

Erweiterte statistische Simulation elektronischer Schaltungen

Wer schon mal mit Spice eine elektronische Schaltung simuliert hat kennt vielleicht die Funktionen mit den man die Auswirkung der Bauteil-Toleranz untersuchen kann.

Da ist zum einen die Monte Carlo Simulation, dabei kann man Bauteilwerte mit einer Toleranz versehen und bei jedem Simulationslauf werden die zufällig Bauteilwerte innerhalb der vorgegebenen Toleranz verändert. Man erhält als Ergebnis eine Kurvenschar, und kann darin die mit dem Cursor die Breite der Toleranz bestimmen.

Weiterhin gibt es in manchen Spice Simulationsprogrammen die Möglichkeit der Parameteranalyse, dabei wird der Reihe nach ein Bauteil mit der maximal vorgegebenen Toleranz verändert und alle anderen Bauteile weisen den Nominalwert auf. Das Ergebnis zeigt den Einfluss jedes einzelnen Bauteils, wobei zur Auswirkung wenn auch andere Werte eine Abweichung zum Nominalwert besitzen gibt es damit keine Aussage.

In einigen Programmen gibt es eine „echte“ Worst Case Analyse die auf Basis der Parameteranalyse alle Bauteiltoleranzen so verändert, dass die betrachtete Ausgangsgröße entweder Minimal wird oder Maximal, „unechte“ Worst Case Analysen simulieren alle MIN MAX Variationen.

Anm.: Der Nachteil ist der unechten Worst Case Analyse ist dass die Anzahl Simulationsdurchläufe exponentiell mit der Anzahl der Bauteile steigt. Bei 10 Bauteilen gibt es 2^{10} Durchläufe=1024, bei der „echten“ Parameteranalyse sind es nur 22 (10 x min 10x max, 1x Worst Case pos. Auslenkung 1x Worst Case neg. Auslenkung), Bei 20 Bauteilen sind es 1048576 zu 42 Durchläufen.

Dies Min Max Kurven sind in der Regel weiter auseinander als die Breite der Monte Carlo Kurvenschar. Wie weit hängt von der Anzahl der Monte Carlo Simulationen ab. Das ist unser erstes Problem, wenn die Anzahl die visuelle Bewertung beeinflusst, wann kann man sich sicher sein, dass das Ergebnis ok ist. Dazu aber später.

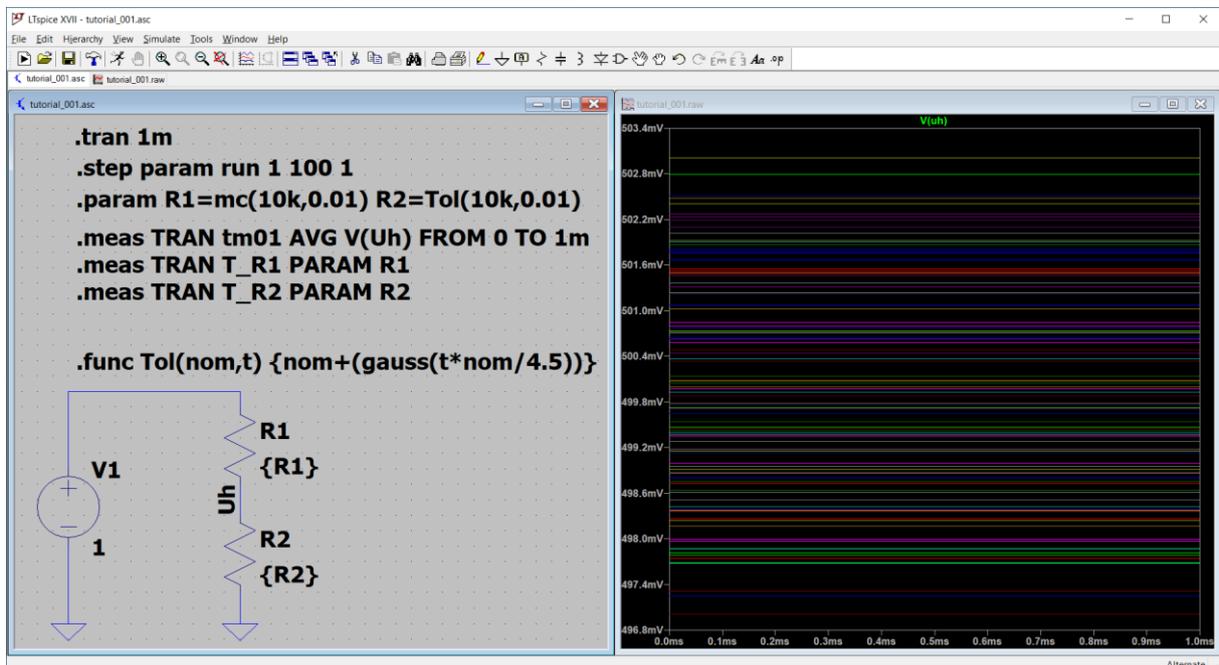
Für ItSpice gibt es Anleitungen eine DOE Matrix selber per Funktionen zu erstellen, ähnlich der unechten Worst Case Analyse. Der Aufwand in ItSpice ist aber relativ groß. Das Ergebnis sind die Design-Eckpunkte, damit kann für das Designverhalten eine mathematische Übertragungsfunktion erstellt werden. Wozu man das benötigt, vielleicht hat der ein oder andere schon mal einen Optimizer gesehen, also ein Software der man Zielwerte vorgibt und die die Eingangsparameter solange verändert bis die Zielwerte erreicht sind. Im Hintergrund arbeiten genau diese mathematischen Übertragungsfunktionen. Dazu ist der Aufwand in ItSpice aber zu groß.

Erste Schritte statistische Simulation

Ich möchte in diesem ersten Tutorial gegenüberstellen was für ItSpice oft beschrieben wird zum Thema statistischer Simulation und wie man mit Hilfe von Excel mehr Informationen aus den Ergebnissen herleiten kann.

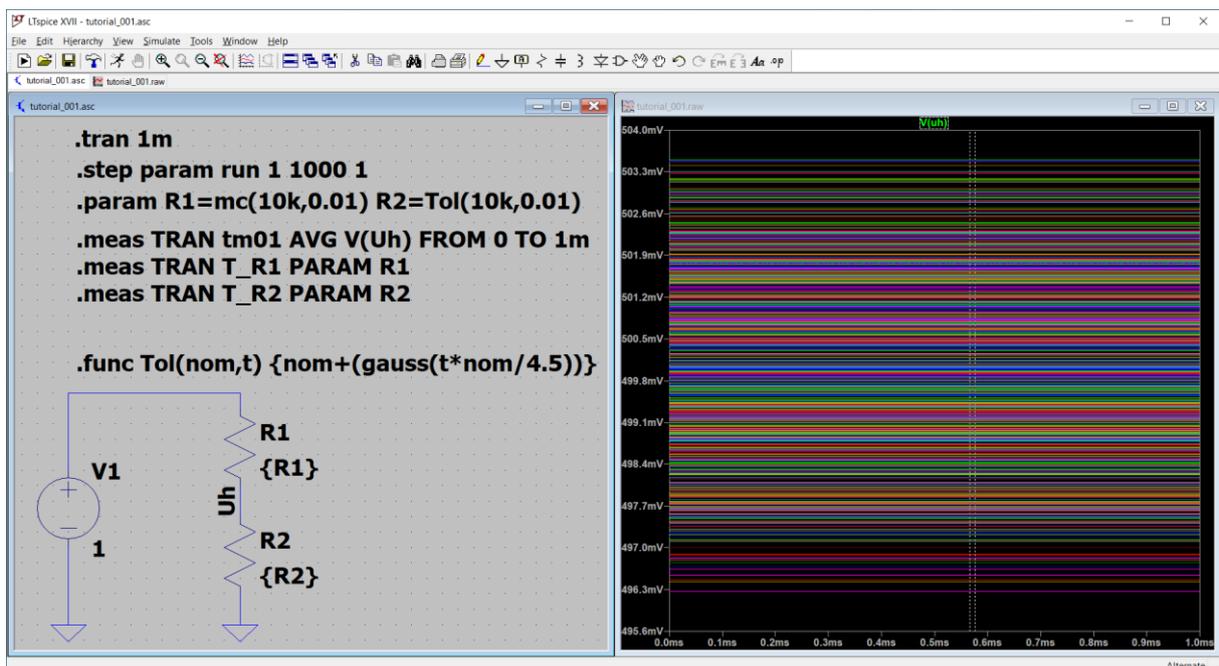
Dazu gibt es ein interessante Excel-Funktionen die uns dies stark erleichtern, auf die ich eingehen werde.

Als erstes simulieren wir eine Schaltung, wo wir wahrscheinlich alle das Ergebnis mehr oder weniger gut abschätzen können, nämlich einen ohmschen Spannungsteiler der eine Gleichspannung halbiert, wer jetzt denkt wie langweilig, es wird auch noch anspruchsvoller, versprochen.



Das ist das erste Beispiel, der Bereich der Ausgangsspannung geht von 497mV bis 503mV, also ein Bereich von $\pm 3\text{mV}$ bei einem Mittelwertwert den wir hier erwarten würden von 0.5V, sind das 0.6% Spannungsvariation.

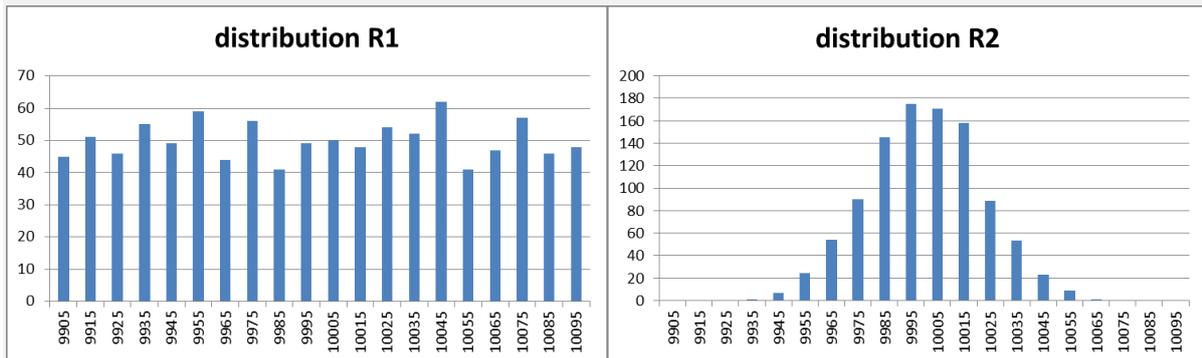
Die Widerstandswerte für R1 habe ich mit der LtSpice Funktion **mc(10k,0.01)** um $\pm 1\%$ variiert, für R2 habe ich eine eigene Funktion definiert Tol(10k,0.01), diese soll das gleiche machen also auch $\pm 1\%$. Das Ergebnis ist nur 0.6%, das liegt daran, dass LtSpice nur 100 Simulationdurchläufe gerechnet hat ind diesem Beispiel, die gleiche Schaltung mit den gleichen Toleranzen und 1000 Simulationdurchläufen sieht so aus



In diesem Ergebnis sehen wir 496.3mV bis 503.5mV, also ein Bereich von $\pm 3.6\text{mV}$, also ein Steigerung um 20%, was wäre bei 1 Million Durchläufen, unrealistisch, nein, in der Elektronikfertigung werden solche Stückzahlen gebaut und da will man wissen wo die Grenzen liegen.

Aber damit man bei komplexen Schaltungen die mehrere Minuten Simulationszeit benötigen nicht 10 Jahre simulieren muss, dafür gibt es in Excel Funktionen die wir nutzen werden.

Anm.: Nochmal zurück zu den Bauteiltoleranzen, ich habe die mc Funktion und die Tol Funktion benutzt. Innerhalb der Tol Funktion wird die ItSpice Funktion Gauss verwendet. Der Unterschied ist die Verteilung der Werte.



An der Y-Achse ist die Anzahl aufgetragen und an der X-Achse die Wert, man sieht bei R1 das wir für alle Werte eine gleiche Anzahl haben, so wie es in der ItSpice Hilfe beschrieben ist „uniform distribution“. Für R2 haben wir eine höhere Anzahl in der Mitte und zu den Rändern wird es deutlich weniger, das ist die Gaus (Dichte) Verteilung. Im Diagramm ist festzustellen, das R2 nur einen Variationsbereich von 120 Ohm hat während R1 einen Bereich von 200 Ohm aufweist. Hätten wir 1 Million Durchläufe gemacht hätten wir auch einen Bereich von 200 erhalten. Viele Toleranzen sind so definiert, das bei 1 Million Teile nur 3 Teile außerhalb der Toleranz sind, das entspricht 4.5 Sigma, 4.5 ist auch der Wert der in meiner Tol Funktion steht.

Die meisten Toleranzen entsprechen einer Gauss-Verteilung. Aber es gibt auch ein paar Fälle wo die gleichmäßige Verteilung zutreffend ist, zum Beispiel bei kalibrierten Bauteilen, sobald der Kalibrierprozess im Zielfenster ist, ist dieser fertig, die gleichmäßige Verteilung ist die Folge davon.

Welche Auswirkung das im Ergebnis hat betrachten wir jetzt indem wir im ersten Durchlauf beide Widerstände mit einer gleichmäßigen Verteilung

Im ersten Durchlauf mit **gleichmäßiger Verteilung** ist der Bereich von **495.9mV bis 504.9mV also 9mV oder $\pm 4.5\text{mV}$**

Im zweiten Durchlauf mit **Gauss Verteilung** ist der Bereich von **498.8mV bis 501.8mV also nur 3mV oder $\pm 1.5\text{mV}$**

Man sieht, sich über die Art der zu benutzenden Verteilung Gedanken zu machen ist wesentlich wenn man korrekte Ergebnisse erhalten möchte.

Wer das Beispiel nachrechnet wird evtl. abweichende Ergebnisse erhalten, die Zufallszahlen sind immer neu generiert.

