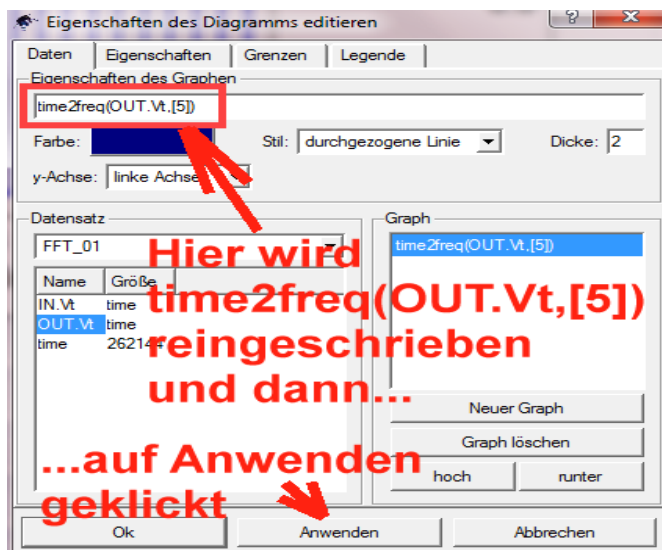
Wir finden den Raum zwischen den Spektrallinien gegenüber dem Eingangsspektrum sehr stark „mit Energie aufgefüllt“ und dieser Effekt klingt erst zu höheren Frequenzen hin ab.

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)

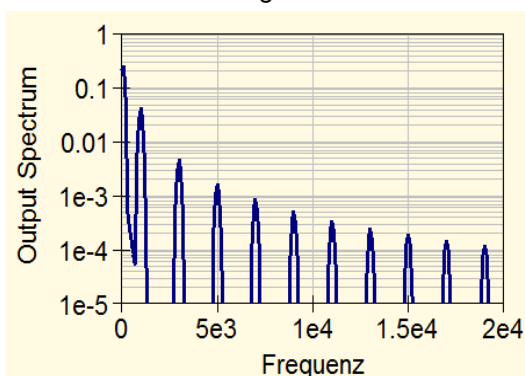
(Ein Tipp: Bitte das Komma **vor die eckige Klammer setzen und nicht vergessen**. Sonst läuft nichts. Und ein weiterer Tipp des Programmautors nach dem Probelesen: **die eckige Klammer KANN weggelassen werden, aber das Komma MUSS BLEIBEN**)



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:

