Labor für Nachrichtentechnik an der Dualen Hochschule Baden Württemberg in Friedrichshafen
==

Harmonic Balance Simulation mit qucsstudio

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

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

27. November 2016

1. Harmonic Balance: wozu?

Simulationen im Zeitbereich (= Time Domain) liefern nicht nur Informationen über den zeitlichen Verlauf aller beteiligten Spannungen und Ströme, sondern zeigen über eine nachfolgende FFT (= Fast Fourier Transformation) auch die durch Nichtlinearitäten entstehenden neuen Frequenzen. **Aber immer nur für die vorgegebenen und konstanten Frequenzen der gerade verwendeten Eingangssignale.**

Ein „**AC-Sweep**“ **ignoriert dagegen alle Nichtlinearitäten und liefert den „Frequenzgang der Übertragungsfunktion**“ für das sinusförmige und „gesweepte“ Ansteuersignal in der Frequency Domain. **Aber immer nur für eine konstante Amplitude der Eingangsspannung und ohne Berücksichtigung der erzeugten Oberwellen.**

Die dadurch noch vorhandene Lücke (= die zusätzliche Simulation der Nichtlinearitäten bei sich ändernder Speisefrequenz und verschiedenen Amplituden des Eingangssignals) schließt „Harmonic Balance“.

Wichtig:
Dieses Programm simuliert NUR in der Frequency Domain“!

2. Wie funktioniert das?

Das zeigt das folgende Bild aus dem qucs-Manual:

As the non-linear elements are still modeled in time domain, the circuit first must be separated into a linear and a non-linear part. The internal impedances Z_i of the voltage sources are put into the linear part as well. Figure 7.1 illustrates the concept.

Let us define the following symbols:

M = number of (independent) voltage sources
N = number of connections between linear and non-linear subcircuit
K = number of calculated harmonics
L = number of nodes in linear subcircuit

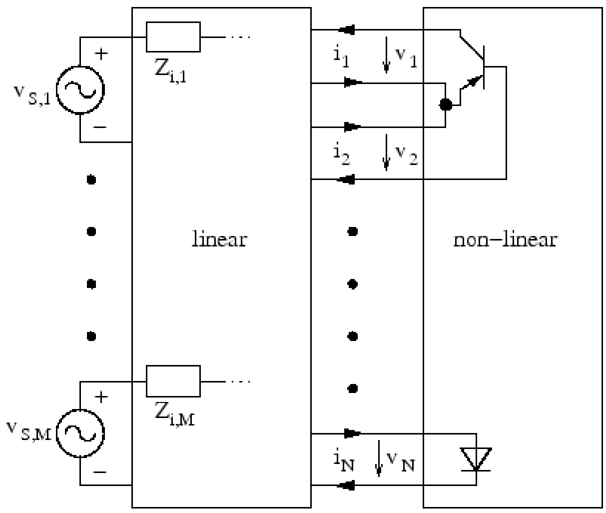


Figure 7.1: circuit partitioning in harmonic balance

Es ist eine raffinierte und deshalb auch patentierte Idee, denn wenn in einer Schaltung lineare Bauteile (z. B. Widerstände, Spulen, Kondensatoren...) und nichtlineare Bauteile (z. B. Dioden, Transistoren, FETs...) zusammen eingesetzt werden, geht das Programm folgendermaßen vor:
a) **Lineare Bauteile** faßt es samt ihren Verbindungsleitungen und Schaltungsknoten komplett in einem „**linearen Block**“ (= linear subcircuit) zusammen.

b) Alle **nichtlinearen Bauteile** samt ihren Verbindungsleitungen und Schaltungsknoten kommen in eine zweite Kiste mit dem Namen „**nichtlinearer Block**“ (nonlinear subcircuit).

c) Dann werden beide Blöcke miteinander über deutlich erkennbare (und exakt benannte) zusätzliche Leitungen (= connections) miteinander verbunden – **damit ist die ursprüngliche Startschaltung wieder korrekt hergestellt**. Nicht vergessen darf man natürlich, noch die gewünschten Eingangssignale anzulegen.

Wer nun die Frage „Und jetzt?“ stellt, bekommt eine verblüffende Antwort:

Alles im **linearen Block** kann direkt (über die Bauteilmodelle) in der **Frequency Domain** berechnet werden.
Alle Berechnungen und Simulationen im **nichtlinearen Block** werden dagegen (nach der Übergabe der Einzelsignale aus dem linearen Block über die „connections“) nach einer Fourier-Transformation im **Zeitbereich** durchgeführt. Dadurch erhält man die „durch Nichtlinearitäten verzerrten Stromverläufe“ – und nach der Rücktransformation jedes Stromes können die Ergebnisse wieder in Form der Grundwelle und aller entstandenen Oberwellen an den linearen Block **im Frequenzbereich zurückgegeben** werden.

Man ahnt, dass der erforderliche Rechenaufwand enorm ist. Es muss nämlich erst mal vom Anwender vor Beginn angegeben, welcher höchste Oberwellengrad berücksichtigt werden soll. Dann werden in Form von „mehrdimensionalen Matrizen“ die „Spektren für alle Verbindungen zwischen linearem und nichtlinearem Block“ berechnet. Das ginge ja noch, aber:

Bei jeder Verbindungsleitung (connection) muß folgende Bedingung erfüllt werden:

Was da reingeht, muß auch wieder herauskommen!

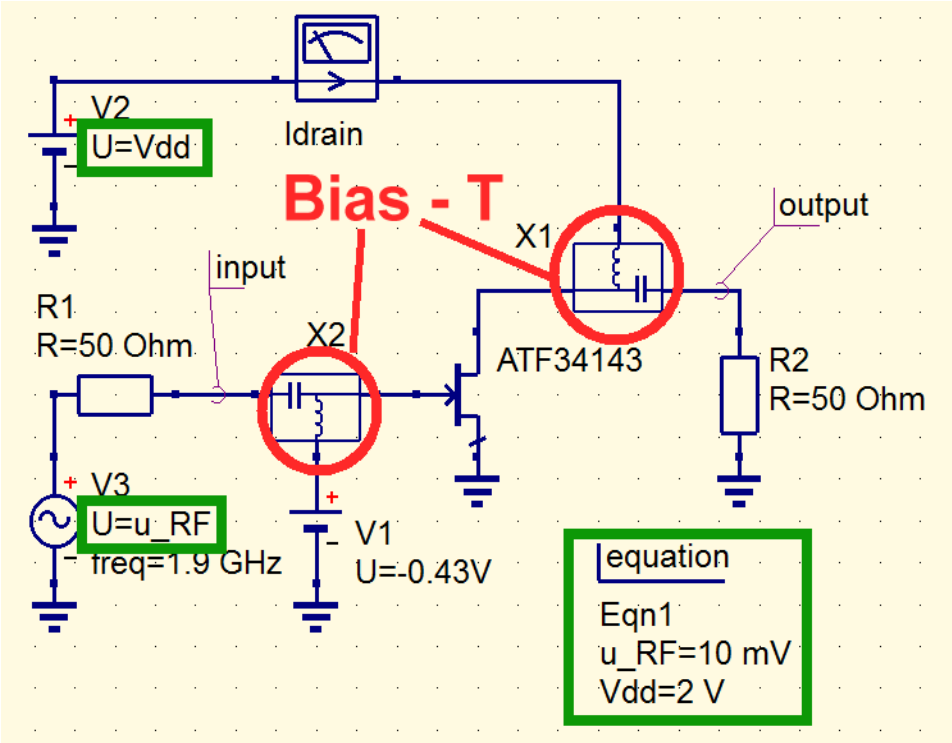
Das bedeutet im Klartext:
Was der lineare Teil (z. B. als Leistung) in Form eines Stroms über eine Verbindungsleitung zum nichtlinearen Teil schickt, muss genau so groß sein wie das vom nichtlinearen Teil nach der Fourier Transformation in dieser Leitung zurückgegebene Spektrum (= Summe der Leistungen von Grundwelle und zugelassenen Oberwellen).

Das funktioniert leider nicht beim ersten Anlauf, denn das Programm startet mit (ungefähren) Vorgabewerten aus dem linearen Teil, die dem nichtlinearen Teil angeboten werden. Dort wird im Zeitbereich gerechnet, es werden die nötigen Fourier Transformationen durchgeführt und hinterher die Leistungsbilanzen bei allen Verbindungsleitungen erstellt. Geht die Rechnung nicht auf (= die vom linearen Teil hineingeschickte Leistung stimmt noch nicht mit der vom nichtlinearen Teil als Spektrum dargestellten Leistung überein), dann wiederholt das Programm mit geänderten Werten die ganze Prozedur. Diese „Iteration“ wird so lange ausgeführt, bis das **Gleichgewicht (= also die „Harmonic Balance“)** erreicht und der Fehler minimal ist. Und das kann dauern, denn linearer und nichtlinearer Teil beeinflussen sich gegenseitig – schließlich handelt es sich ja um EINE zusammengehörige Schaltung.....man ahnt: die Theorie und der mathematische Hintergrund sind recht kompliziert.
Die sich ergebenden Möglichkeiten einer erfolgreichen Harmonic Balance Simulation sind jedoch faszinierend. Wenn man **zusätzlich mit einem „Parameter Sweep“ arbeitet**, hat man anschließend einen riesigen und kompletten Ergebnis-Datensatz **für die Ansteuerung mit Signalamplituden von „sehr klein“ bis „jetzt reicht es aber..“** zur Verfügung. Damit kann man z. B. den Frequenzgang der Verstärkung bei kleiner und bei großer Eingangsspannung vergleichen, kann den 1 dB Kompressionspunkt ermitteln, kann den IP3-Punkt errechnen lassen, kann sich Oberwellen-Spektren über der Frequenz ansehen.....usw. ….usw.

3. Untersuchung eines einstufiger Verstärkers mit einem FET

3.1. Die verwendete Schaltung

Sie findet sich in der **qucsstudio-Homepage** als **„FET_P1dB.sch“** in einer gepackten Sammlung, die über „**Beispiele**“, gefolgt von „**HB-Analyse**“ als **SimulationHarmonicBalance.qucs** heruntergeladen und dann entpackt werden kann. Das gilt aber nur für den „Hausgebrauch“, denn in der DHBW haben Sie keine Schreibrechte für Ihre Festplatte (...nur für Ihr Studenten-verzeichnis!). Also erstellen wir zur Abwechslung diese Schaltung mal selbst).



Details:

Die Betriebsspannung wird als Variable **„Vdd“** angegeben, damit ein Parameter Sweep (SW1) vorgenommen werden kann.