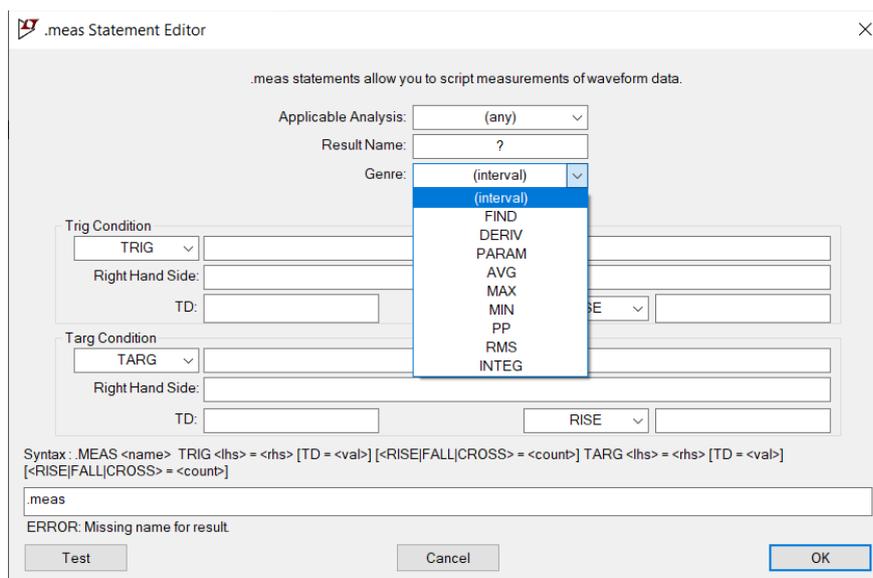
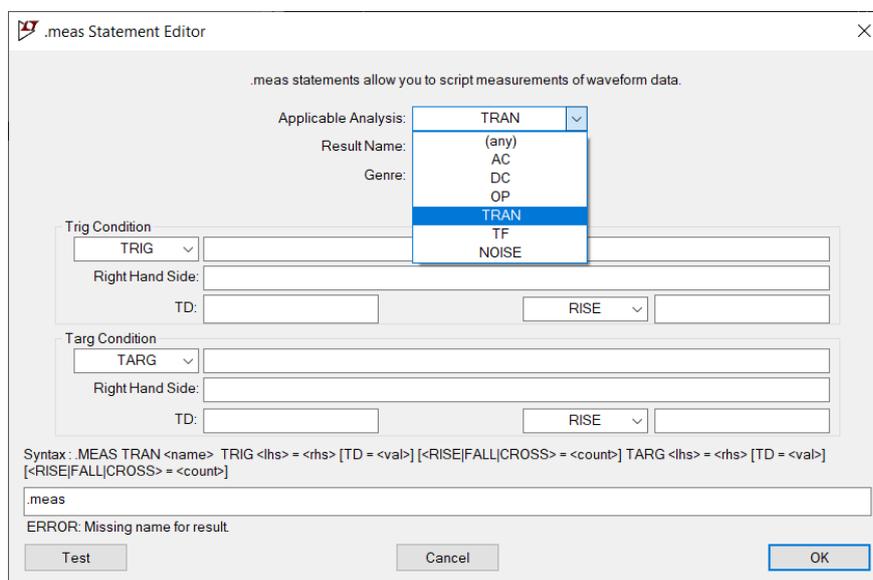
Bewertung der Ergebnisse

Aber wie kann ich jetzt das Ergebnis von 100 Durchläufen bewerten, dazu muss man das Ergebnis in Excel kopieren.

Da hilft uns der `.meas` Befehl weiter

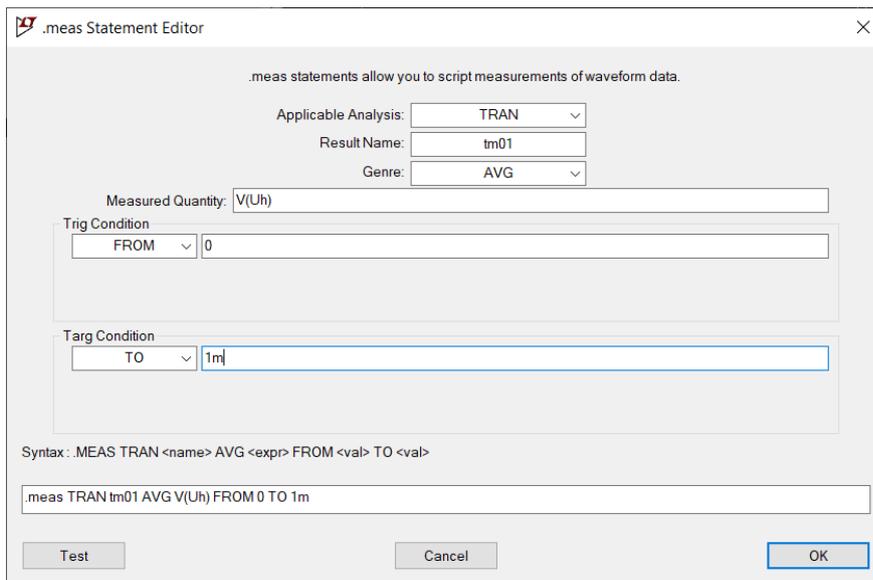
```
.meas TRAN tm01 AVG V(Uh) FROM 0 TO 1m  
.meas TRAN T_R1 PARAM R1  
.meas TRAN T_R2 PARAM R2
```

Wenn man die Messungen erstellen will muss man nicht die ganze Syntax kennen, man platziert mittel `.op` den `.meas` Befehl auf den Schematic und clickt dann mit der rechten Maustaste darauf und erhält das folgende Menü



Für Trigger Condition bleiben wir am Anfang bei „From“ und „To“ und geben die Zeit vor. Den Trigger Conditions kann man ein eigenes Tutorial widmen. Wenn ich welche benutze werde ich die aber erklären.

Für die Measure Zeile **.meas TRAN tm01 AVG V(Uh) From 0 to 1m** wurde das Formular so bedatet.



Die Analyse wo der measure Befehl benutzt wird ist die Transienten Analyse

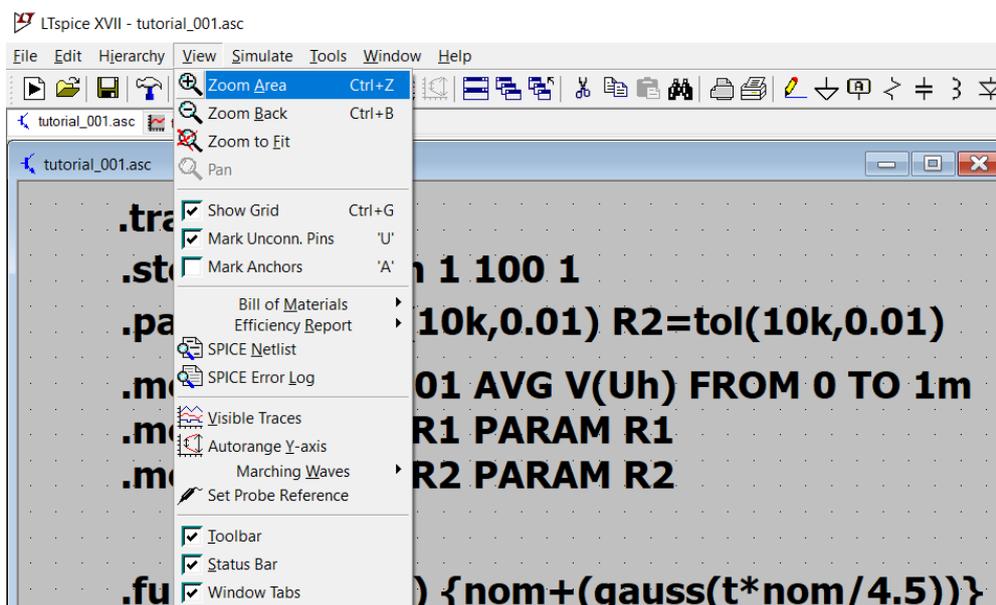
Der von mir definierte Name ist **tm01** (da ist jeder frei, für mich ist T für Transient M für Messung und dann durchnummeriert)

Genre ist der Typ der Messung, ich möchte den Mittelwert, **AVG** steht für average.

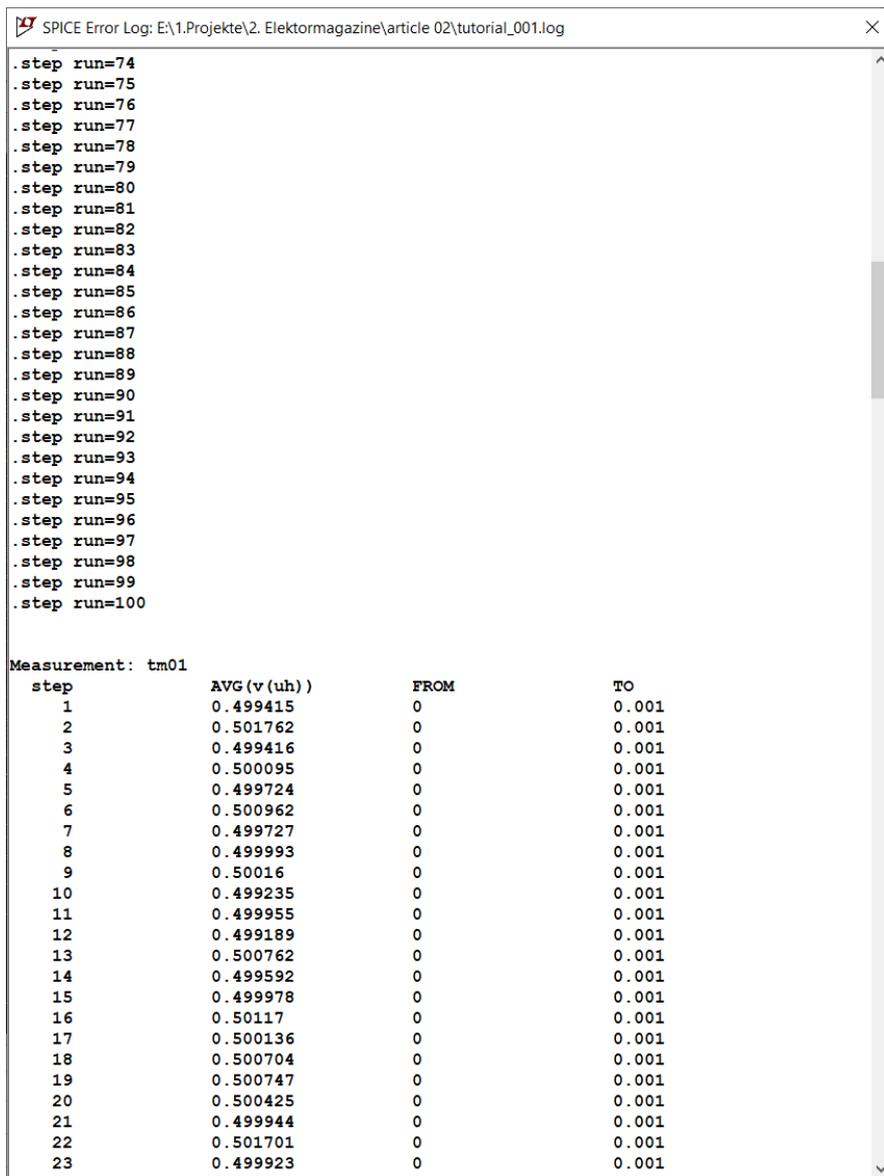
Die gemessene Größe ist die Spannung im Schaltplan am Netz Uh, somit also V(Uh). Es ist wichtig die Netze zu benennen, wenn man bei komplexen Schaltungen was hinzufügt oder löscht ändern sich die automatisch vergebenen Netznamen. Außerdem kann man sinnvolle Abkürzungen verwenden so dass man es auch schneller wiederfindet im Plan.

Für die Trigger Bedingung nehme ich in diesem Fall **from=0** Start der Simulation bin **to=1m** Ende der Simulation.

Nachdem die 100 Durchläufe simuliert sind öffnen wir das Logfile, ->View->SPICE Error Log



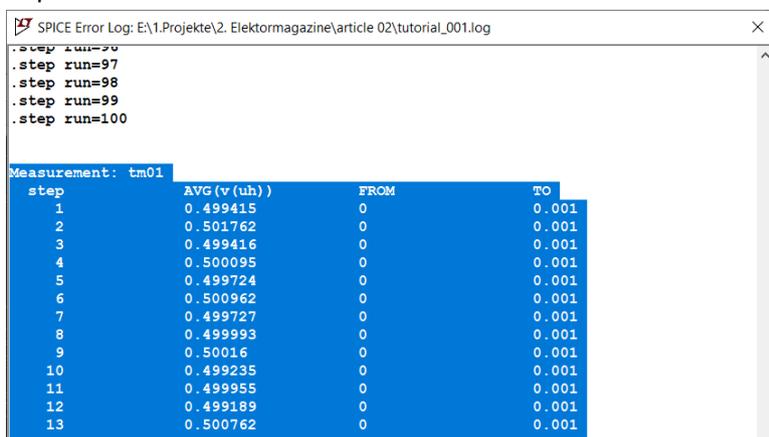
Hier finden wir die Ergebnisse die uns der Measure Befehl liefert



```
SPICE Error Log: E:\1.Projekte\2. Elektormagazine\article 02\tutorial_001.log
.step run=74
.step run=75
.step run=76
.step run=77
.step run=78
.step run=79
.step run=80
.step run=81
.step run=82
.step run=83
.step run=84
.step run=85
.step run=86
.step run=87
.step run=88
.step run=89
.step run=90
.step run=91
.step run=92
.step run=93
.step run=94
.step run=95
.step run=96
.step run=97
.step run=98
.step run=99
.step run=100

Measurement: tm01
step      AVG (v (uh) )    FROM      TO
1         0.499415         0         0.001
2         0.501762         0         0.001
3         0.499416         0         0.001
4         0.500095         0         0.001
5         0.499724         0         0.001
6         0.500962         0         0.001
7         0.499727         0         0.001
8         0.499993         0         0.001
9         0.50016         0         0.001
10        0.499235         0         0.001
11        0.499955         0         0.001
12        0.499189         0         0.001
13        0.500762         0         0.001
14        0.499592         0         0.001
15        0.499978         0         0.001
16        0.50117         0         0.001
17        0.500136         0         0.001
18        0.500704         0         0.001
19        0.500747         0         0.001
20        0.500425         0         0.001
21        0.499944         0         0.001
22        0.501701         0         0.001
23        0.499923         0         0.001
```

Für das erste Ergebnis tm01 sehen wir hier die Schritte 1 bis 100 in der Spalte AVG(v(uh)), das ist das was wir benötigen. Den Bereich für step 1 bis 100 markieren und mit STRG+C in die Zwischenablage kopieren.