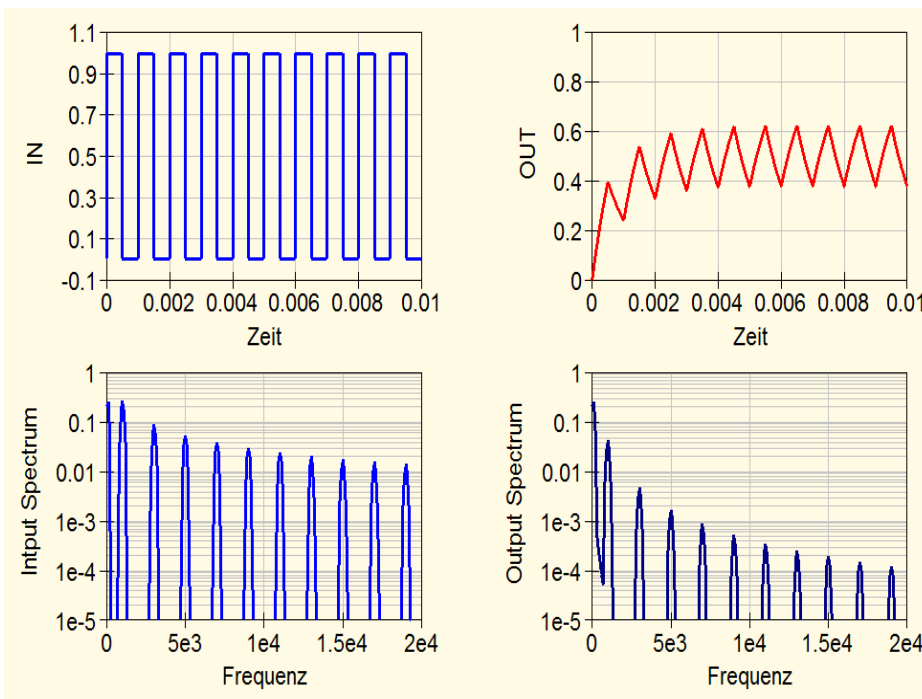


Man erkennt, dass der Einfluss des Startvorgangs nun fast verschwunden ist, aber:

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 = $1 / \text{Simulationszeit}$).....

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:

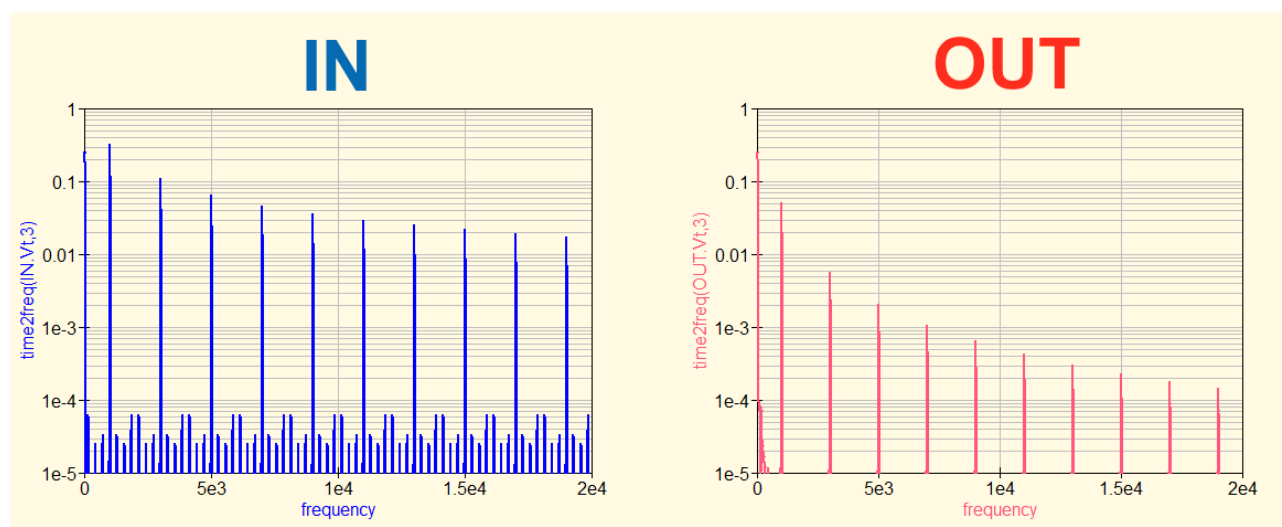
Es werden nicht nur die Spektrallinien breiter – **durch die wegen der Window-Funktion „weggeworfenen Samples“ wird auch der korrekte Absolutwert der Spannungsamplituden vermindert** (...aber die relativen Unterschiede zwischen den Linien bleiben erhalten). **Zur Ermittlung absoluter korrekter Amplitudenwerte muss also die Window-Funktion ausgeschaltet werden!**

Aufgabe:

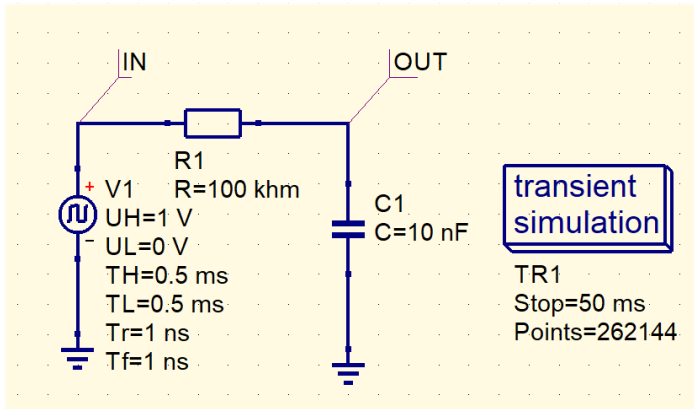
Durch die Window-Funktion hat sich ja die Anzahl der zur Verfügung stehenden Samples vermindert. Das entspricht einer kürzeren Simulationszeit und ergibt deshalb eine schlechtere Frequenz-Auflösung.