Als **Ansteuersignal dient die Spannungsquelle V3 mit der Frequenz f = 1,9 GHz**. Die Urspannung der Quelle ist eine Variable **„u_RF“** für einen weiteren Parameter Sweep (SW2).

Der FET erhält über ein „**Bias T**“ eine konstante Gate-Vorspannung von **-0,43V.**

Am Drain-Anschluss findet sich ein weiteres „**Bias T**“ für die Speisung des Drain-Anschlusses mit der Versorgungsspannung UND der Auskopplung des verstärkten Signals an R1.

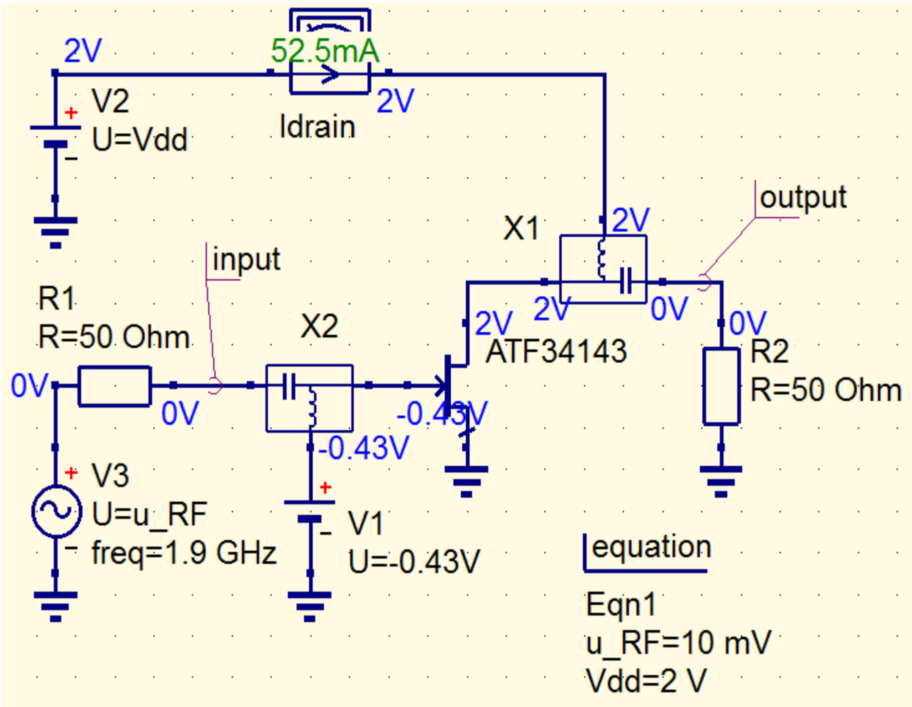
Das „Bias-T“ findet sich unter „Components / Lumped Components“ (= Bauteile / Diskrete Bauteile)

Der ATF34143 steckt im Ordner „Libraries / Special Transistors“.

Das Ampere-Meter zur Strommessung kann aus „Components / Devices“ als „current probe“ geholt werden.

Wichtig:
Wenn wir die beiden Spannungen (u_RF bzw. Vdd) als Variable vorgeben, müssen wir natürlich unter Verwendung einer Gleichung („equation“) ihre Startwerte vorschreiben.

3.2. DC-Simulation = Bestimmung des Arbeitspunktes



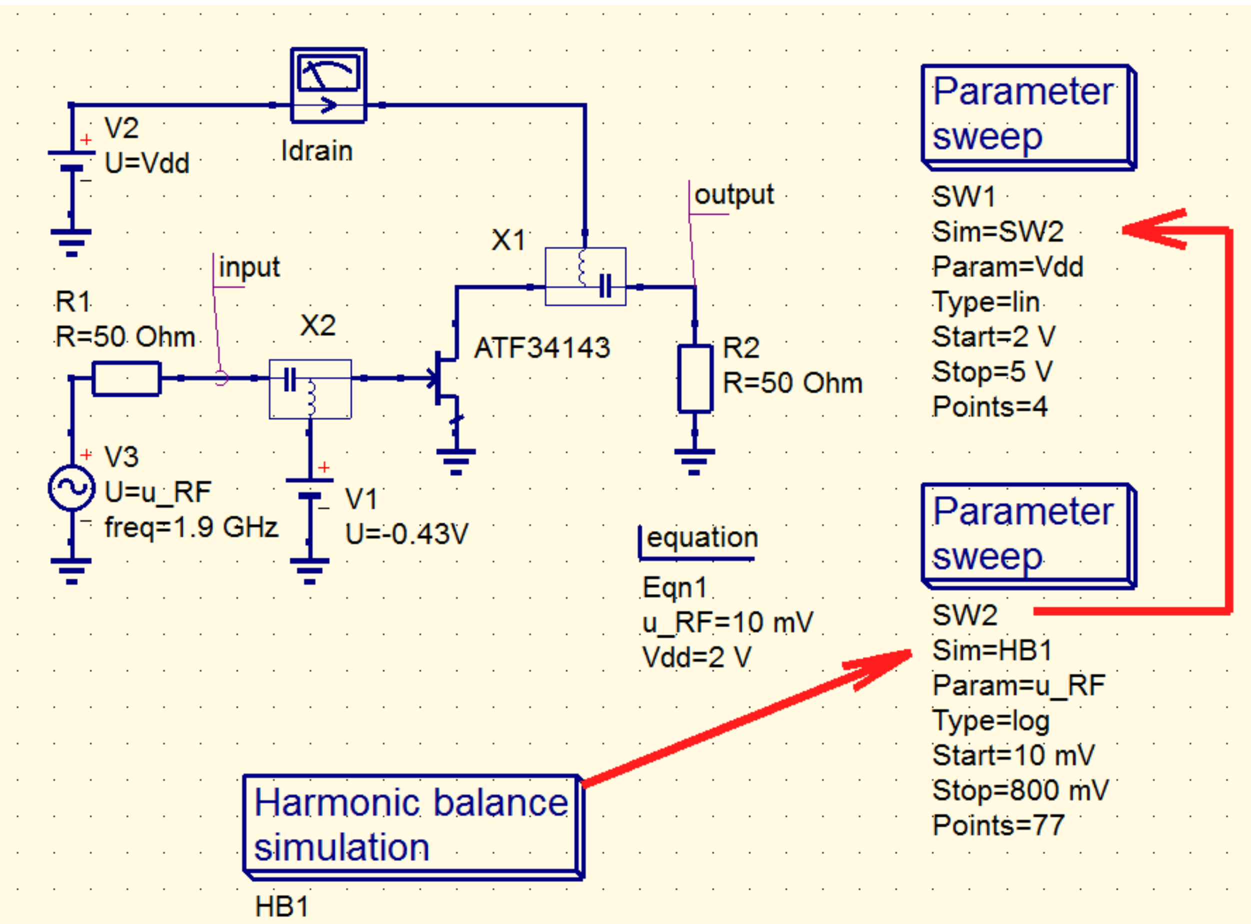
Rechts oben in der Menüleiste findet sich ein Button in Form eines **Zahnrades und der roten Aufschrift „DC“.**

Wenn wir ihn drücken, erhalten wir das nebenstehende Bild mit den Spannnungen an allen Schaltungsknoten sowie dem fließenden Drain-Ruhestrom.

Damit lautet die Arbeitspunkt-Beschreibung:

„Es fließt ein Ruhestrom von 52,5mA bei einer Drainspannung von +2V und einer Gate-Vorspannung von -0,43V“.

3.3. Programmierung der Harmonic Balance - Parametersimulation



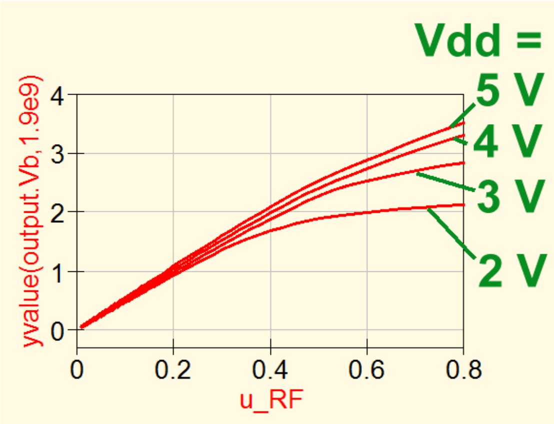
Das beginnt mit dem Aufruf der **HB-Simulation** (...findet sich unter „Components / Simulations“).

Dann brauchen wir einen **Parameter-Sweep SW2**, bei dem die Eingangsspannung **u_RF von 10mV bis 800mV variiert wird**.

Und dieses Parameter-Sweep SW2 wiederholen wir nacheinander **für Betriebsspannungswerte mit 2, 3, 4 und 5 Volt mit einem weiteren Parameter-Sweep SW1**.

Nach der Simulation haben wir nun unzählige und unglaubliche Möglichkeiten bei der Auswertung.

3.4. Die Ausgangsspannung als Funktion der Eingangsspannung



In einem kartesischen Diagramm geben wir als Graph Properties vor:

yvalue(output.Vb,1.9e9)

und das ist nicht schwer zu verstehen. Es heißt:

„Zeige den Wert der mit Harmonic Balance ermittelten Ausgangsspannung „output.Vb“ für die Frequenz f = 1,9 GHz und für Eingangsspannungen von 10mV bis 800mV“. Dies wird für die Werte 2V / 3V / 4V / 5v der Betriebsspannung wiederholt.