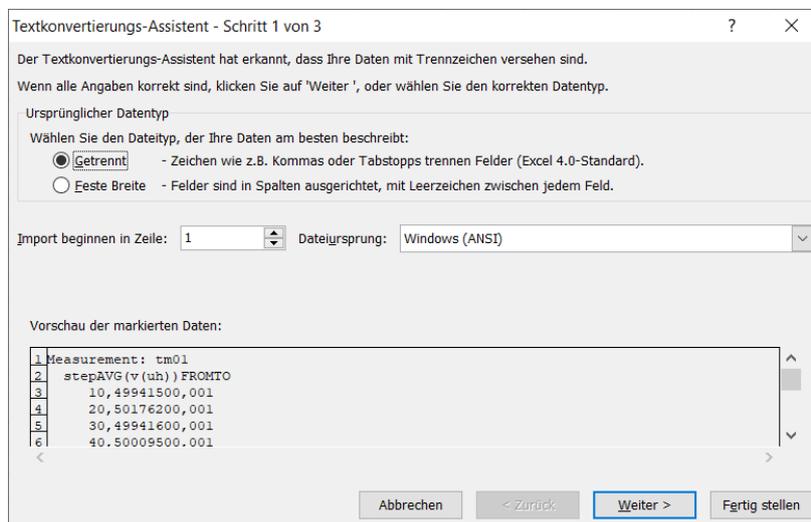
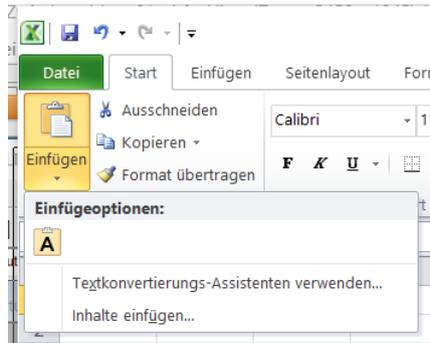


```
SPICE Error Log: E:\1.Projekte\2. Elektormagazine\article 02\tutorial_001.log
.step run=96
.step run=97
.step run=98
.step run=99
.step run=100

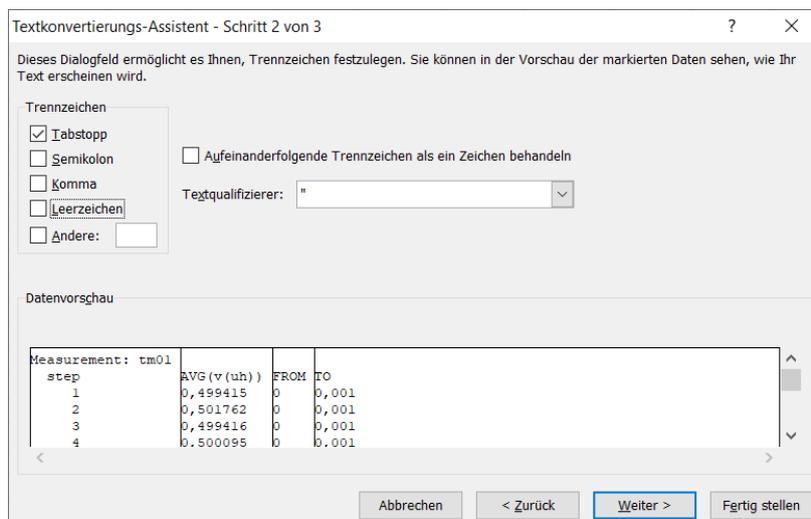
Measurement: tm01
step      AVG (v (uh) )    FROM      TO
1         0.499415         0         0.001
2         0.501762         0         0.001
3         0.499416         0         0.001
4         0.500095         0         0.001
5         0.499724         0         0.001
6         0.500962         0         0.001
7         0.499727         0         0.001
8         0.499993         0         0.001
9         0.50016         0         0.001
10        0.499235         0         0.001
11        0.499955         0         0.001
12        0.499189         0         0.001
13        0.500762         0         0.001
14        0.499592         0         0.001
```

Es ist hilfreich einen Zwischenschritt einzufügen und zuerst die Daten in einen Texteditor wie Notepad++ oder Textpad einzufügen und die Zeichen Punkt durch Komma zu ersetzen, zumindest bei der deutschen Systemeinstellung.

Dann vom Texteditor wieder in die Zwischenablage kopieren und in Excel mit dem Textkonvertierungsassistenten einfügen



Hier die Voreinstellung belassen und weiter



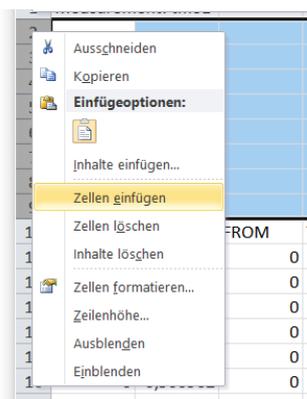
Hier die Einstellung für Trennzeichen auf Tabstopp ändern und auf fertigstellen.

In Excel sollte es dann so aussehen

	A	B	C	D	E
1	Measurement: tm01				
2	step	AVG(v(uh))	FROM	TO	
3	1	0,499415	0	0,001	
4	2	0,501762	0	0,001	
5	3	0,499416	0	0,001	
6	4	0,500095	0	0,001	
7	5	0,499724	0	0,001	
8	6	0,500962	0	0,001	
9	7	0,499727	0	0,001	
10	8	0,499993	0	0,001	
11	9	0,50016	0	0,001	
12	10	0,499235	0	0,001	
13	11	0,499955	0	0,001	
14	12	0,499189	0	0,001	
15	13	0,500762	0	0,001	
16	14	0,499592	0	0,001	
17	15	0,499978	0	0,001	
18	16	0,50117	0	0,001	
19	17	0,500136	0	0,001	
20	18	0,500704	0	0,001	
21	19	0,500747	0	0,001	
22	20	0,500435	0	0,001	

Step geht bis 100

Wenn wir die Daten so vorliegen haben wollen wir die ersten Excel Befehle anwenden, zuerst aber machen wir uns ein wenig Platz, Markieren die Zeilen 2 bis 9 und wählen mit rechten Mausklick auf die Zeilennummerierung „Zellen einfügen“

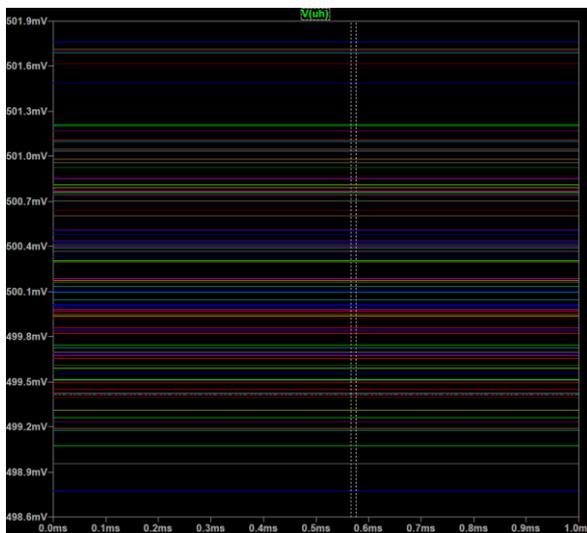


Jetzt benutzen wir die erste Excel Funktion

	A	B	C	D
1	Measurement: tm01			
2				
3		=mittelwert(B11:B110)		
4				
5				
6				
7				
8				
9				
10	step	AVG(v(uh))	FROM	TO
11	1	0,499415	0	0,001
12	2	0,501762	0	0,001
13	3	0,499416	0	0,001
14	4	0,500095	0	0,001

Das Ergebnis ist bei mir 0.50022, das ist wie erwartet, bei 100 zufälligen Durchläufen ganz gut, das wird aber auch mal mehr oder weniger sein.

Den Mittelwert hätten wir in der grafischen Darstellung nicht exakt ermitteln können, nur abschätzen.



Als zweite Excel Funktion benutzen wir die Standardabweichung für eine Stichprobe, das Ergebnis ist 0,000685, da es eine Spannung ist die wir messen können wir auch 0.685mV schreiben

	A	B	C	D	E
1	Measurement: tm01				
2					
3		mittelwert	0,50022		
4		staabw	=STABW.S(B11:B110)		
5			STABW.S(Zahl1; [Zahl2]; ...)		
6					
7					
8					
9					
10	step	AVG(v(uh))	FROM	TO	
11	1	0,499415	107Z x 1S	0,001	
12	2	0,501762	0	0,001	
13	3	0,499416	0	0,001	
14	4	0,500095	0	0,001	

Die Standardabweichung ist eine Kenngröße der Gauss Kurve. Die Gausskurve ist definiert das innerhalb ± 1 Standardabweichung 68.3% aller Werte liegen, innerhalb ± 2 Standardabweichungen 95.5% usw. wie in der Tabelle aufgezeigt.

sigma	number
0,75	54,7%
1	68,3%
1,25	78,9%
1,5	86,6%
1,75	92,0%
2	95,4%
2,25	97,6%
2,5	98,8%
2,75	99,4%
3	99,73%
3,25	99,88%
3,5	99,95%
3,75	99,98%
4	99,99%
4,25	99,998%
4,5	99,999%
4,75	99,9998%
5	99,9999%

Die Tabelle zeigt wieviel Prozent der Werte innerhalb n x Standardabweichung liegt, für n ist der Ausdruck Sigma Level gebräuchlich.

Anm.: Die Funktion in Excel für „number“ zu ermitteln lautet $=1-2*(1-STANDNORMVERT(sigma))$

Aber zurück zu unserem Ergebnis, unser **Standardabweichung ist 0.6885mV**, wenn wir den Anspruch haben das 99.9990% der Teile innerhalb unsere ermittelten Toleranz liegt (*4.5 Sigma ist Industrie Standard*) dann müssen wir einen Sigma Level von 4.5 auswählen.

Unsere 0.685mV einfache Standardabweichung wird dann mit 4.5 multipliziert, daraus ergibt sich $\pm 3.083mV$, dann ist gewährleistet das 99.9990% aller gefertigten Spannungsteiler eine Toleranz kleiner gleich $\pm 3.083mV$ aufweisen bzw. eine Spannungstoleranz für V(Uh) von 0.6%.

Wir erinnern uns dass wir aus dem Diagramm **498.8mV bis 501.8mV also nur 3mV oder $\pm 1.5mV$** abgelesen haben für 100 Durchläufe. Wenn wir 1 Million Durchläufe simuliert hätten, dann könnten wir die $\pm 3mV$ aus der Excel Berechnung bestätigen. Aber das ist nicht notwendig, wir können der Statistik da schon trauen.

Wir können auch noch mal andersherum rechnen, wir hatten aus dem Diagramm $\pm 1.5mV$ abgelesen, wenn wir diesen Wert durch die Standardabweichung 0.685mV dividieren erhalten wir einen Wert, den man als Sigma Level bezeichnet $= 1.5/0.685 = 2.18$. Schauen wir in die Tabelle dann sehen wir das wir zwischen 2 entsprechend 95.4% und 2.25 entsprechend 97.6% liegen mit 2.18 wären wir ca. bei 97%.

Das bedeutet das 3% der produzierten Schaltungen eine größere Abweichung haben als $\pm 1.5mV$. Hätten wir jetzt für eine Elektronik Serienproduktion gesagt, wir haben $\pm 1.5mV$ Toleranz und diesen Wert unserem Kunden versprochen, dann hätten wir 3% der Teile reklamiert bekommen, ein KO Kriterium.

Anm.: zum ersten Beispiel, beide Widerstände hatten 1% Toleranz und die Spannung hatte nur 0.6% Toleranz, warum? Nur in der ungünstigsten Konstellation also R1+1% und R2-1% oder andersherum, ergibt sich ein Spannungsfehler von 1% beim Teiler 1:1

Aber Toleranzen addieren sich nicht arithmetisch, als nicht Tol1 + Tol2 usw,

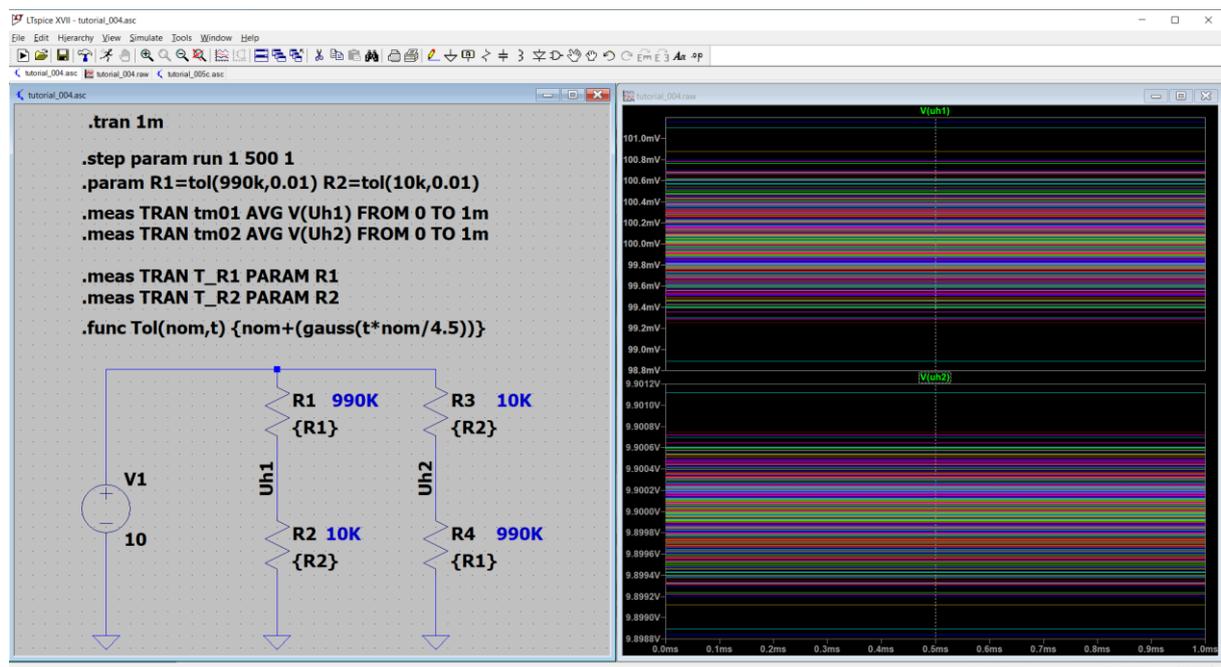
Toleranzen werden geometrisch addiert, also $\sqrt{Tol1^2 + Tol2^2 + \dots}$. Daraus würde sich theoretisch bei jeweils 1% dann ein Fehler von 0.7% ergeben, die 0.6% sind den nur 100 Simulationen durchläufen geschuldet und nicht optimaler Zufallszahlen, mit einer geringen gleichläufigen Abhängigkeit zwischen R1 und R2 und etwas zu geringer Streuung.

Betrachtung komplexerer Schaltungen und Analyse

Nochmal zwei weitere Spannungsteiler, das ist für die Analyse ein ausgezeichnetes Beispiel weil diese unterschätzt werden und sich gerade deshalb Überraschungseffekte einstellen.

Im ersten Beispiel hatten wir ein Spannungsteiler 1:1 und die Toleranz der Ausgangsspannung war mit 0.6% kleiner als die Toleranz der Widerstände.