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 / ohne Window“.

Ergebnis:



Zusammenfassende Aufgabe:



Verwenden Sie diese 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).

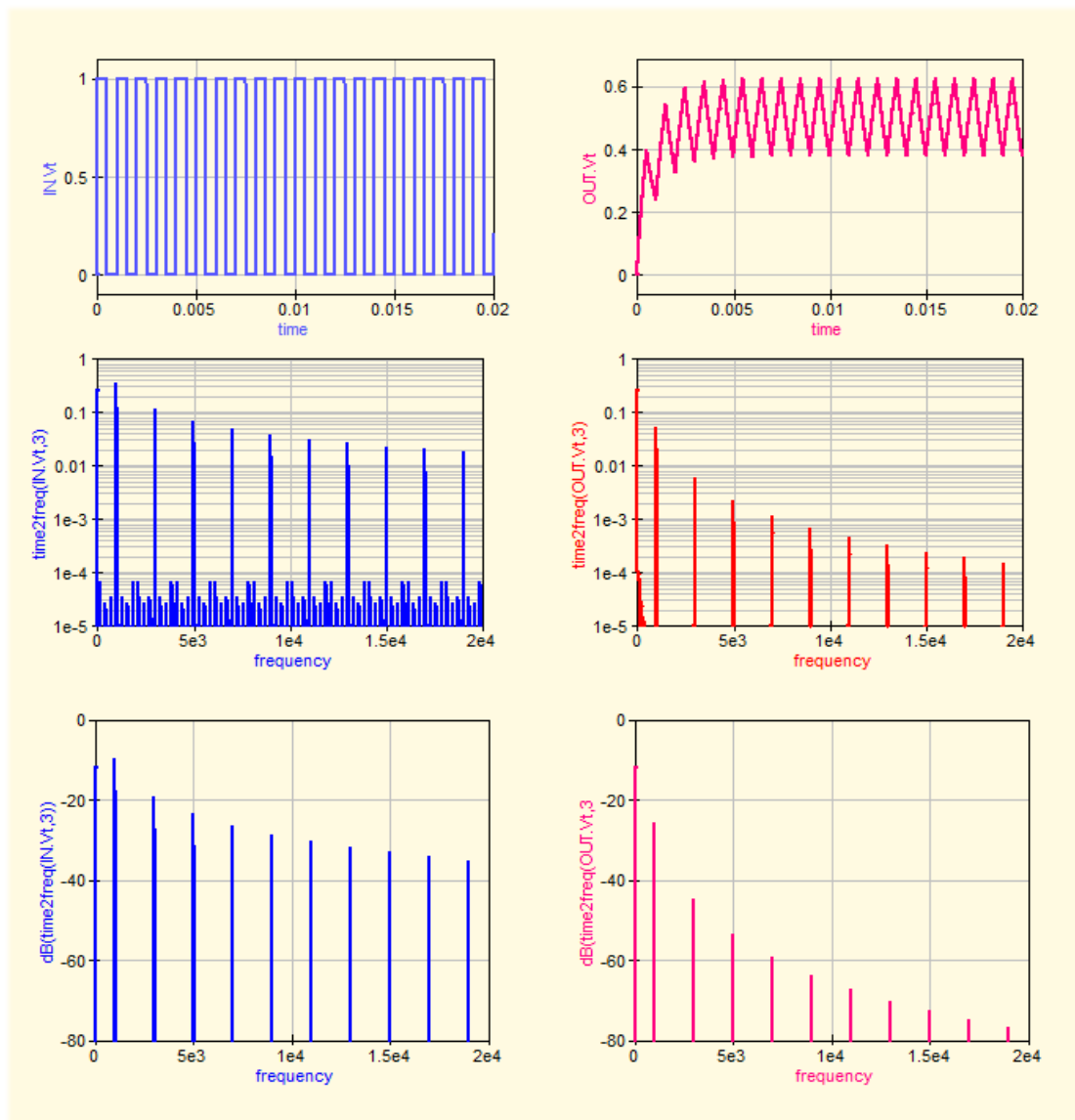
Simulieren Sie zusätzlich diese Spektren mit „dB“-Anzeige bis $f = 20$ kHz (Achtung: senkrechte Achse auf „linear“ einstellen).