3.5. Die „Forward Transmission S21“ als Funktion von u_RF

Die Formel für S21 findet sich – natürlich mathematisch korrekt formuliert – in den passenden Lehrbüchern für Nachrichten-Übertragung und lautet:

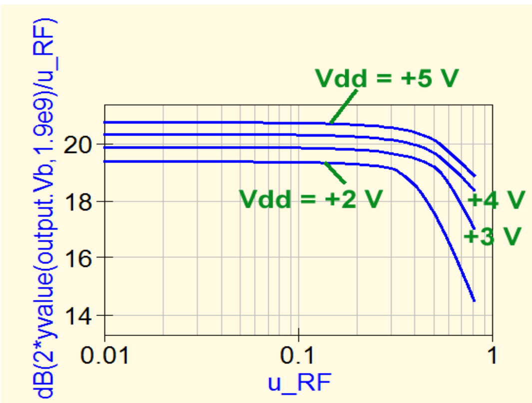
S21 = Ausgangsspannung (hier: bei 1,9 GHz) geteilt durch die Hinlaufende Welle mit U = u_RF/2

Und ganz rund wird die Sache, wenn man gleich die Ausgabe in „dB“ vornimmt. Dann muss man als Graph Property schreiben:

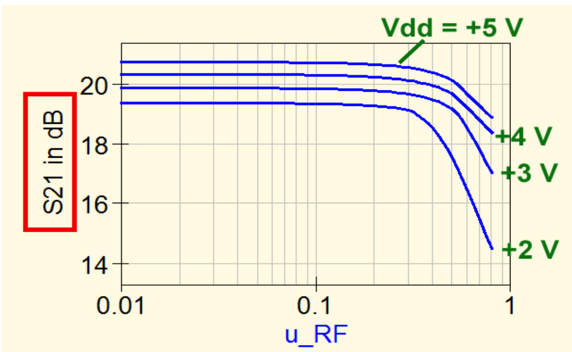
dB(yvalue(2*yvalue(output.Vb, 1.9e9) / u_RF))

...und noch ein Tipp. Bitte eine passende Achsbezeichnung selbst wählen:

Senkrechte Achse mit Formel



Senkrechte Achse mit geändertem Label „S21 in dB“



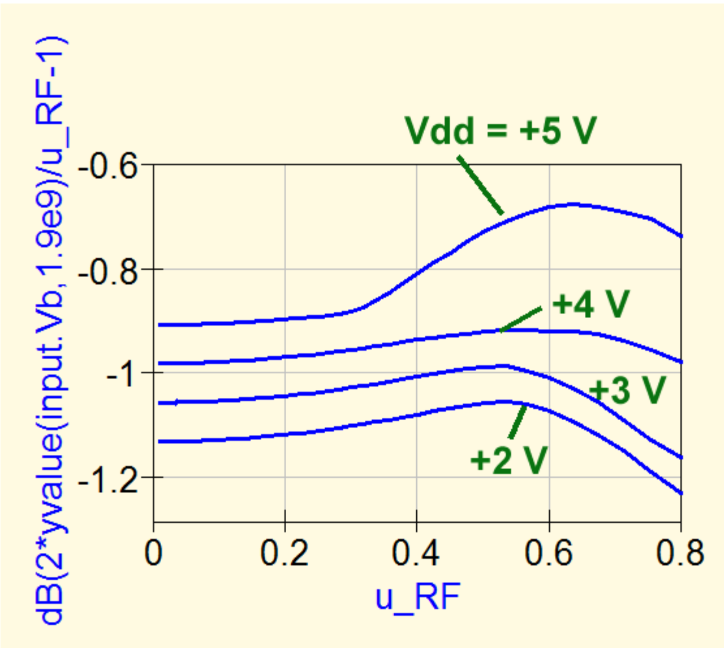
3.6. Die „Input Reflection S11“ als Funktion der Eingangsspannung

Dazu ist wieder ein Blick in das Nachrichtentechnik-Fachbuch nötig und dort steht:

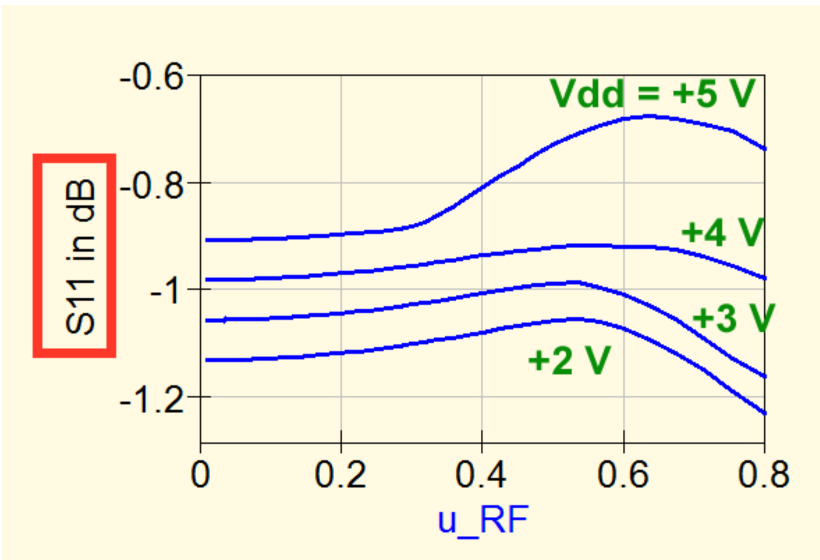
Teile zuerst die Eingangsspannung (hier: f = 1,9 GHz) durch die Hinlaufende Welle mit U = u_RF /2. **Von diesem Ergebnis ziehe hinterher die Zahl „1“ ab.** **Dann hast Du S11.** **Und am besten folgt gleich eine Umrechnung in dB....**

Für unser qucsstudio gibt das natürlich – zusammen mit der **dB-Anzeige** – eine Riesenschlange als Graph Property – Eingabe....und mit den vielen Klammern muß man verflixt aufpassen....

dB(2*yvalue(input.Vb,1.9e9)/u_RF-1)



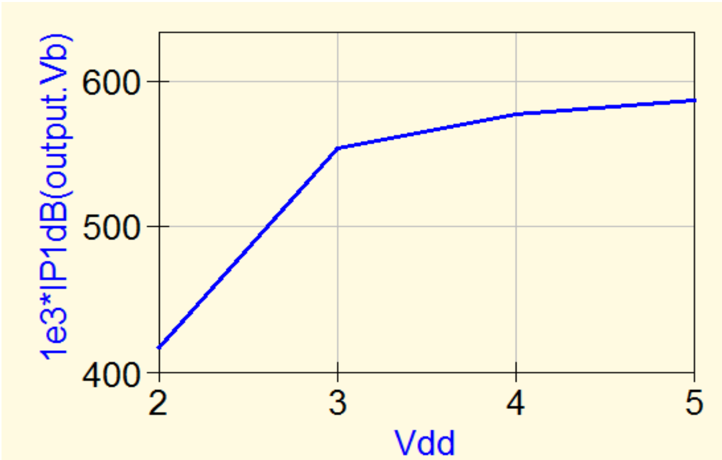
Das ist der „Roh-Zustand“ mit der Angabe der kompletten Gleichung an der senkrechten Achse.....



...aber so sieht es besser aus.

3.7. Der Input- und Output- 1dB Compression Point

Wenn man die Eingangsleistung eines Verstärkers immer weiter erhöht, dann gerät er schließlich in die **Begrenzung**. Man merkt es daran, dass plötzlich die Ausgangsleistung „zurückhängt", also nicht mehr im gleichen Mass ansteigt und irgendwann konstant bleibt. Hier hat man sich bei der Datenblatt-Angabe auf einen Wert geeinigt, bei dem „**1 dB bei der Ausgangsleistung fehlt**" und bezeichnet diesen Punkt als „**1dB- Kompression**". Den kann man aber für die Eingangsspannung ODER die Ausgangsspannung angeben.

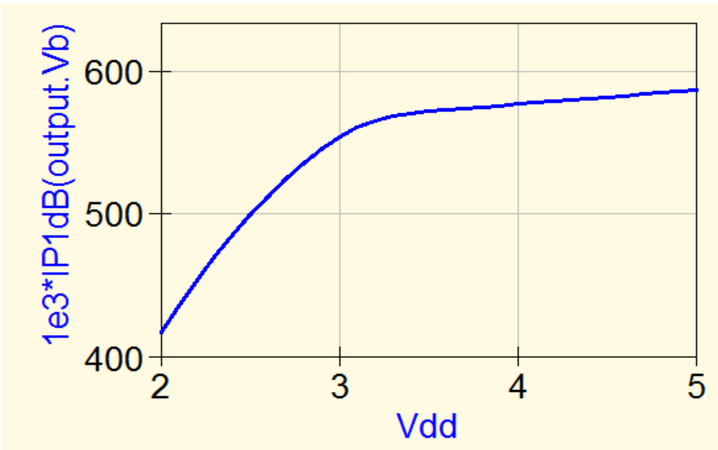


Für den **Input 1 dB Compression Point** in Abhängigkeit von der Betriebsspannung gilt die Graph Property

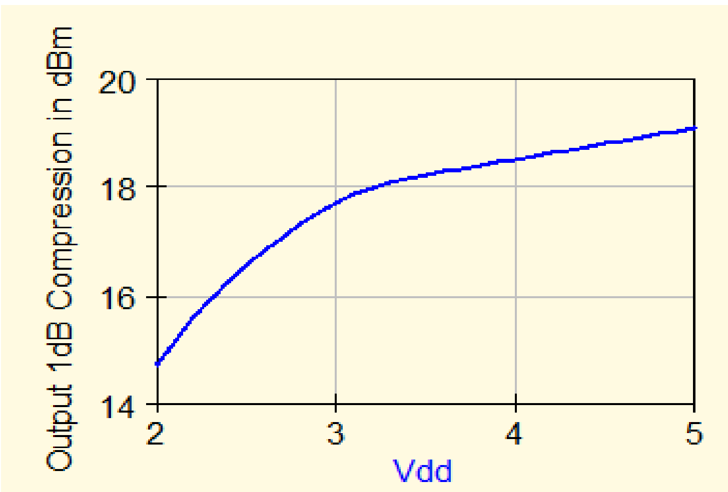
1e3*IP1dB(output.Vb)

(**Der Faktor 1e3 = 1000 am Anfang sorgt für eine Kalibrierung der Anzeige in Millivolt)**

Das Ergebnis sieht natürlich recht „eckig" aus, da wir bei der waagrechten Achse mit Schritten von 1V arbeiten.



Wenn wir dagegen im **Parametersweep SW1 durch die Eingabe von 31 statt 4 Simulationspunkten** eine Auflösung von 0,1V erzielen, sieht das Ganze schon viel besser aus.



Nun wollen wir den **Output 1 dB Compression Point für Vdd = +2V.....+5V** bestimmen – **aber gleich in dBm und mit schönem Y-Achsen -Label!**

Das geschieht durch

dBm(OP1dB(output.Vb, 1.9e9))

4. Zusätzliche Simulationen beim FET-Verstärker-Beispiel

Nachdem wir das bei qucsstudio mitgelieferte Beispiel genügend ausgequetscht haben, wollen wir noch einige interessante Sachen zusätzlich untersuchen und darstellen. Der riesige Datensatz des Simulationsergebnisses beim Parameter-Sweep reizt einfach dazu....

4.1. Die Sache mit den Oberwellen

Ohne Verzerrungen geht es bei Verstärkern nie ab und deshalb kann man hierzu viele Fragen stellen. Zum Beispiel:

a) Wie sieht das Oberwellen-Spektrum für eine bestimmte zugeführte Frequenz aus, wenn wir die Eingangsspannung und die Betriebsspannung konstant halten?