Jetzt simulieren wir einen Spannungsteiler 1:99 und 99:1 indem wir die gleichen Widerstände vertauschen



Ich habe für R3 den gleichen Wert wie für R2 verwendet und für R4 den Wert von R1, damit ist ein exakter vergleich gewährleistet

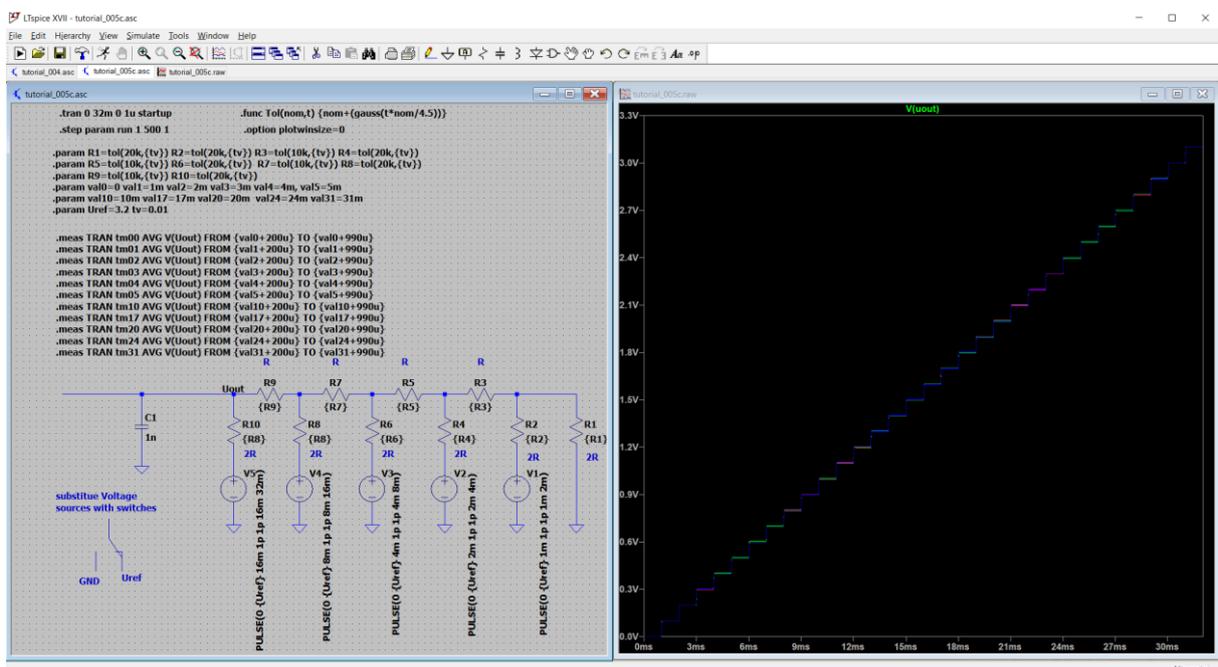
Das Ergebnis für 500 Simulationen in Excel importiert sieht folgendermaßen aus

	A	B	C	D	E
1	expected Value		100mV		9900mV
2					
3	average [mV]		100,0		9900,0
4	4.5 x stdev [mV]		1,347		1,347
5					
6	voltage tolerance		1,347%		0,014%
7					
8					
9					
10	Measurement	tm01			tm02
11	step	AVG(v(uh1))			AVG(v(uh2))
12	1	0,0997684			9,90023
13	2	0,1007			9,8993
14	3	0,099769			9,90023
15	4	0,100038			9,89996
16	5	0,0998909			9,90011
17	6	0,100382			9,89962
18	7	0,099892			9,90011
19	8	0,0999971			9,9
20	9	0,100064			9,89994
21	10	0,0996973			9,9003
22	11	0,099823			9,90002
23	12	0,0996795			9,90032
24	13	0,100302			9,8997
25	14	0,0998385			9,90016
26	15	0,0999912			9,90001

Die Toleranz (4.5 x Stddev) ist für beide mit 1,347mV für beide Teiler gleich, das muss auch so sein, für die größere Spannung ist aber der relative Fehler mit 0,014% kleiner als für die kleine Spannung mit 1,347% , das ist auch leicht nachvollziehbar weil die Spannung ca. Faktor 100 größer ist.

Der theoretische Fehler für 1:99 ist wurzel(2) weil sich die Fehler geometrisch addieren und bei diesen Teiler-Verhältnissen beiden Widerstandstoleranzen vollen Einfluss auf das Ergebnis haben. Der große Wert bestimmt maßgeblich den Strom während der kleine Wert über den die Spannungs abgegriffen wird die Spannung die aus dem Strom resultiert bestimmt. Bei Einem Teiler 1:1 ist der Einfluss eines Widerstands auf den Strom quasi halbiert.

Im nächsten Beispiel simulieren wir ein R2R Netzwerk



Die Referenz-Spannung beträgt 3.2V und wir machen 32 Schritte von 0 bis 3.1V. Man kann auf diese Weise einen manuellen DA Wandler aufbauen der mit Schalter eine genaue Spannung einstellen kann, sofern wir eine präzise Referenz haben oder mit der wir Spannungsteiler Verhältnisse überprüfen können.

Die Spannungsquellen repräsentieren ein digitales Bitmuster mit V5 dem MSB und V1 dem LSB, gezählt wird beginnend bei 0 und geht bis 31 (digital).

Das Bitmuster wechselt im 1ms Takt, da auch die numerische Simulation „mathematische Störungen“ verursacht habe ich am Ausgang einen Kondensator platziert, der sorgt für entsprechend kurze Simulationsschrittweiten beim Bitmusterwechsel.

Wegen dieser aus dem Kondensator resultierenden Zeitkonstanten von 10us, wegen $R_i=10k$ und $C=1n$, wird das jeweilige Treppenplateau erst nach 20 Zeitkonstanten beginnen ausgewertet mit dem measure Befehl (200us) die Verkürzung von 10us vor dem nächsten Wechsel ist eigentlich unnötig. Macht aber Sinn wenn man später auch zeitliche Toleranzen simuliert.

Beginnen nach 20 Zeitkonstanten verbleibt noch ein theoretischer Fehler für den Mittelwert von 0,0000000041%, damit ist der Einfluss geringer als die Simulationsgenauigkeit.

Die Simulationsergebnisse in Excel importiert sehen folgendermaßen aus, es wurden 500 Durchläufe simuliert.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
1	expected Value		0mV	100mV	200mV	300mV	400mV	500mV	1000mV	1700mV	2000mV	2400mV	3100mV										
3	average [mV]	0	100,0	200,0	300,0	400,0	500,0	1000,0	1700,1	2000,0	2400,0	3100,0											
4	4.5 x stdev [mV]	0	0,316	0,556	0,765	1,100	1,257	0,730	1,415	0,995	1,768	0,297											
6	voltage tolerance	0%	0,32%	0,28%	0,26%	0,28%	0,25%	0,07%	0,08%	0,05%	0,07%	0,01%											
9	table results in Volt																						
10	Measurement:	tm00	tm01	tm02	tm03	tm04	tm05	tm10	tm17	tm20	tm24	tm31											
11	step	AVG(v(uout))																					
12	1	0	0,0999984	0,20034	0,300338	0,397878	0,497876	1,00081	1,70128	1,99916	2,40175	3,09996											
13	2	0	0,0997819	0,199714	0,299496	0,400422	0,500204	0,999596	1,70014	2,00078	2,40024	3,10015											
14	3	0	0,0999065	0,199478	0,299385	0,400389	0,500295	0,998144	1,70172	2,00221	2,40048	3,10026											
15	4	0	0,100203	0,200242	0,300445	0,400521	0,500725	1,00081	1,69817	1,99848	2,39853	3,0995											
16	5	0	0,100076	0,200001	0,300077	0,398161	0,498237	0,999812	1,70183	1,99992	2,40157	3,09981											
17	6	0	0,100144	0,199636	0,29978	0,400553	0,500697	1,00038	1,69862	1,99902	2,39922	3,09955											
18	7	0	0,100464	0,200472	0,300936	0,401224	0,501688	0,999352	1,69878	1,99954	2,3972	3,09936											
19	8	0	0,0995348	0,200118	0,299653	0,400371	0,499906	1,00043	1,69922	2,00006	2,4	3,10002											
20	9	0	0,100504	0,200566	0,30107	0,40029	0,500794	1,00066	1,69841	1,9982	2,398	3,09936											
21	10	0	0,0994609	0,200365	0,299826	0,398268	0,497729	1,00102	1,70077	1,99958	2,40196	3,10006											
22	11	0	0,100114	0,199941	0,300055	0,398074	0,498188	0,999819	1,70241	2,00037	2,40217	3,1003											
23	12	0	0,100448	0,200363	0,300811	0,401309	0,501757	1,00069	1,69724	1,9981	2,39712	3,09924											
24	13	0	0,100061	0,199851	0,299912	0,398833	0,498894	1,00004	1,70091	1,99968	2,40104	3,09978											
25	14	0	0,100015	0,200276	0,300291	0,398817	0,498832	1,00025	1,70104	1,99984	2,401	3,10011											
26	15	0	0,100000	0,200000	0,300000	0,398833	0,498833	1,00000	1,70000	1,99999	2,40000	3,09999											

Hier sehen wir das gleich wie für die beiden Spannungsteiler 1:99 und 99:1, bei kleinen Spannungen ist der relative Fehler größer als bei den großen Spannungen. Das ist nach dem vorigen Beispiel aber nicht mehr ganz so überraschend.

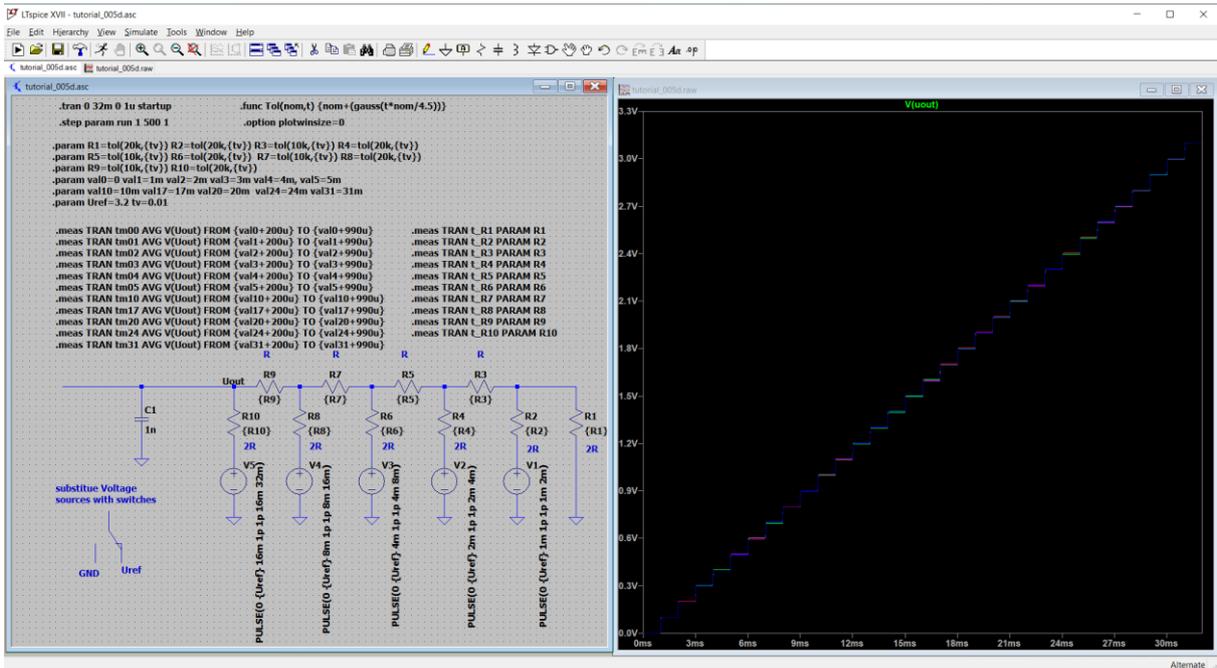
Die spannende Frage wäre aber, welche Widerstände haben auf die kleinen Wert den signifikanten Einfluss oder muss ich für alle Widerstände die Toleranz entsprechend einschränken?

Detaillierte Analyse der Toleranzursachen

Um diese Frage zu beantworten bieten keine mir bekannte Spice Simulationswerkzeuge geeignete Werkzeuge an um dies schnell und einfach zu untersuchen. Es gibt aber mathematische Methoden die einen dabei unterstützen können und einem schnell einen Überblick verschaffen welche Bauteiltoleranzen kritisch sind und welche nicht.

Das Stichwort ist Regressionsanalyse, keine Angst, Excel hat da eine passende Funktion im Angebot, wir müssen nur die Anwendung verstehen und wie man die Ergebnisse interpretiert, das ist nicht ganz so schwierig wie es sich anhört. Insbesondere wenn man die hier gezeigten Vorlagen verwendet.

Aber wenn man nur LtSpice und Excel verwendet so wie bisher, ist es noch etwas mehr Arbeit mit dem kopieren der Daten in das Excel Tabellenblatt. Man benötigt alle Widerstandswerte und Ergebnisse.

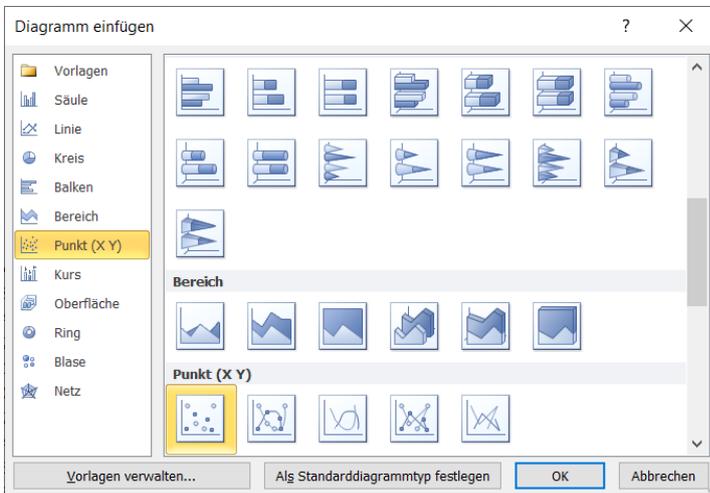


Der gleiche Plan mit dem zusätzlichen measure Befehlen um die Widerstandswerte in das Logfile zu schreiben.