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 Spannungsunterschiede 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 $Z = 50 \Omega$ ergibt das einen Effektivwert der Spannung von

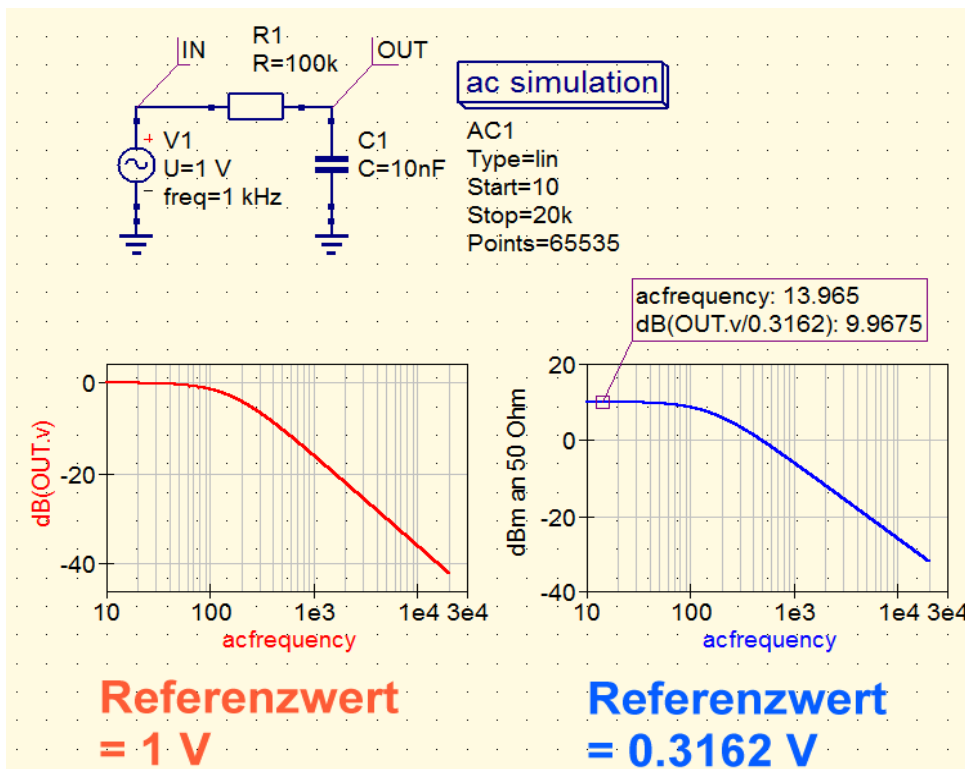
$$U = \sqrt{\text{Leistung} \cdot \text{Widerstand}} = \sqrt{1 \text{exp}(-3) \cdot 50 \Omega} = 0,2236 \text{ V}$$

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

$$0,2236 \text{ V} \cdot \sqrt{2} = 0,3162 \text{ V}$$

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:

$\text{Spannungspegel in dBm} = 20 * \log(U / U_{\text{ref}}) = 20 * \log(\text{OUT.v} / 0,3162)$

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“).**