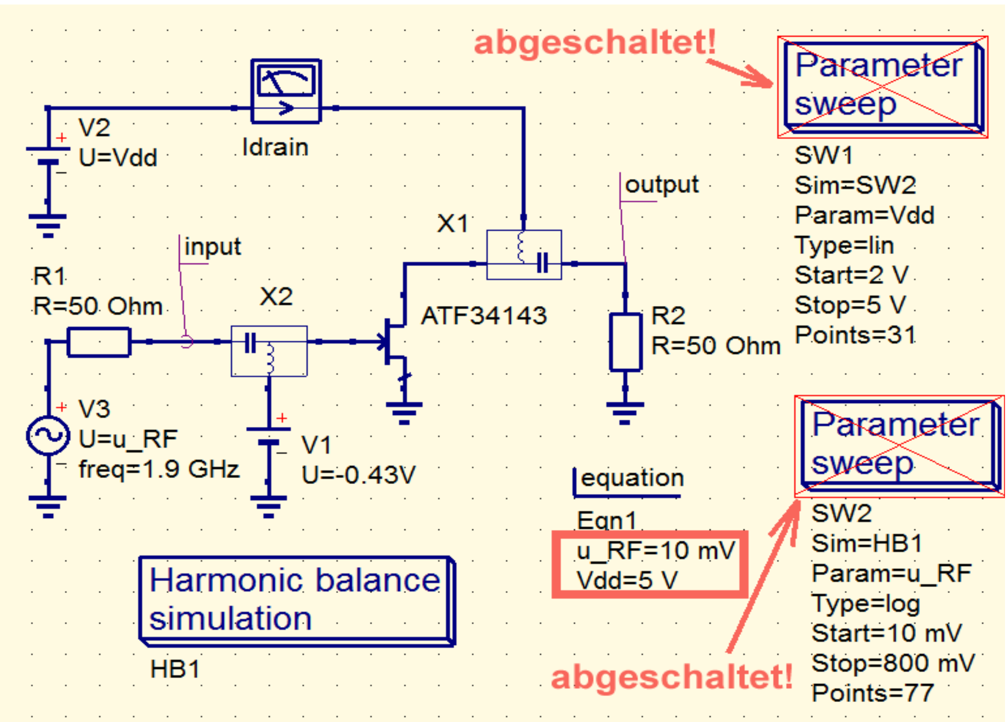
b) Wie sieht das Oberwellen-Spektrum für eine bestimmte zugeführte Frequenz aus, wenn wir die Eingangsspannung erhöhen und zusätzlich die Betriebsspannung variieren?

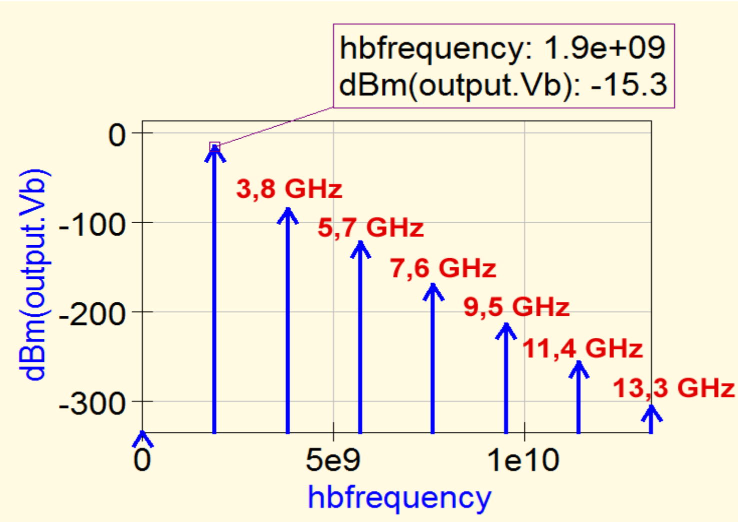
c) Wie sieht das Oberwellen-Spektrum aus, wenn wir die Eingangsfrequenz in einem bestimmen Bereich sweepen? Und wie ändert sich dieses Spektrum mit der Eingangsspannungs-Amplitude bzw. der Betriebsspannung bzw. dem Ruhestrom? Usw.....usw....usw...

Das erforderliche grundsätzliche Vorgehen sehen wir uns für die Frage a) und eine **konstante Betriebsspannung mit Vdd = +5 V** näher an. Dazu setzen wir den **Parameter-Sweep SW1 nach dem Anklicken von „Deactivate" (...findet man nach einem Rechtsklick auf den Parameter-Sweep SW1 in seinem Property Menü - Aufruf) außer Betrieb**. Ausserdem brauchen wir nun einen **Startwert in „equations" von Vdd = +5 V.**

4.1.1 Oberwellenspektrum bei konstanter Eingangsspannung und Betriebsspannung (Vdd = +5V)

Wir steuern die Schaltung nur mit einem **Signal von 1,9 GHz an, das eine konstante Amplitude von u_RF = 10 mV** aufweist. Dazu wird **auch der Parameter-Sweep SW2 (mit „Deactivate" nch einem Rechtsklick auf sein Symbol) abgeschaltet** und dieser Startwert von 10 mV für u_RF unter „equations" eingetragen. Anschließend kann simuliert werden.





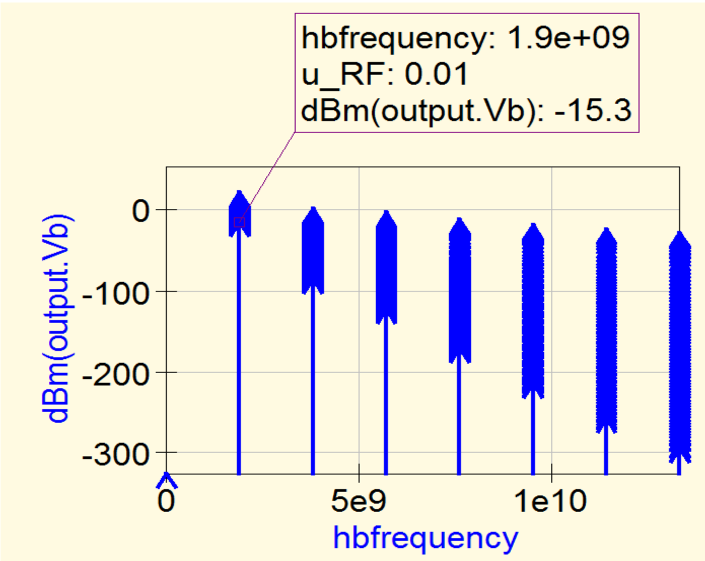
Das ist das Simulationsergebnis, nachdem ein kartesisches Diagramm aufgerufen und als „Graph Properties“

dBm(output.Vb)

verwendet wurde. Mit einem Marker ist die Grundfrequenz von 1,9 GHz gekennzeichnet und ihr Ausgangspegel beträgt -15,3 dBm.

Durch den Einsatz weiterer Marker lassen sich jetzt sehr einfach die Dämpfungen der einzelnen Harmonischen bestimmen.

4.1.2. Oberwellenspektrum bei konstanter Frequenz und Betriebsspannnung, aber unterschiedlicher Aussteuerung (= u_RF wird geweept)



Wir geben jetzt den **Parameter-Sweep SW2 frei** (...geht wieder über „Deactivate / Activate“) **und sweepen damit (logarithmisch) die Amplitude der Eingangs-Spannung von 10 mV bis 800 mV mit 77 Punkten.** Die Betriebsspannung bleibt weiterhin Vdd = +5 V.

Die dBm-Darstellung wird beibehalten, aber nun sehen die Spektrallinien irgendwie aus wie Schilfgewächse. Ist aber logisch, denn die Simulationsergebnisse der 77 simulierten Punkte werden alle übereinander geschrieben. Da müssen wir etwas tun...

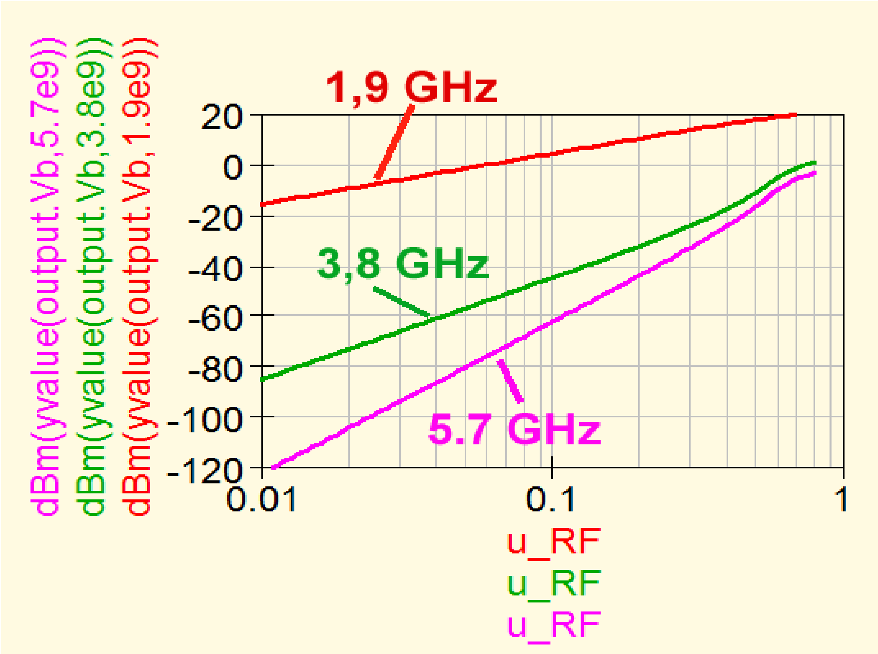
Abhilfe:
Darstellung der Grundwelle sowie der zweiten und der dritten Harmonischen in Abhängigkeit von der Eingangsspannung

Jetzt machen wir Nägel mit Köpfen und lassen uns die **Amplituden der Grundwelle (f = 1,9GHz) sowie der doppelten und der dreifachen Frequenz zusammen in einem Diagramm mit „dBm“-Teilung in Abhängigkeit von der Eingangsspannung anzeigen**.
Damit können direkt die Dämpfungen bei einer bestimmten Amplitude des Eingangs-Signals abgelesen werden. Das erfordert beim Ergebnis-Diagramm eine **logarithmische X-Achse** sowie folgende Graph-Properties:

den Pegel der **Grundwelle** sehen wir mit **dBm(yvalue(output.Vb,1.9e9))**

den Pegel der **doppelten Frequenz** sehen wir mit **dBm(yvalue(output.Vb,3.8e9))**

den Pegel der **dreifachen Frequenz** sehen wir mit **dBm(value(output.Vb,5.7e9))**