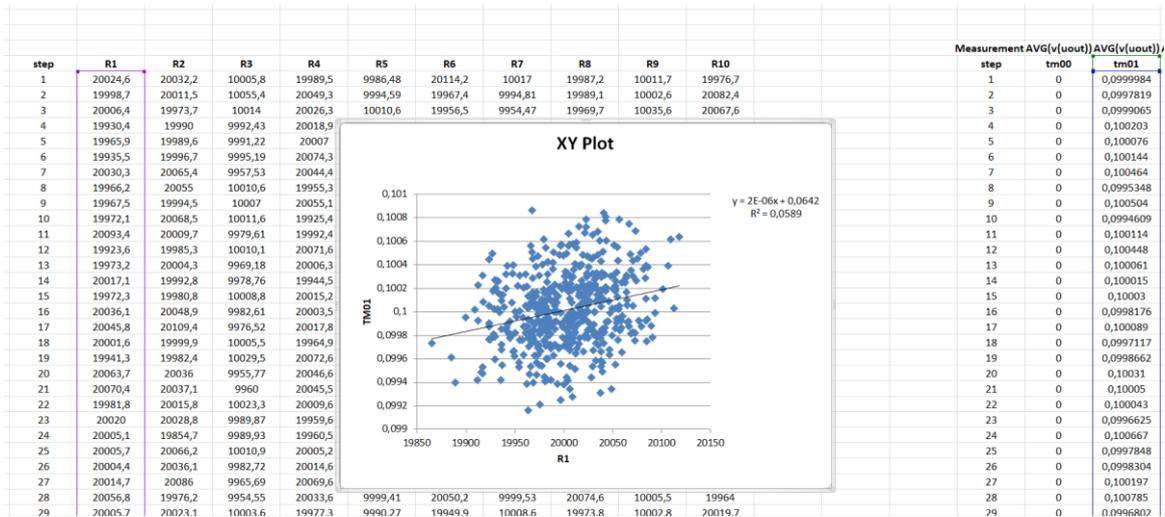
Nach 20 Minuten kopieren und schieben sieht das Logfile in Excel jetzt so aus, auch hier wieder 500 Simulationen gerechnet, step von 1 bis 500.

step	r1	r2	r3	r4	r5	r6	r7	r8	r9	r10	tm00	tm01	tm02	tm03	tm04	tm05	tm10	tm17	tm20	tm24	tm31	
1	20024.6	20012.2	10005.4	19989.3	9986.48	20114.2	10017	19982.2	10011.7	19976.7	0	0.0997819	0.199714	0.299496	0.408422	0.508204	0.999596	1.70014	2.00078	2.40024	3.10015	
2	19998.7	20011.5	10005.4	20049.3	9994.59	19967.4	9994.81	19989.1	10002.6	20082.4	2	0	0.0997805	0.199748	0.299486	0.408399	0.508205	0.998144	1.70172	2.00221	2.40048	3.10026
3	20004.4	19977.7	10014	20026.3	10010.9	19994.5	9994.47	19989.7	10016.8	20067.6	4	0	0.100203	0.200202	0.300445	0.408251	0.508725	1.00081	1.69817	1.99848	2.39803	3.0995
4	19990.4	19990	9991.41	20018.9	10008.3	20027.5	9974.66	20005.6	9975.35	20085.5	6	0	0.100076	0.200001	0.300077	0.398161	0.498237	0.998812	1.70183	1.99992	2.40157	3.09981
5	19965.9	19989.6	9991.22	20007	9966.6	20069.2	9979.83	19953.5	10012	20019.5	8	0	0.100144	0.199938	0.29978	0.408553	0.508687	1.00038	1.69862	1.99923	2.39922	3.09955
6	19935.3	19996.7	9991.19	20074.3	10017.6	20014.5	10011.3	20013.6	9978.8	19982.3	10	0	0.100064	0.200472	0.300936	0.401224	0.505888	0.99952	1.69878	1.99954	2.3972	3.09936
7	20030.3	20065.4	9997.53	20044.4	9989.36	20011.3	9963.24	20063.4	10014.2	19997.3	12	0	0.0995348	0.200118	0.299653	0.400371	0.499906	1.00043	1.69922	2.00006	2.4	3.10002
8	19964.2	20055	10010.6	19953.3	10007.6	19985.4	10006.2	20000.4	9990.5	19975.9	14	0	0.100054	0.200046	0.300107	0.408209	0.508794	1.00066	1.69841	1.9982	2.398	3.09936
9	19967.9	19984.5	10007	20015.1	10005	20028.1	9979.3	20048	9981.63	19994.4	16	0	0.0994609	0.200385	0.299826	0.398268	0.497729	1.00102	1.70077	1.99958	2.40196	3.10006
10	19972.1	20068.5	10011.6	19925.4	9997.68	20020.6	10000.3	19945.7	9989.16	19979.2	18	0	0.100114	0.199941	0.300055	0.398074	0.498188	0.999819	1.70241	2.00037	2.40117	3.1001
11	20093.4	20000.7	9979.81	19992.4	9995.38	20075.3	9991.03	19960.1	10024.7	20015.2	20	0	0.100048	0.200033	0.300111	0.401309	0.501757	1.00069	1.69724	1.9981	2.39712	3.09934
12	19973.2	20004.3	9969.18	20006.3	9992.48	20049.9	10022.1	20000.1	10013.6	19968.3	22	0	0.100061	0.199851	0.299912	0.398833	0.498894	1.00004	1.70091	1.99968	2.40104	3.09978
13	19973.2	20004.3	9969.18	20006.3	9992.48	20049.9	10022.1	20000.1	10013.6	19968.3	24	0	0.100015	0.200276	0.300291	0.398817	0.498832	1.00025	1.70104	1.99984	2.401	3.10011
14	20017.1	19992.8	9997.76	19944.3	9997.9	20025.5	10008.8	19978.1	10030.6	20046.6	26	0	0.100015	0.200276	0.300291	0.398817	0.498832	1.00025	1.70104	1.99984	2.401	3.10011
15	20017.1	19992.8	9997.76	19944.3	9997.9	20025.5	10008.8	19978.1	10030.6	20046.6	28	0	0.100015	0.200276	0.300291	0.398817	0.498832	1.00025	1.70104	1.99984	2.401	3.10011

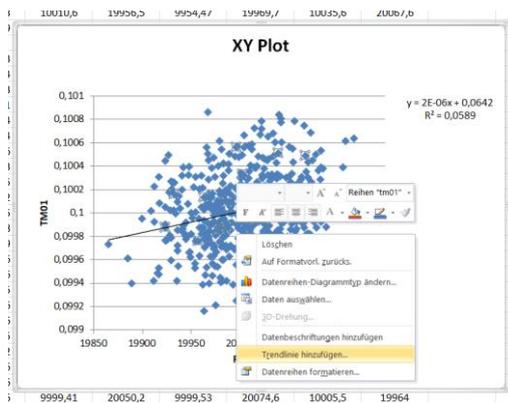
Im nächsten Schritte beginnen wir mit der Suche nach Abhängigkeiten, dazu verwenden wir das XY Diagramm



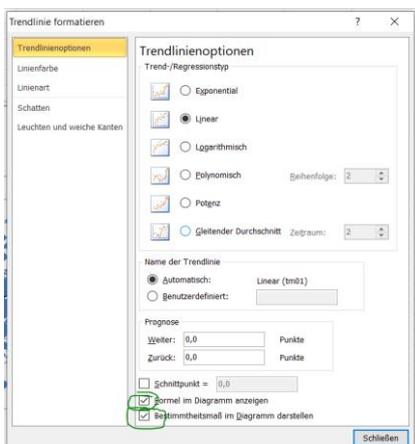
Wir wählen für die X-Achse die Eingangsgröße, also die Spalte mit den Werten für R, hier starten wir mit R1 und für die Ausgangsgröße die Ausgangs-Spannung, hier TM01



Um diesen Diagramm zu erhalten machen wir einen rechten Mausklick auf die Punkte und bekommen das Auswahlmenue welches uns eine Trendlinie anbietet.

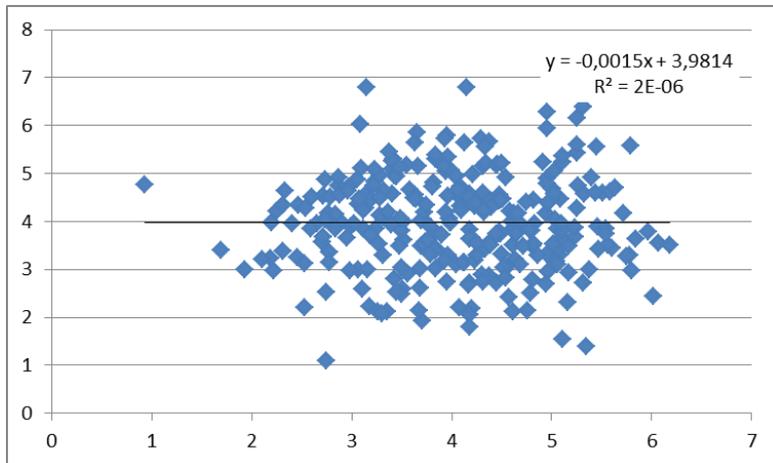


In Menu behalten wir die Vorauswahl Linear und aktivieren „Formel im Diagramm“ und „Bestimmtheitsma...“

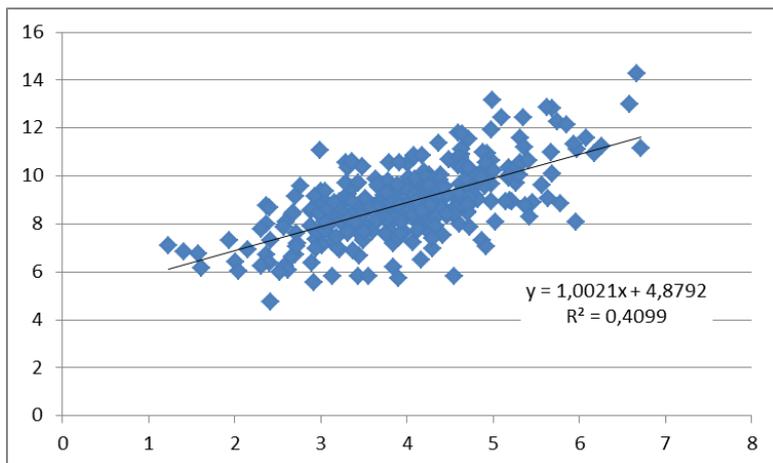


Wenn man die Punkte anklickt sehen wir im Tabellenblatt jeweils einen Rahmen um die X-Werte und Y-Werte. Wir können die Rahmen verschieben und erhalten dann die entsprechende Aktualisierung des Diagramms inclusive der Trendlinie Formel der Geradengleichung und Bestimmtheitsmaß R^2 .

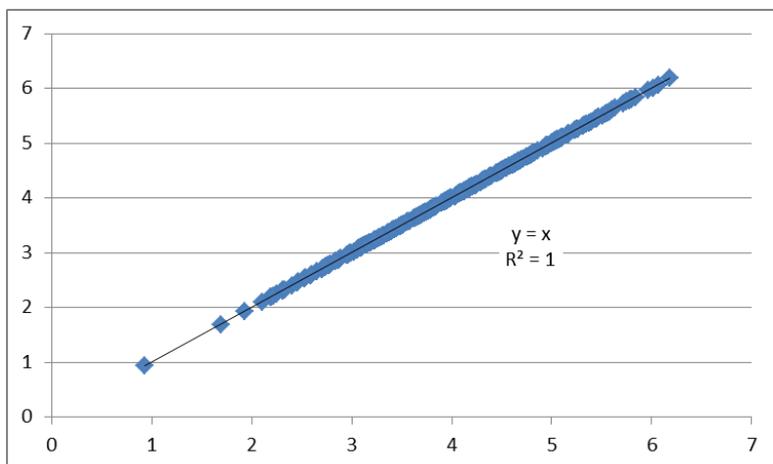
Für die Bewertung der Diagramm gilt, je runder die Punktwolken umso weniger Abhängigkeit, je mehr die Punkte auf einer Linie liegen desto größer die Abhängigkeit. Alternativ gilt für die Kennwerte der Trendlinie, kleines $R^2 < 0.05$ fast keine Abhängigkeit $R^2 > 0.2$ Abhängigkeit $R^2 = 1$ perfekte Korrelation.



Keine Abhängigkeit, $R^2=0.000002$

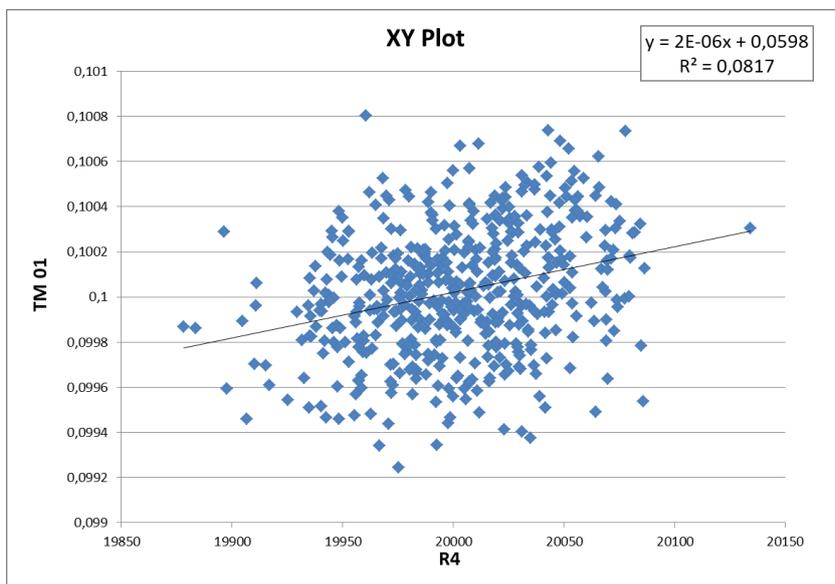
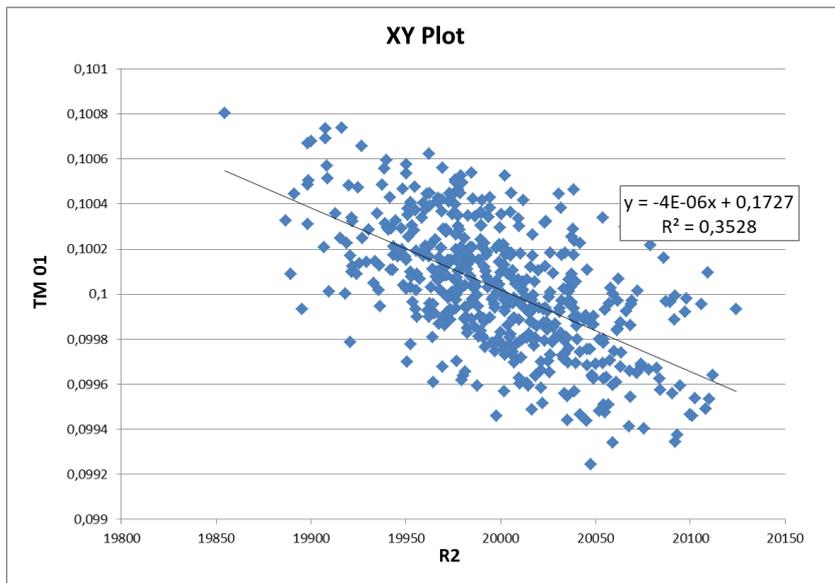
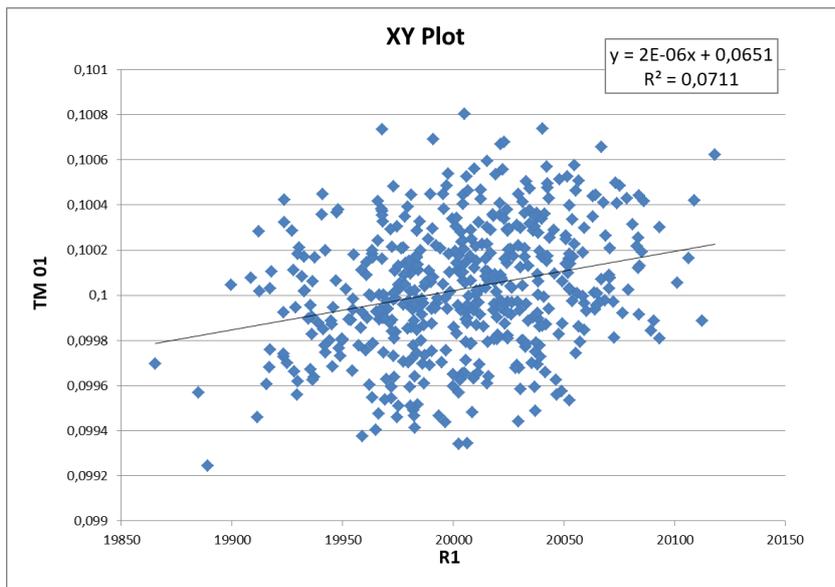