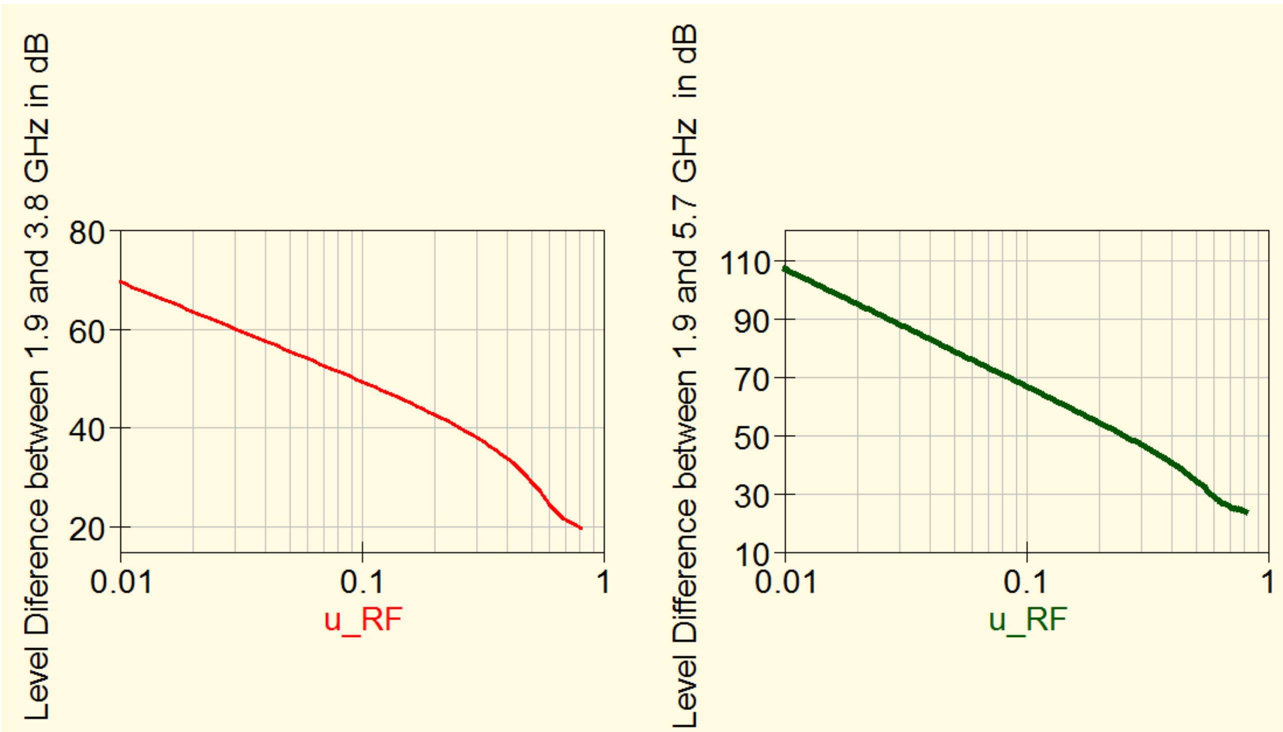


Dieses Bild zeigt den Erfolg. Ausserdem lassen sich jederzeit weitere Oberwellen hinzufügen.

Aufgabe:
Stellen Sie die „Pegeldifferenz in dB“ zwischen der Grundwelle (1,9 GHz) und der zweiten Harmonischen (3,8 GHz) in einem eigenen Diagramm dar.

Wiederholen Sie diese Darstellung für 1,9 GHz und 5,7 GHz.

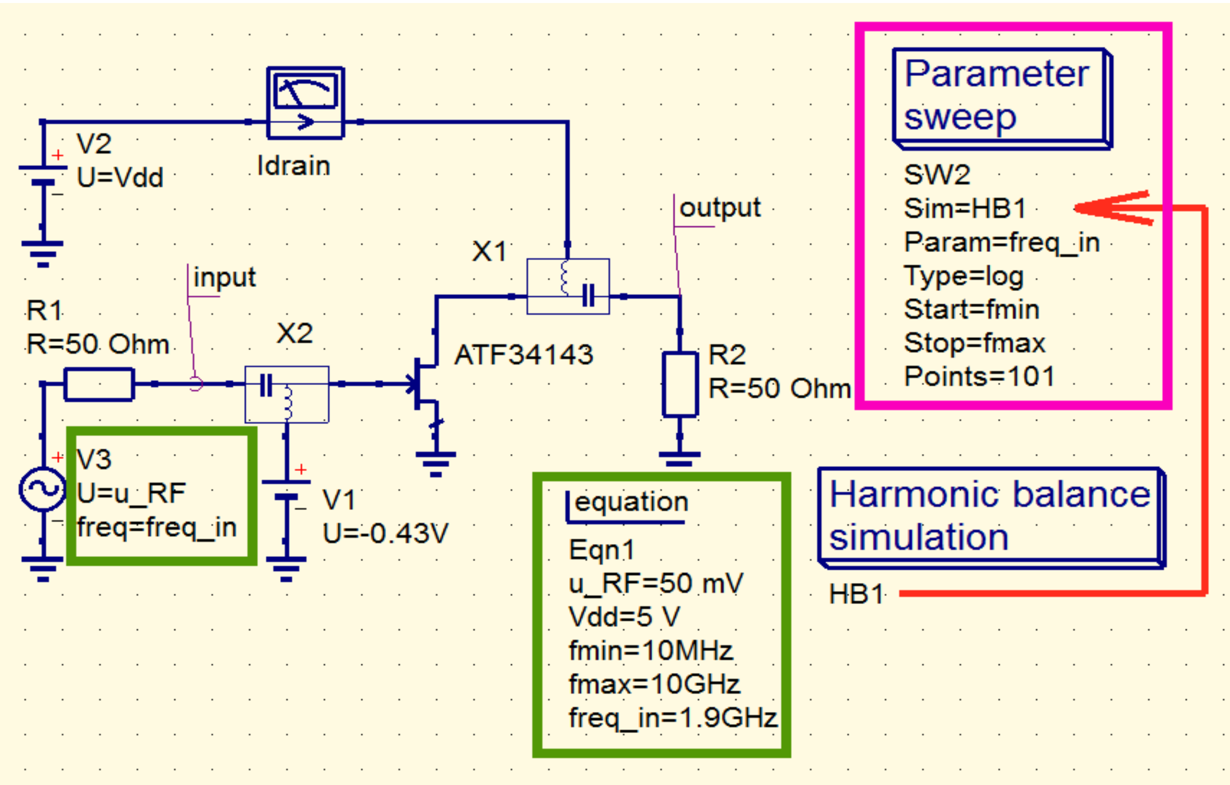
(Information:
dieser Pegelunterschied wird in der Praxis oft als „Klirrdämpfung“ bezeichnet)



4.2. Jetzt wird die Frequenz gesweept.

4.2.1. Die Ausgangsspannung als Funktion der Frequenz und der Eingangsspannung

Dazu müssen wir vorher etwas umbauen, bevor wir die Ausgangsspannung bei verschiedenen Frequenzen und verschiedenen Eingangsspannungs-Amplituden darstellen können:



Bei der **Eingangsspannung** arbeiten wir den beiden Variablen **„u_RF“** und **„freq_in“**.

Unter „**equation**“ weisen wir den verwendeten Variablen ihre Startwerte zu:

u_RF=50mV
Vdd=5V
freq_in=1.9GHz

Dazu geben wir noch den **Minimal- und Maximalwert für die gesweepte Frequenz** vor:

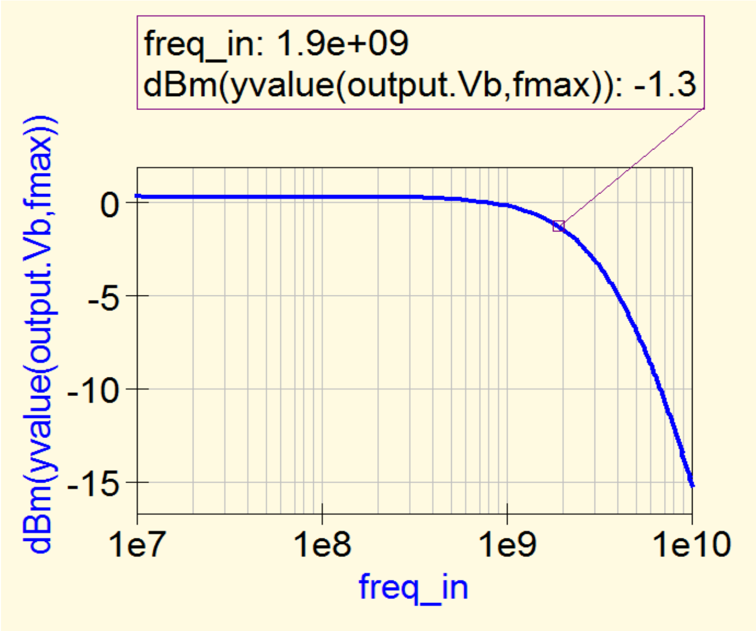
fmin=10MHz
fmax=10GHz

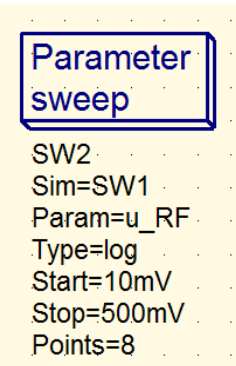
Dann brauchen wir die Harmonic Balance Simulation und einen Parameter Sweep.
Der Parameter Sweep sorgt für eine komplette Berechnung der Schaltung bei 101 Frequenzpunkten zwischen fmin = 10 MHz und fmax = 10 GHz (logarithmischer Sweep).

Wir wollen uns nun die Ausgangsspannung „output“ als Pegel in dBm für diesen Frequenzbereich ausgeben lassen und brauchen dazu nach der Simulation ein kartesisches Diagramm. Die nötige Graph Property lautet:

dBm(yvalue(output.Vb,fmax))

und das kommt dabei heraus (...der Marker zeigt den Wert bei f = 1,9 GHz)



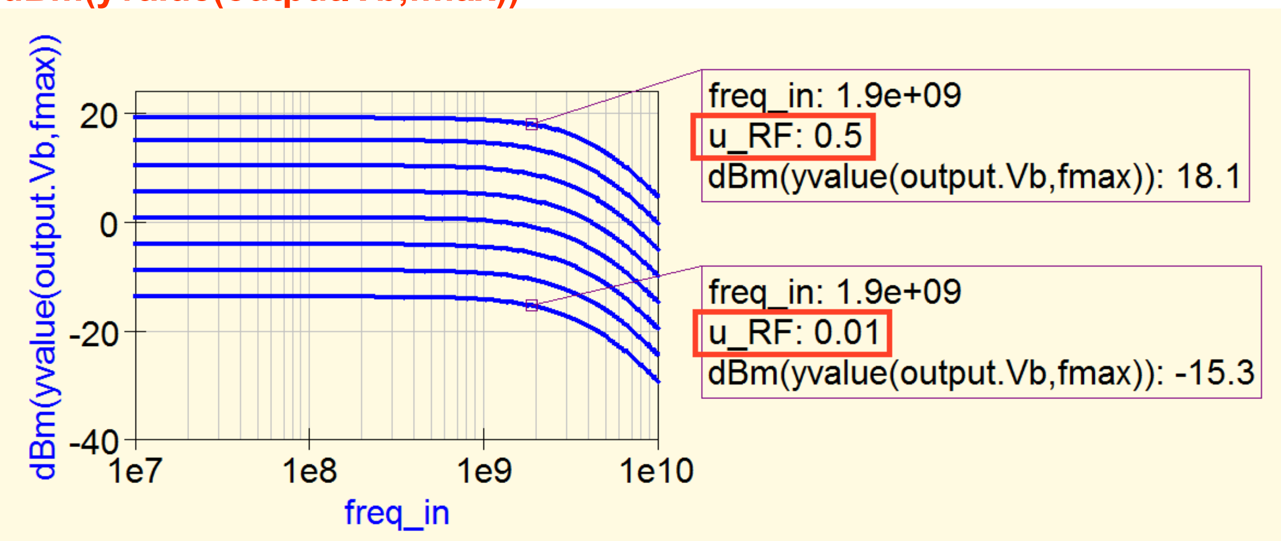


Jetzt könnte man z. B. für Eingangsspannungen mit 8 Stufen zwischen 10 mV und 500mV nach der Ausgangsspannung in Abhängigkeit von der Frequenz fragen. Das wird über diesen zusätzlichen Parametersweep gelöst.

(...Bitte die Eingangsspannung nicht höher wählen, sonst gibt es Rechenprobleme wegen Übersteuerung. Ebenso sind viel mehr Punkte nicht nötig – sie treiben nur die Rechenzeit hoch).

Die gesuchte Lösung erhalten wir mit der Graph Property

dBm(yvalue(output.Vb,fmax))



4.2.2. Die Forward Transmission S21 als Funktion der Frequenz und der Eingangsspannung

Aufgabe:

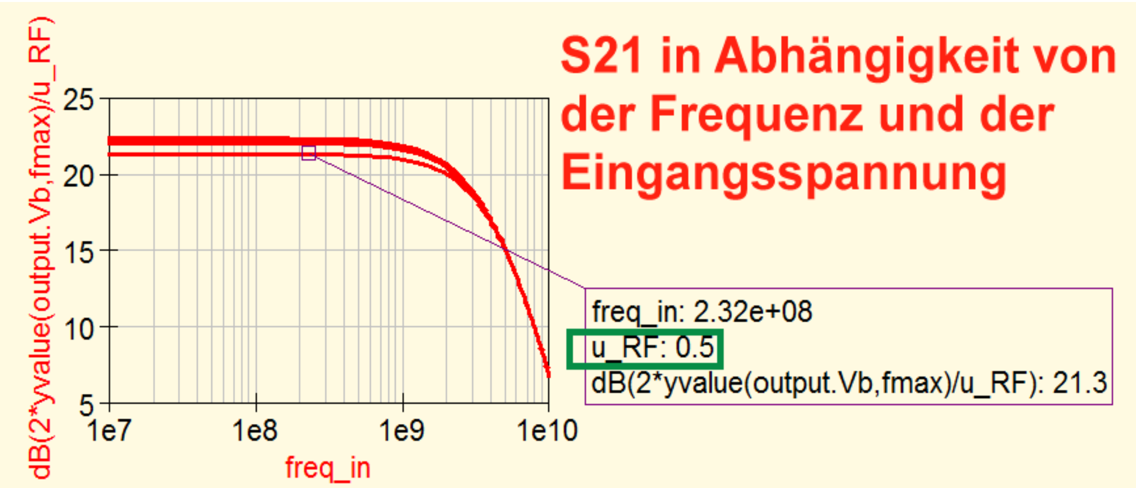
Stellen Sie die „Forward Transmission S21 in dB“ für diese acht verschiedenen Eingangsspannungen in einem Diagramm in Abhängigkeit von der Frequenz dar. Benützen Sie eine logarithmisch geteilte Frequenzachse. Da braucht man erst mal die Formel zur Bestimmung von S21:

S21 = Ausgangsspannung, geteilt durch die Hinlaufende Welle mit U = u_RF/2

Das ergibt folgende Graph Property

dB(2*yvalue(output.Vb,fmax)/u_RF)

So sieht das aus – und bei u_RF = 0,5 V hat die Übersteuerung schon begonnen....denn wenn man genau hinschaut, sieht man eine Abnahme um 1dB.....da war doch was mit der 1dB Kompression...



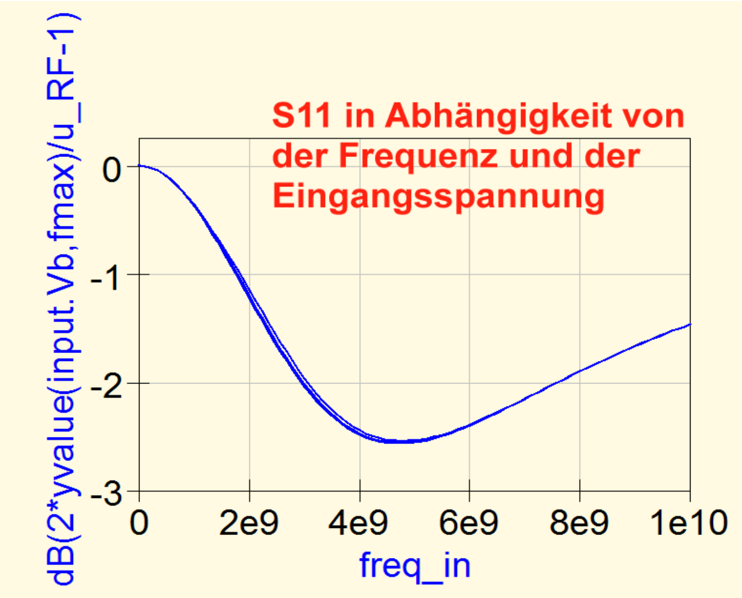
4.2.3. Die „Input Reflection S11“ als Funktion der Frequenz und der Eingangsspannung

Dazu ist wieder der Blick in das Nachrichtentechnik-Fachbuch nötig und dort steht:

Teile zuerst die Eingangsspannung (hier: f = 1,9 GHz) durch die Hinlaufende Welle mit U = u_RF /2. **Von diesem Ergebnis ziehe hinterher die Zahl „1“ ab.** **Dann hast Du S11.** **Und am besten folgt gleich eine Umrechnung in dB....**

Für unser qucsstudio gibt das natürlich – zusammen mit der **dB-Anzeige** – wieder eine Riesenschlange als Graph Property – Eingabe....und mit den vielen Klammern muß man verflixt aufpassen....

dB(2*yvalue(input.Vb,fmax)/u_RF-1)



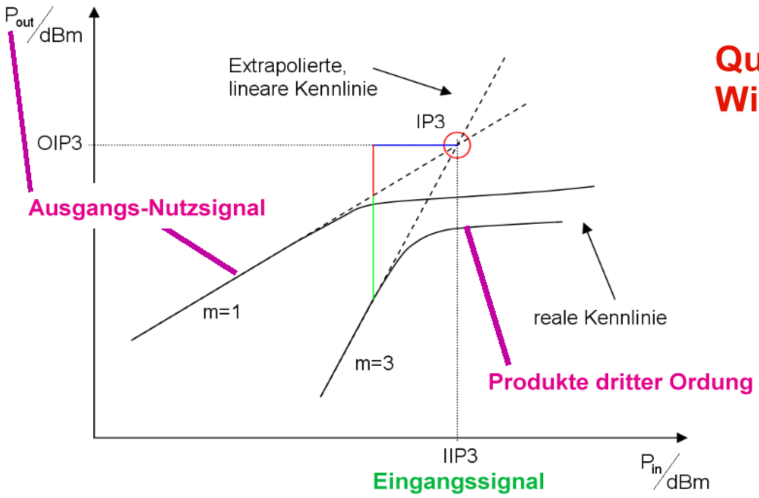
Wie man sieht, ändert sich S11 nur mit der Frequenz, aber kaum mit der Amplitude der Eingangsspannung.

----

5. Bestimmung des IP3 - Punktes

5.1. Was steckt hinter dem IP3 Punkt?

Wenn man die Eingangsleistung eines Verstärkers immer weiter erhöht, dann gerät er schließlich in die **Begrenzung**. Man merkt es daran, dass plötzlich die Ausgangsleistung „zurückhängt“, also nicht mehr im glleichen Mass ansteigt und irgendwann konstant bleibt. Hier hat man sich bei der Datenblatt-Angabe auf einen Wert geeinigt, bei dem „**1 dB bei der Ausgangsleistung fehlt**“ und bezeichnet diesen Punkt als „**1dB- Kompression**“. Zum **Eingangspegel** kommt man immer, indem man vom **Ausgangspegel (in dBm) die gerade dort geltende Verstärkung (in dB) abzieht und dabei den Verstärkungs-Rückgang durch die einsetzende Begrenzung berücksichtigt**



Quelle: Wikipedia

Aber lange, bevor man in die Begrenzung gerät, beginnen die Verzerrungen am Ausgang deutlich anzusteigen. Besonders unangenehm sind dabei die **Produkte dritter Ordn**ung, da sie dreimal schneller steigen als das zugeführte Eingangssignal **UND in der Nähe der zugeführten Speisefrequenz liegen**. Dafür gilt:
Wird der Eingangspegel um 10 dB erhöht, dann nimmt der Abstand des Nutz-Ausgangssignals zu den Störprodukten dritter Ordnung um 20 dB ab!