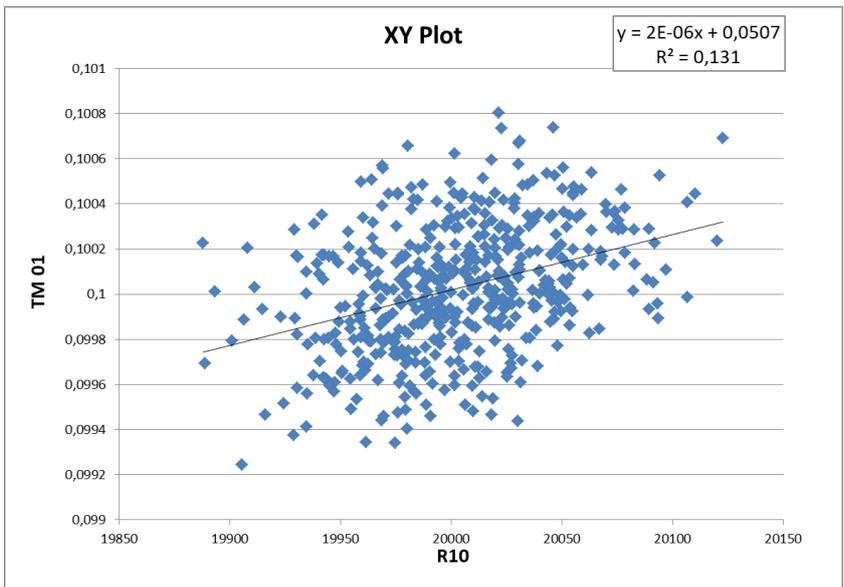
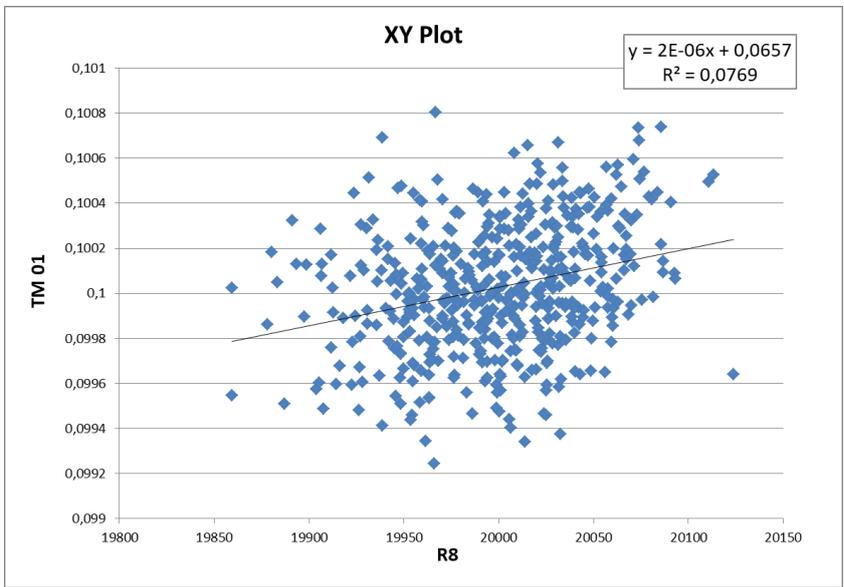
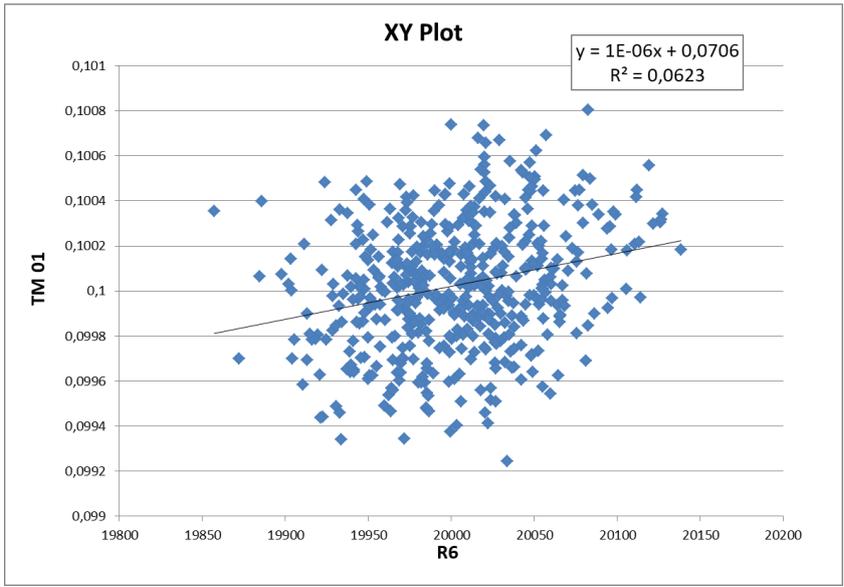
Deutliche Abhängigkeit, besonders wenn mehrer Bauteile Einfluss haben ist das schon ein signifikanter Wert



Ganz klare Abhängigkeit, bei $R^2=1$ gibt es keine weiteren Einflussparameter

Bewertung unser R2R Netzwerkes



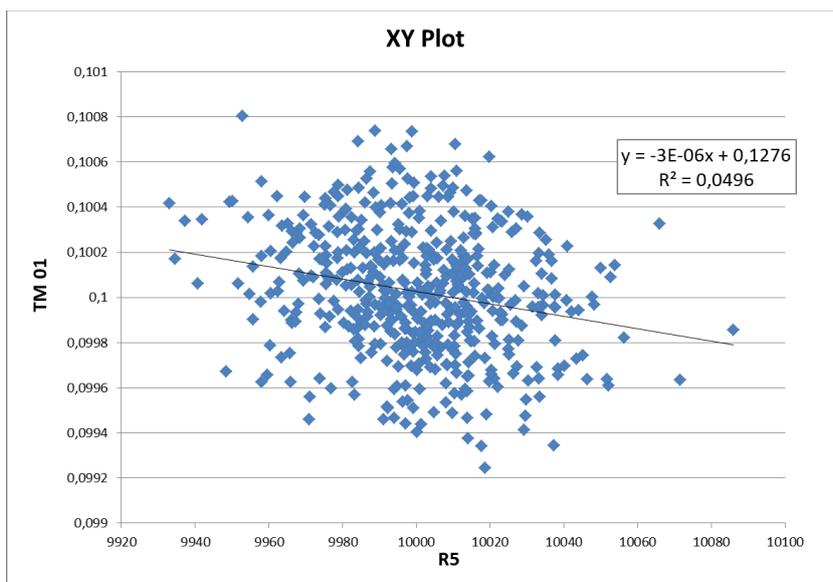
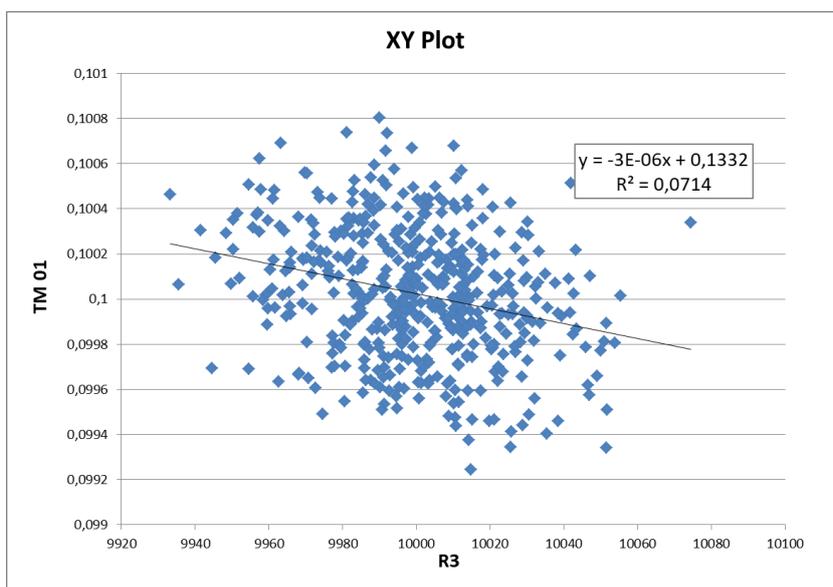


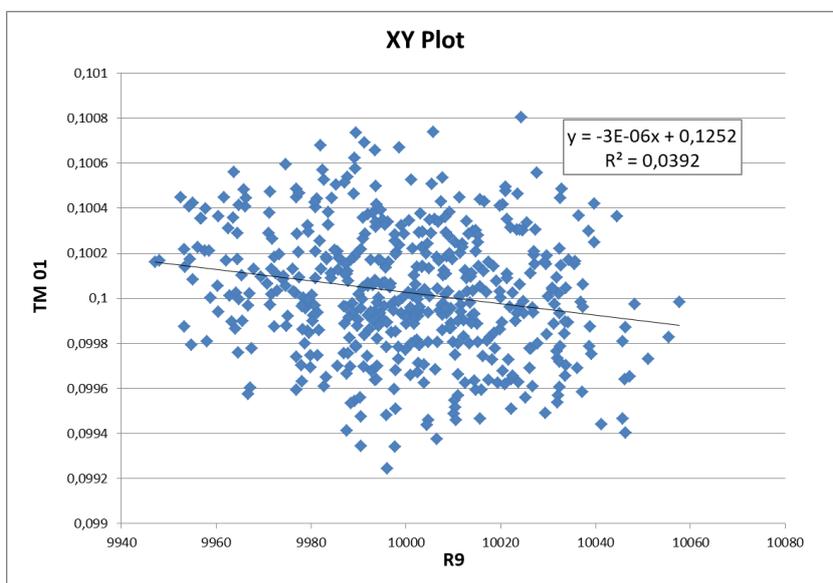
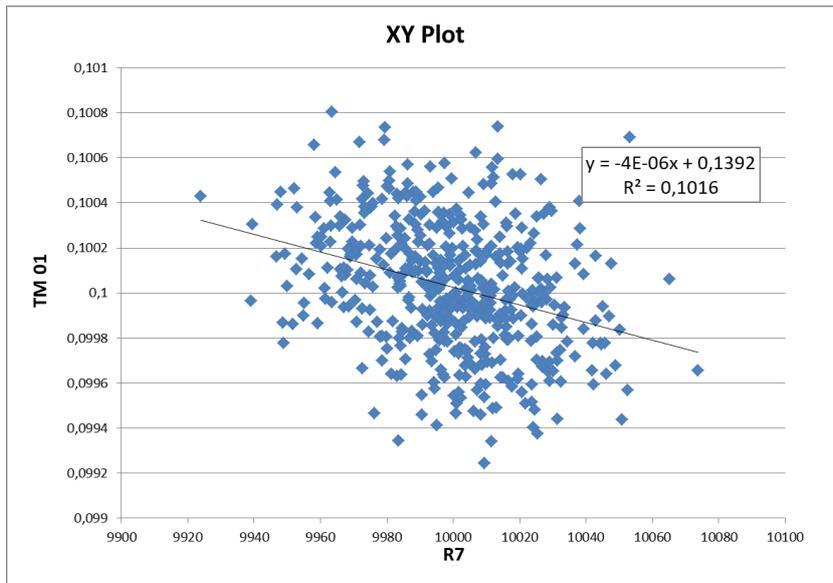
Für die Widerstände 2R ist der Widerstand R2 der für TM01 den Strom liefert der mit dem signifikanten Einfluss, alle anderen haben einen geringen Einfluss.

Um die Ergebnisse direkt miteinander zu vergleichen müssen wir bedenken das dies der Wert ist mit dem die Widerstandswertänderung die Spannung beeinflusst. Wenn wir unterschiedliche Widerstandswerte bzw. Widerstandstoleranzen in der Schaltung haben, dann reicht es nicht nur die Steigungswerte aus dem Diagramm miteinander zu vergleichen.

Man muss die Steigungswerte aus dem Diagramm mit der absoluten Toleranzvariation multipliziert werden, bei 20k Ohm und 1% rel Toleranz ist die absolute Toleranz 200 Ohm. Die absolute Toleranz ist der Widerstandsnominalwert multipliziert mit der relativen Toleranz.

Die folgenden Diagramm zeigen alle Widerstandswerte R aus dem R2R Netzwerk

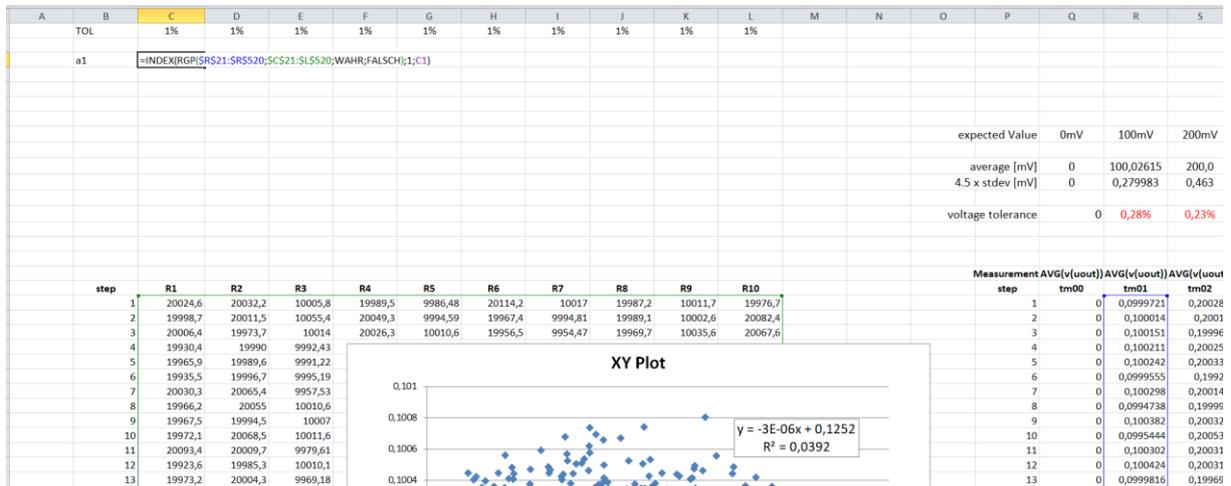




Die Widerstände R haben alle geringen Einfluss

Der Einfluss auf TM01 ist im Wesentlichen durch R2, aber auch die Summe der 9 anderen Widerstände

Diese Zahlenwerte vergleicht man am besten im Tabellenblatt, dazu kann man diese Information aus dem Diagramm mit einer Excel Funktion in den Zellen berechnen lassen, die Darstellung wird damit übersichtlicher und der Aufwand geringer.



Wichtig ist hier die Vorbereitung der oberen Zeile mit den Index Werte, beginnen mit der Anzahl Spalten wird dies abwärts gezählt, das muss sein, weil die automatische Sortierung von Microsoft andersherum gewählt wurde, dann hätte man den Koeffizienten für R10 über R1 stehen und umgekehrt, sowie R2 über R9 usw. das ist zu unübersichtlich. Deshalb benutze ich die Index Funktion um die Rückgabematrix der RGP Funktion sinnvoll zu sortieren. Wichtig sind auch die \$ Zeichen vor den Spalten, für die Zeilen ist es egal, stört aber nicht.

```
=INDEX(RGP($R$20:$R$520;$C$21:$L$520;WAHR;FALSCH);1;C1)
```

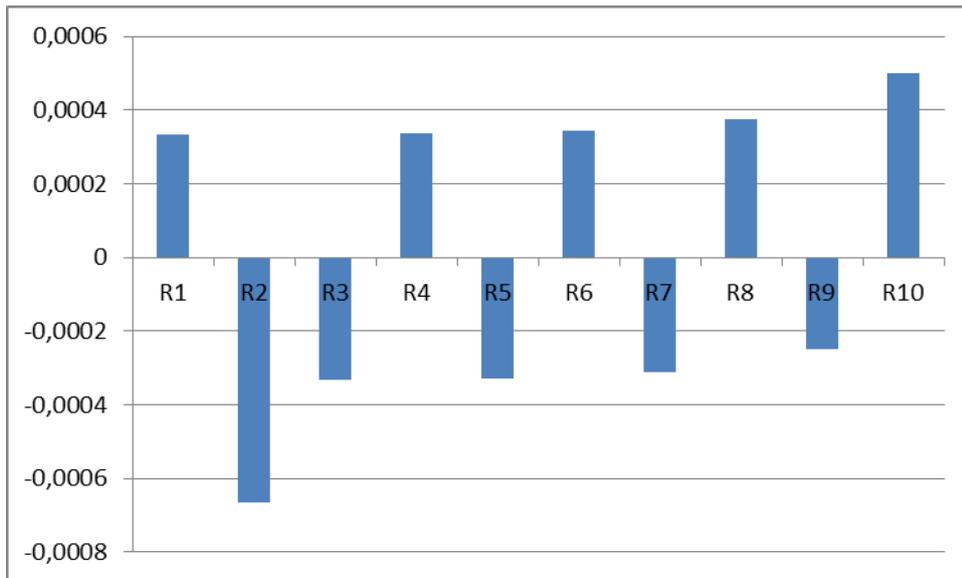
Diese Formel kann man dann von Spalte C nach rechts bis Spalte L kopieren

Und als nächstes die Steigung mit der Widerstandstoleranz und dem Widerstandswert multiplizieren

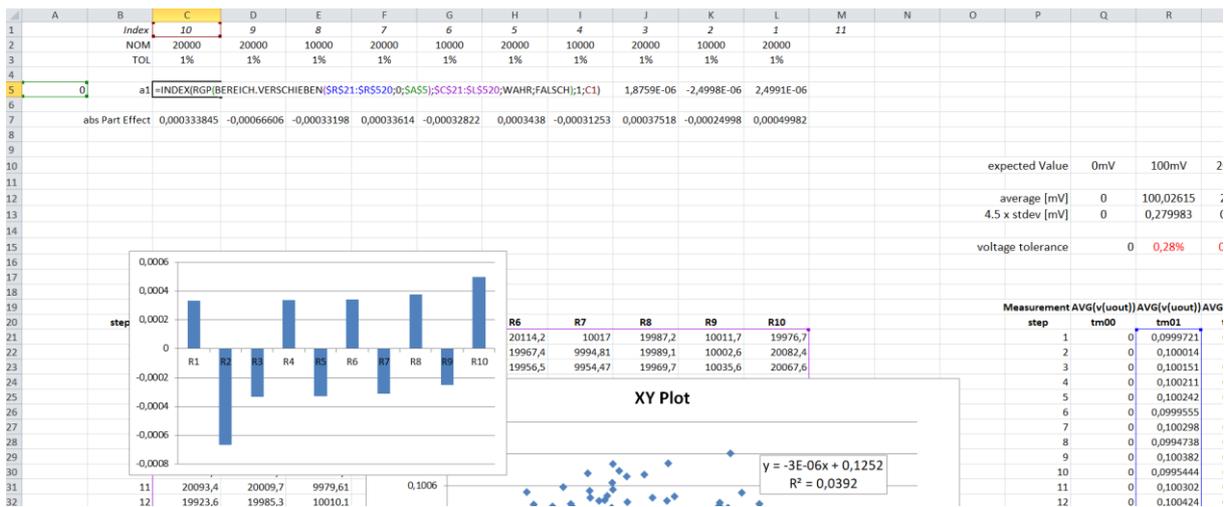
	A	B	C	D	E	F	G	H	I	J	K	L
1		Index	10	9	8	7	6	5	4	3	2	1
2		NOM	20000	20000	10000	20000	10000	20000	10000	20000	10000	20000
3		TOL	1%	1%	1%	1%	1%	1%	1%	1%	1%	1%
4												
5		a1	1,66923E-06	-3,3303E-06	-3,3198E-06	1,6807E-06	-3,2822E-06	1,719E-06	-3,1253E-06	1,8759E-06	-2,4998E-06	2,4991E-06
6												
7		abs Part Effect	=C2*C3*C5		-0,00033198	0,00033614	-0,00032822	0,0003438	-0,00031253	0,00037518	-0,00024998	0,00049982
8												
9												
10												
11												

„abs“ bezieht sich nicht auf das Vorzeichen sondern steht als Gegensatz zu „relativ“.

Die Einflussgrößen als Balkendiagramm



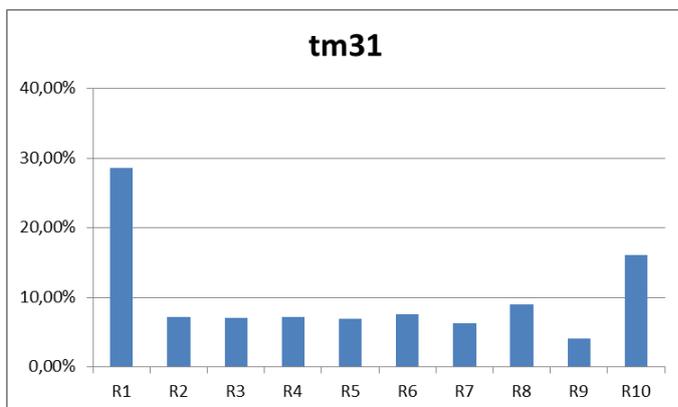
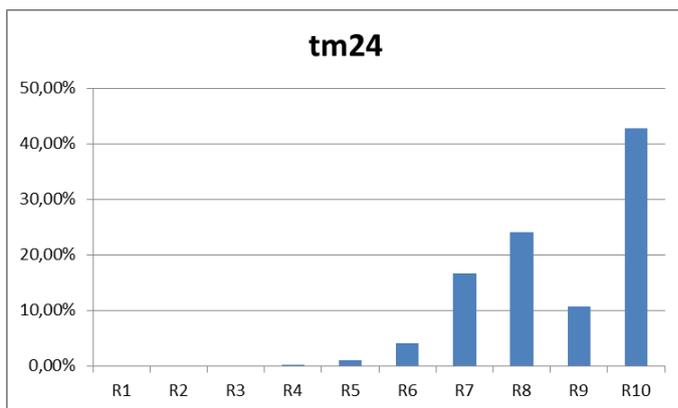
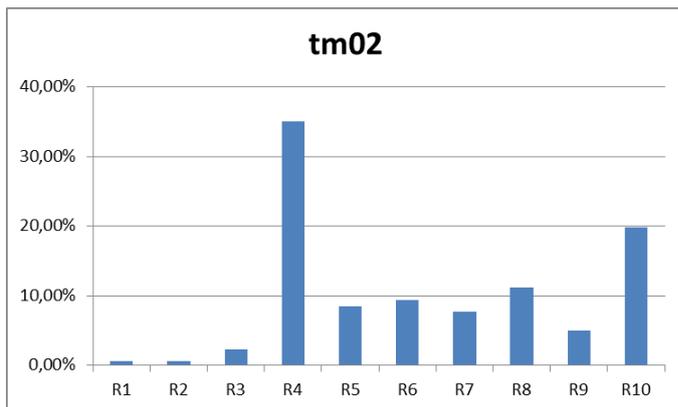
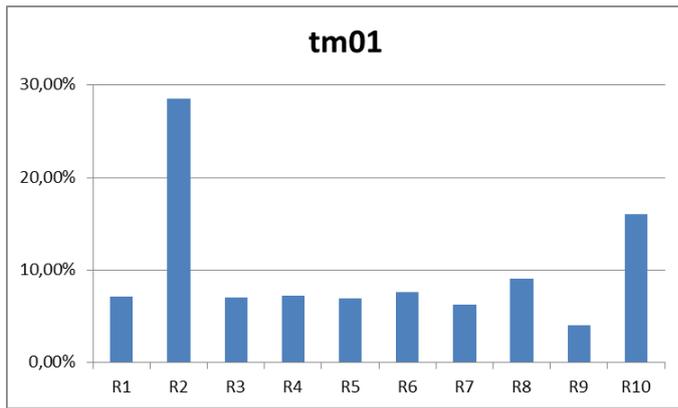
Das kann man jetzt für alle Ausgangsgrößen machen, oder man erweitert die Funktion mit Index und RGP noch um die Funktion Bereich.verschieben so dass wir die Ausgangsspannung TM01 bis TM31 nacheinander betrachten können.



`a1 =INDEX(RGP(BEREICH.VERSCHIEBEN(R21:R520;0;A5);C21:L520;WAHR;FALSCH);1;C1)`

Ich habe den Index um die Spalten zu verschieben in Zelle \$A\$5 definiert, auch hier die \$ Zeichen beachten, nach der Anpassung die Formel wieder bis in Spalte L nach rechts kopieren.

Für den Komfort kann man noch eine Schaltfläche einfügen



Diese Darstellung berücksichtigt aber nicht, dass bei TM24 und TM31 generell die Toleranz schon sehr klein gewesen ist, deshalb kann man in diesem Fall hingehen und diese Werte nochmals mit dem jeweiligen absoluten Fehler gewichten.

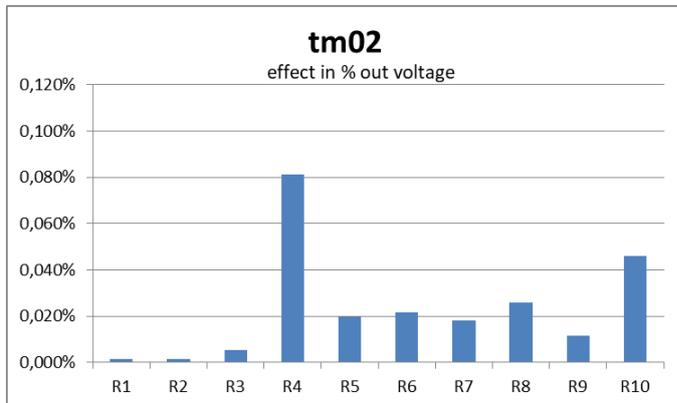
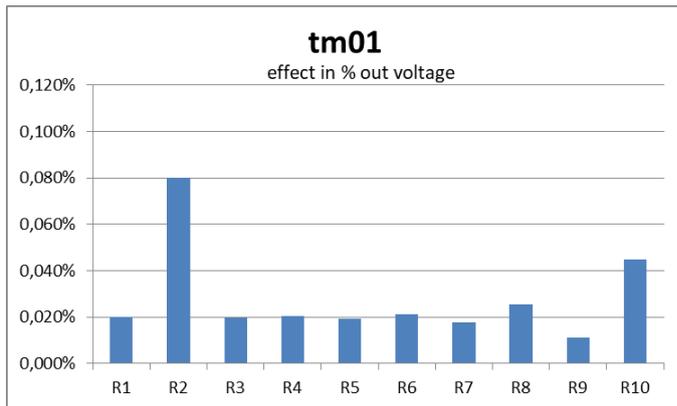
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1		Index	10	9	8	7	6	5	4	3	2	1	11					
2		NOM	20000	20000	10000	20000	10000	20000	10000	20000	10000	20000						
3		TOL	1%	1%	1%	1%	1%	1%	1%	1%	1%							
4																		
5		a1	2,09809E-06	-2,8992E-06	-1,607E-06	3,4034E-06	3,5891E-06	-1,1405E-05	-1,5628E-05	9,3824E-06	-1,2497E-05	1,2497E-05						
6		tm05																
7		abs Part Effect	0,000419617	-0,00057984	-0,0001607	0,00068067	0,00035891	-0,0022809	-0,00156277	0,00187647	-0,00124971	0,0024993						
8		square effect	1,8E-07	3,4E-07	2,6E-08	4,6E-07	1,3E-07	5,2E-06	2,4E-06	3,5E-06	1,6E-06	6,2E-06	sum square	2,0E-05				
9		effect %	0,88%	1,67%	0,13%	2,30%	0,64%	25,88%	12,15%	17,51%	7,77%	31,07%						
10		voltage tol																
11																		
12																		
13		=BEREICH.VERSCHIEBEN(R15;0;A5)		0,003%	0,000%	0,005%	0,001%	0,054%	0,025%	0,036%	0,016%	0,065%						
14																		
15																		
16																		

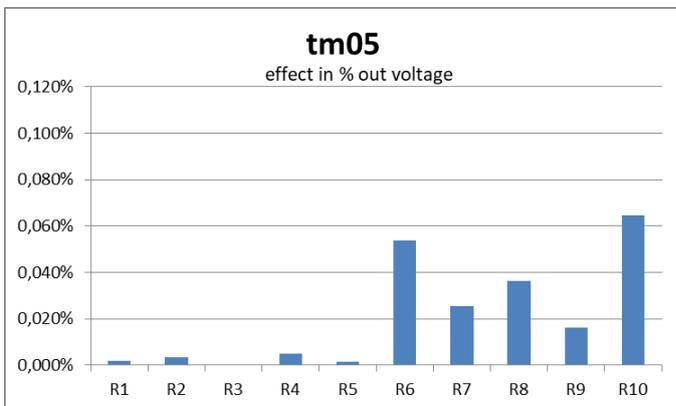
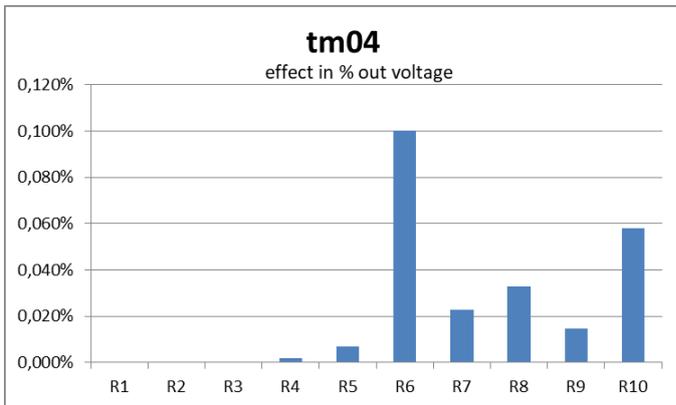
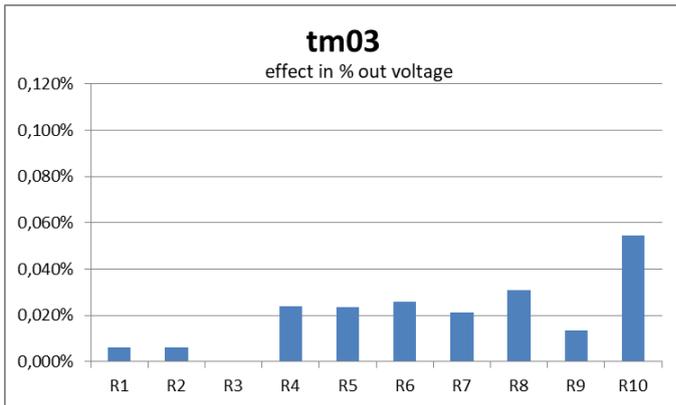
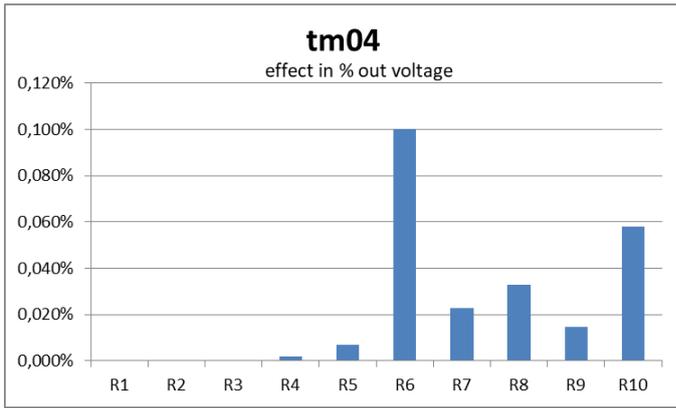
Dazu nehme wieder die Funktion Bereich.Verschieben und benutze den gleichen Index aus Zelle A5, und nehme den Bezug für den prozentualen Fehler.