*Bevor sich jedoch die beiden Kennlinien schneiden, setzt die vorhin besprochene Begrenzung ein. Würde man jedoch diese beiden Geraden gedanklich bis zu ihrem Schnittpunkt verlängern, dann käme man **zum „Intermodulation Intercept Point, Third Order“, kurz „IP3“ genannt. Er kann für die Eingangs- oder die Ausgangsseite angegeben werden, aber man zieht die Ausgangsangabe vor (= OIP3). Mit seiner Hilfe kann später der „intermodulationsfreie Dynamikbereich“ sowie für jede beliebige Eingangsspannungs-Amplitude der Abstand zu den entstehenden Produkten dritter Ordnung über einfache Geradengleichungen (rechnerisch oder grafisch) bestimmt werden.**
Die Untergrenze für kleine Signale wird hierbei durch das Eigenrauschen der Stufe festgelegt, in dem die Störprodukte versinken. Die Obergrenze des Dynamikbereiches zieht dann z. B. der 1 dB-Kompressionspunkt.

5.2. Bestimmung des IP3-Punktes (durch Zweiton-Messung oder Simulation)

Dazu steuert man die Stufe mit **zwei gleich großen Signalen** an, die nur einen geringen Frequenzunterschied aufweisen. Mit zunehmender Amplitude nehmen die „Störprodukte dritter Ordnung“ dreimal schneller zu als die beiden Eingangssignale und man findet plötzlich am Ausgang im Spektrum direkt neben den gewünschten Nutzsignalen diese unerwünschten Vögel.

Ihr Frequenzabstand zu den beiden Eingangsspannung entspricht exakt dem Frequenzunterschied der Ansteuersignale!

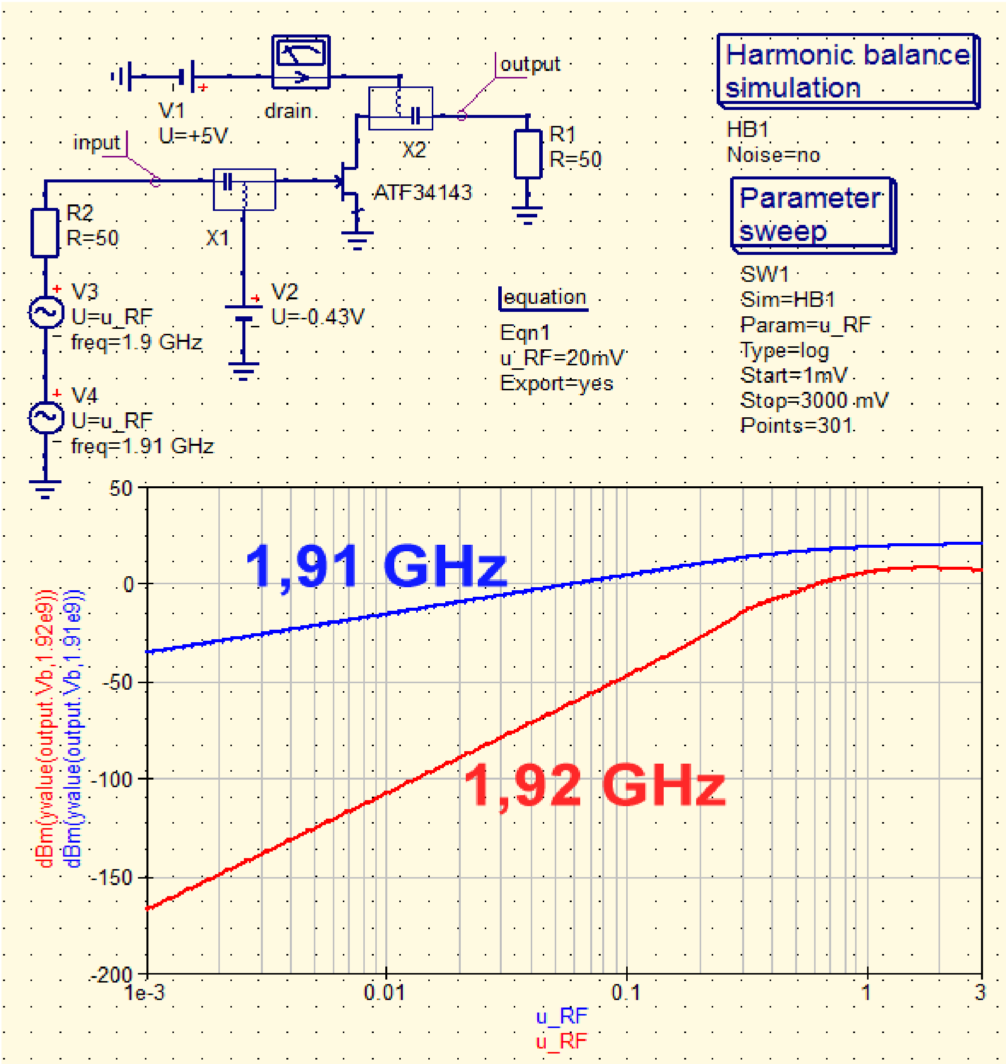
5.2.1. IP3-Simulation mit qucsstudio

Wir bleiben bei unserem Beispiel, schalten aber beim Eingang eine weitere Spannungsquelle in Reihe. Beide Ansteuersignale sind gleich groß und weisen die **Frequenzen 1,9 GHz bzw. 1,91 GHz** auf. Ihr **Frequenzabstand beträgt also 10 MHz.** Dann müssen wir die „Störprodukte bei

(1,9 GHz - 10 MHz) = 1,89 GHz **und** **(1,91 GHz + 10 MHz) = 1,92 GHz**

suchen. Wir verbleiben bei unserem Parameter-Sweep SW1 und sehen darin einen Amplitudenbereich von 1 mV bis 3 V mit 301 Punkten vor.
Nach der Simulation brauchen wir ein kartesisches Diagramm und lassen uns mit den beiden Gleichungen:

dBm(yvalue(output.Vb,1.91e9) sowie **dBm(yvalue(output.Vb.1.92e9)**



die Ausgangleistungen in dBm für das „obere Nutzsignal mit 1,91 GHz" und sein „oberes Störprodukt" (f = 1,92 GHz) mit steigender Eingangsamplitude anzeigen.

(Hinweis:
die Formel für die Bestimmung der Frequenzen für diese betrachteten ersten Störprodukte dritter Ordnung lautet:

$f_{Stör} = 2*f_1 - f_2$

Wer nachrechnet, kommt auf unsere 1,89 GHz bzw. 1,92 GHz und es ist dabei egal, ob wir das obere oder untere Störsignal betrachten).

Das sehen wir sehr schön im obigen Bild, das nicht nur die Schaltung mit den zwei Quellen, sondern auch das Diagramm mit den gewählten Ausgangssignalen enthält. Aber:

Entscheidend für die korrekte Darstellung ist dabei eine logarithmisch geteilte x – Achse. Nur damit erhalten wir im linearen Teil der Kurven auch wirklich Geraden (...“dBm“ ist halt ein logarithmisches Maß und deshalb benötigen wir ein doppelt-logarithmisches Diagramm...).

Nun wird es spannend, denn durch die beiden Kurven müssen wir Geraden legen und mit ihnen jeweils den linearen Kurventeil annähern. Das ist nicht allzu schlimm und wir üben es gleich mal am Verlauf der „Grundwelle mit 1,91 GHz“. Dafür schreiben wir eine weitere Geradengleichung, mit der die verstärkte Eingangsspannung beschrieben wird. Bei dem bereits früher bestimmten Verstärkungswert von ca. 5,5 lautet sie:

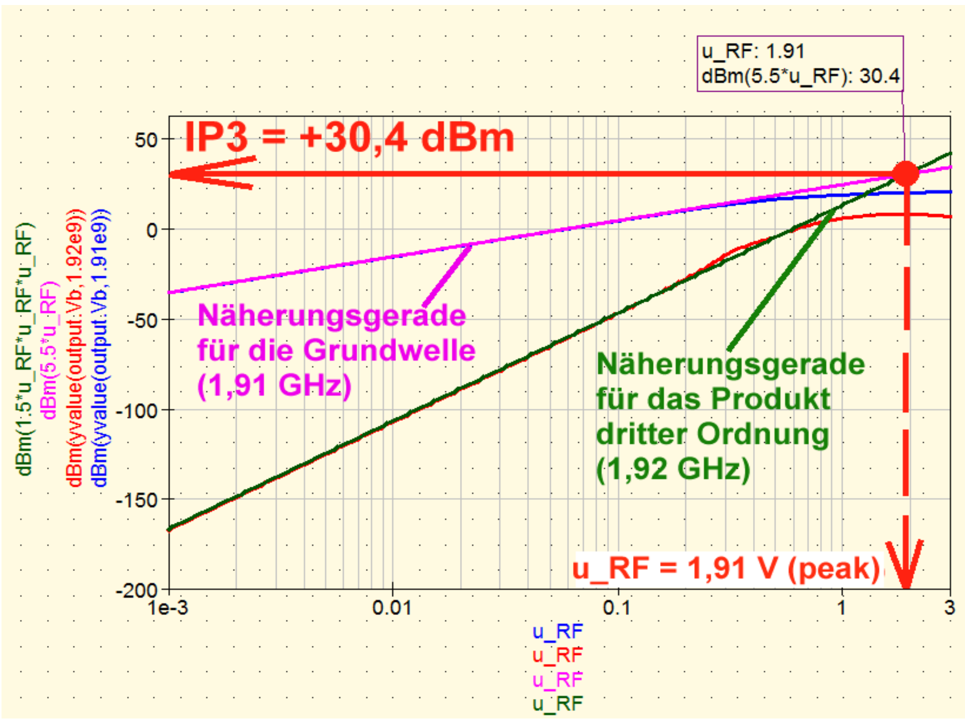
dBm(5.5*u_RF)

Doch jetzt zum Störprodukt dritter Ordnung“ mit f = 1,92 GHz. **Bei ihm steigt die Amplitude mit (u_RF)^3 an** und deshalb brauchen wir eine Graphengleichung, die grundsätzlich so aufgebaut ist:

dBm(u_RF*u_RF*u_RF)

Wenn man diese Gerade in unser Diagramm einträgt, so zeigt sich, dass sie noch etwas nach oben geschoben und deshalb mit einem Faktor multipliziert werden muss. Nach einigen Versuchen landet man dann bei der endgültigen Gleichungsversion

dBm(1.4*(u_RF*u_RF*u_RF))



und in diesem Bild ist nun sehr schön der gesuchte Schnittpunkt zu erkennen. Man bestimmt ihn exakt mit einem Marker und sieht dann:

Der Output Intercept Point OIP3“ liegt bei einem Ausgangspegel von

+30,4 dBm.

Der Spitzenwert der zugehörigen **Eingangsspannung** beträgt (Siehe die waagrechte Achse...) **u_Rf = 1,91 V** und das entspricht einem Effektivwert von **1,91V / sqrt(2) = 1,35 V**

bzw einem Pegel von **20*log10(1,38 / 0,224V) dBm = +15,6 dBm**

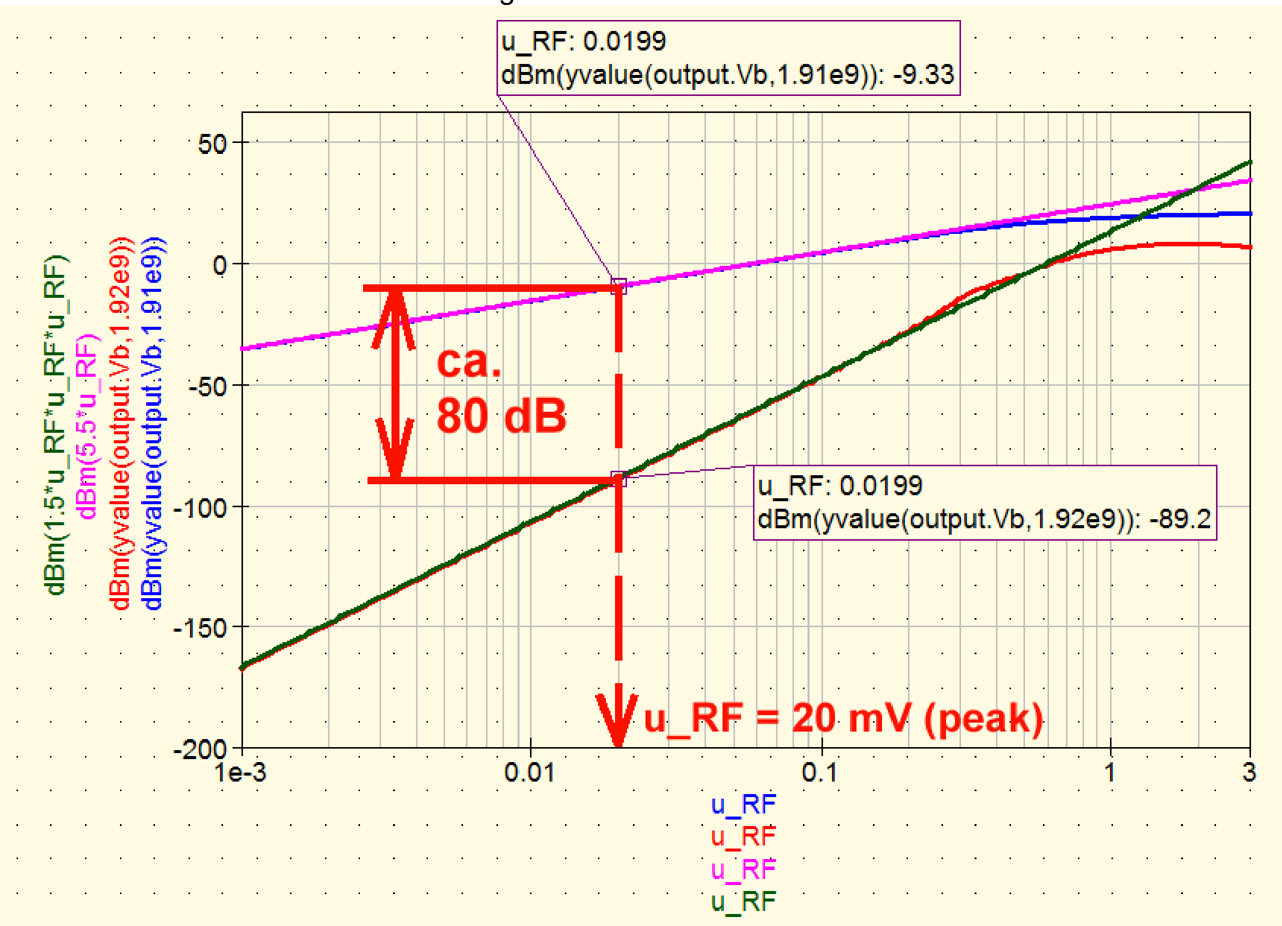
Das ist der zugehörige **„Input Intercept Point IIP3“ mit +15,6 dBm.**

5.2.2. Eine praktische Anwendung

An die Stufe werden beide Signale (1,9 und 1,91 GHz) mit einer Amplitude von je u_RF = 0,02 V (= Spitzenwert, der Effektivwert beträgt dann 14,1 mV, der Pegel = -24 dBm) angelegt. **Um wieviel dB liegen nun die Störprodukte dritter Ordnung unter diesem Wert**?

Lösung 1:

Trägt man in das obige Diagramm eine senkrechte Gerade bei u_RF = 0,02 V (peak) ein, so schneidet diese die beiden Näherungsgeraden. Der Pegelunterschied der beiden Schittpunkte wird über zwei Cursoren ermittelt und beträgt **80 dB.**



(**Zusatzaufgabe:**
Die Theorie sagt, dass sich der Pegelunterschied um 20 dB verschlechtert, wenn der Eingangspegel um 10 dB ansteigt. Prüfen Sie diese Behauptung am obigen Diagramm nach).

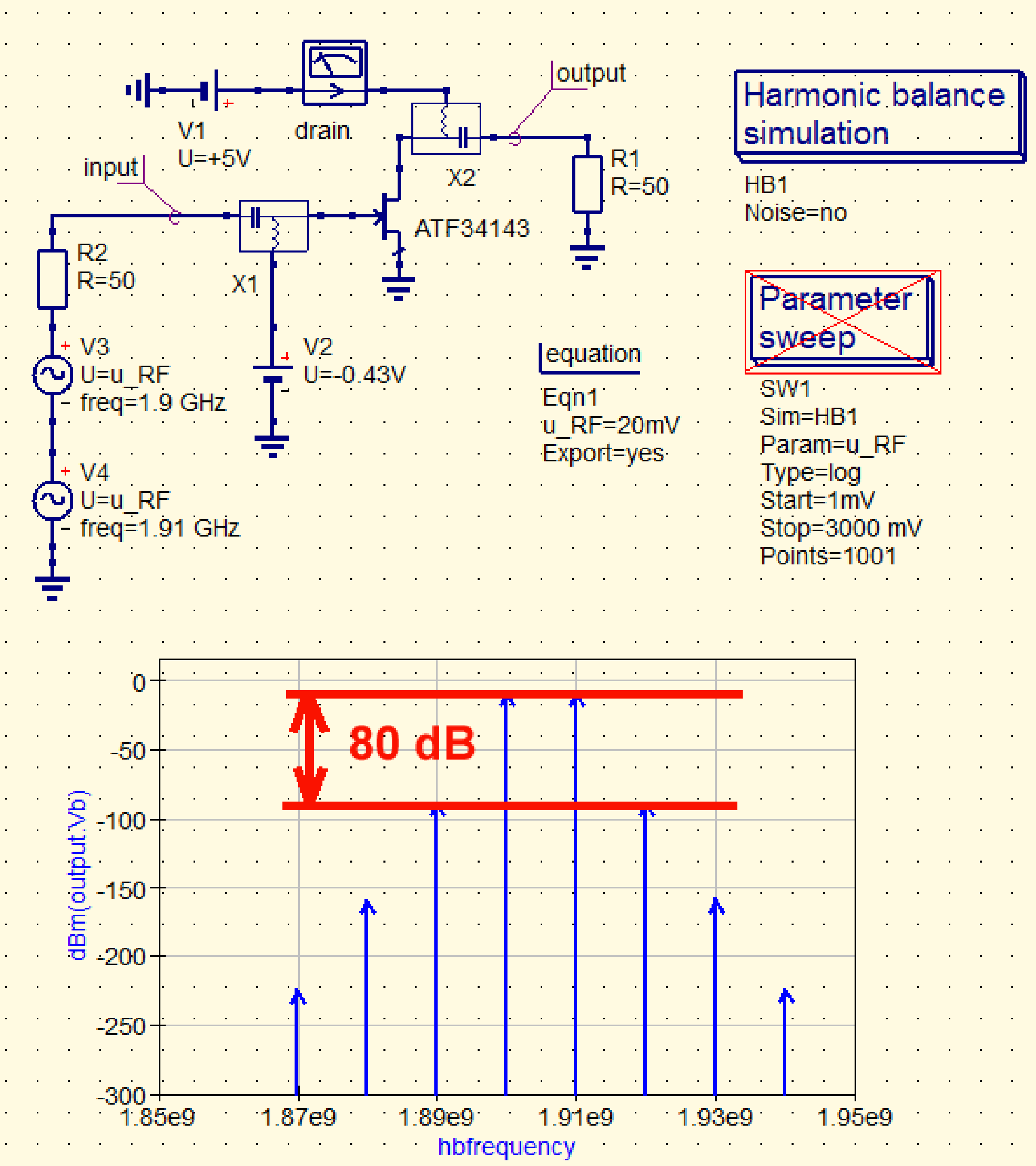
Lösung 2: Bestimmung des IP3-Punktes über das Spektrum

Da wird zuerst der Parameter-Sweep abgeschaltet und kontrolliert, ob unter „equations“ ein Spitzenwert von 20 mV für beide Eingangssignale eingetragen ist.

Als Simulationsergebnis wollen wir

dBm(output.Vb)

sehen. Aber wir lassen uns nur den Frequenzbereich zwischen 1,85 und 1,95 GHz anzeigen.



Das Ergebnis sollte uns doch sehr bekannt vorkommen! Wir sehen die beiden Eingangssignale mit 1,9 GHz und 1,91 GHz, dazu die im letzten Kapitel erwähnten beiden „Störprodukte dritter Ordnung“ mit 1,89 GHz und 1,92 GHz. **Der Pegelabstand zwischen Nutz- und Störsignalen beträgt 80 dB – und exakt das war das Ergebnis von Lösung 1.**

Aber wegen des großen dargestellten Pegelbereiches tauchen jetzt **weitere Störprodukte auf. Sie haben eine noch höhere Ordnung und steigen bei Erhöhung des Eingangspegels noch schneller an** (z. B. haben wir bei den nächsten Produkts den Grad N = 5 und deshalb die fünffache Steigung gegenüber den Eingangssignalen….da wird es schnell zappenduster, wenn man weiter aufdreht...).