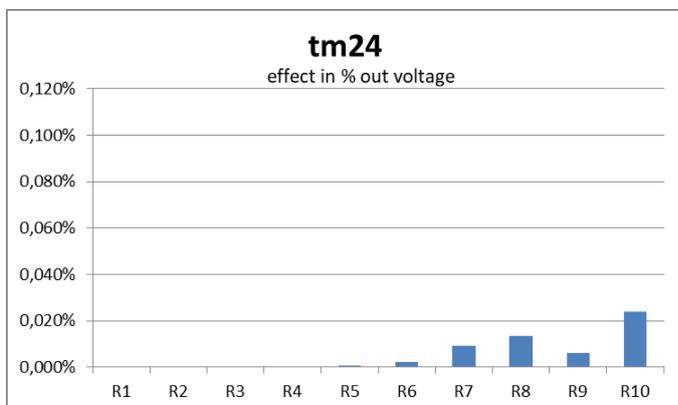
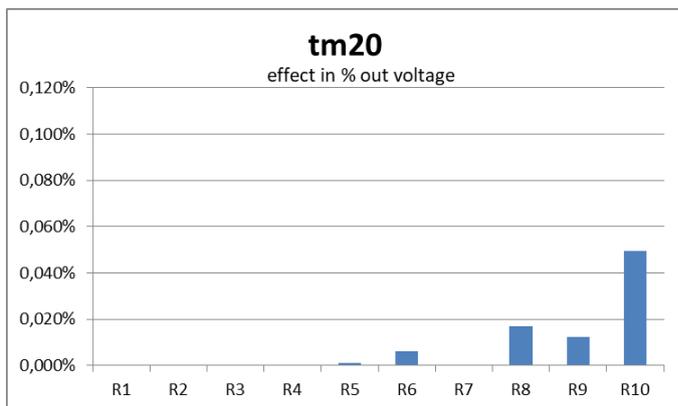
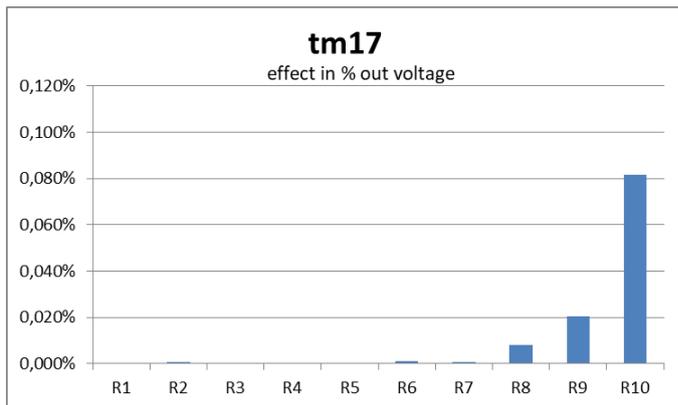
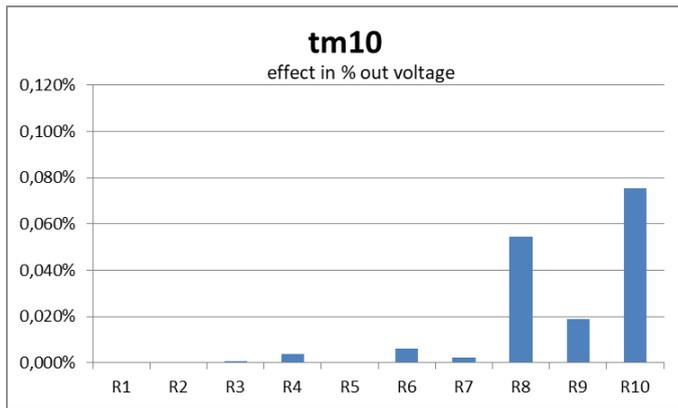
	effect %	0,88%	1,67%	0,13%	2,30%	0,64%	25,88%	12,15%	17,51%	7,77%	31,07%
voltage tol											
	0,208%	=SAS13*C11	0,003%	0,000%	0,005%	0,001%	0,054%	0,025%	0,036%	0,016%	0,065%

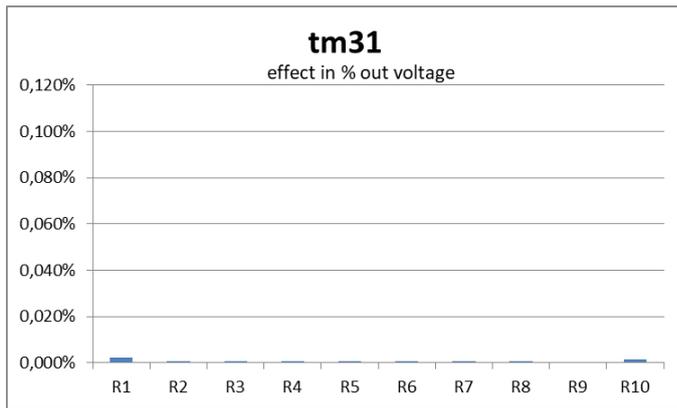
In einer weiteren Zeile wird dann der %effect jeweils mit dem Fehler verknüpft. Auch diese Formel wird bis Spalte L nach recht kopiert.

Die Balkendiagramme sehen das so aus









Man erkennt das alle Widerstände mit geraden Index, R2, R4, R6, R8, R10, mit 20K einen höheren Einfluss haben als die mit ungeraden Index die alle 10k haben.

Um dies nochmal zu verdeutlichen habe ich zwei weitere Simulationsdurchläufe mit jeweils 500 Simulationen gemacht. In einem Szenario habe ich alle Widerstände mit 20K mit 0.1% und alle 10K mit 1% simuliert, was auf Basis vom Ergebnis optimal wäre und im zweiten Szenario genau das Gegenteil, alle Widerstände mit 20K bei 1% belassen und alle mit 10K auf 0.1%. Das Ergebnis habe ich hier im Vergleich alles mit 1% zusammengefasst.

expected Value	0mV	100mV	200mV	300mV	400mV	500mV	1000mV	1700mV	2000mV	2400mV	3100mV
average [mV]	0	100,0	200,0	300,0	400,0	500,0	1000,0	1700,0	2000,0	2400,0	3100,0
4.5 x stdev [mV]	0	0,154	0,246	0,360	0,410	0,483	0,622	0,860	0,724	0,728	0,196
voltage tolerance all 1%	0%	0,32%	0,28%	0,26%	0,28%	0,25%	0,07%	0,08%	0,05%	0,07%	0,01%
voltage tolerance R2 R4 R6 R8 R10 0.1%	0%	0,15%	0,12%	0,12%	0,10%	0,10%	0,06%	0,05%	0,04%	0,03%	0,01%
voltage tolerance R1 R3 R5 R7 R9 1%	0%	0,23%	0,20%	0,17%	0,21%	0,18%	0,15%	0,10%	0,08%	0,05%	0,01%

Measurement: tm00	tm01	tm02	tm03	tm04	tm05	tm10	tm17	tm20	tm24	tm31
step	AVG(v(uout))									
1	0	0,099828	0,199997	0,299979	0,399415	0,499998	1,00032	1,7007	2,00014	2,40073
2	0	0,099859	0,200233	0,300092	0,400222	0,500081	1,00032	1,69959	1,99995	2,40073
3	0	0,099995	0,200077	0,300077	0,400419	0,500419	0,998934	1,70069	2,00111	2,39955
4	0	0,100081	0,200235	0,300316	0,400736	0,500817	1,00073	1,69811	1,99877	2,39852
5	0	0,100128	0,200238	0,300366	0,399767	0,499895	0,999736	1,7002	1,99984	2,39957
6	0	0,0998181	0,199673	0,299491	0,400244	0,500062	1,00024	1,69938	1,9998	2,40013
7	0	0,100306	0,200114	0,30042	0,400223	0,500528	0,999053	1,70054	2,00045	2,39917
8	0	0,0997813	0,199839	0,29962	0,400215	0,499996	1,00018	1,69963	2,00006	2,40019
9	0	0,100071	0,200235	0,300306	0,400408	0,500448	1,00011	1,69925	1,99958	2,39905
10	0	0,0999082	0,200128	0,300036	0,399947	0,499855	1,00046	1,69951	1,99955	2,39993
11	0	0,100208	0,199995	0,300203	0,399715	0,499924	0,999582	1,70096	2,00046	2,40033

Original ist die grau hinterlegt Zeile, die empfohlene Optimierung grün hinterlegt und die eigentlich falsche Entscheidung orange hinterlegt.

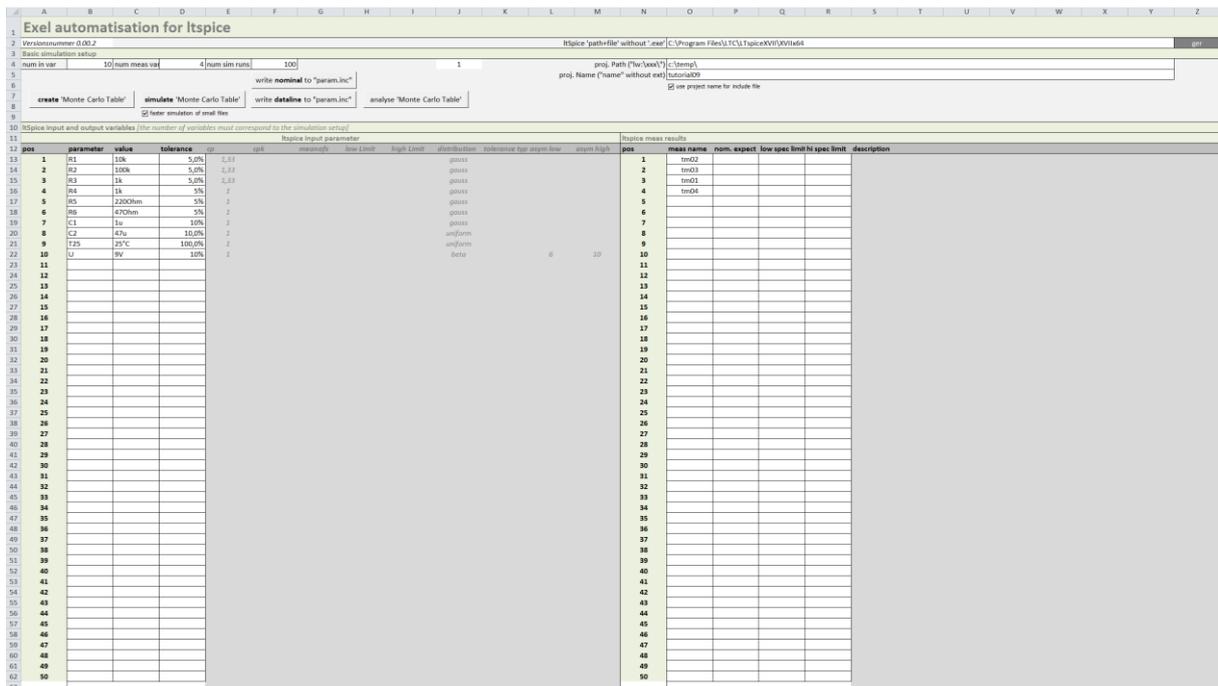
Insbesondere im mittleren Ausgangsspannungsbereich 400mV bis 1700mV erkennt man, dass das Ergebnis bei der empfohlenen Optimierung um Faktor 2 besser ist als die „falsch“ Optimierung und das bei gleicher Anzahl 0.1% Widerstände und damit gleichen Kosten für die Optimierung.

Ich hoffe mit der detaillierten Beschreibung der Methodik ein wenig das Verständnis für die Grundlagen und auch Vertrauen in die Statistik aufgebaut zu haben. Das Ganze ist aber sehr aufwendig und fehleranfällig, deshalb soll als nächstes ein kleines Excel Hilfsmittel vorgestellt werden das uns diese Arbeit erleichtert.

ItSpice Simulation aus Excel gesteuert

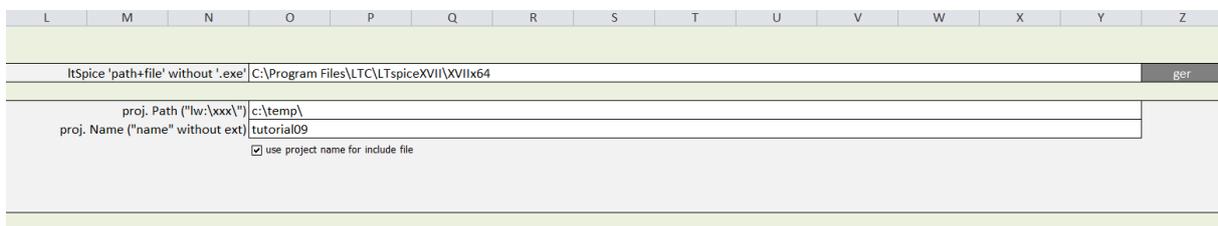
dazu habe ich ein Makro geschrieben welches die Werte für eine Monte Carlo Analyse automatisch erstellt und in das Tabellenblatt schreibt. Mit dem Start der Simulation aus Excel heraus wird ItSpice im Batch Modus betrieben und Excel liest nach erfolgter Simulation die Werte aus dem Logfile und schreibt dies in das Tabellenblatt.

Ein Nachteil ist allerdings das Excel nicht besonders schnell ist andere Programme aufzurufen, d.h. für jeden Simulationsdurchlauf benötigt Excel ca. 1s, bei dem Beispiel Spannungsteiler sind wir dann bzgl. reiner Rechenzeit um Faktor 10 langsamer, bei großen Schaltungen relativiert sich das dann aber.



Das Programm hat auf der Setup Seite vier Bereiche

1. Die Projekt und ItSpice Pfadangaben



- Hier wird der Pfad zu der ItSpice.exe angegeben incl. Dateinamen aber ohne die Extension
- Der Projektpfad der ItSpice Daten, also wo die *.asc Datei sich befindet
- Der Projekt Name, auch ohne die Extension
- Sowie die Angabe ob der Standardname „param.inc“ oder *projectname*+“inc“ für die in ItSpice einzubindende Include genutzt wird, im Bild oben würde die Include Datei „tutorial09.inc“ heißen
- Weiterhin kann zwischen der Landeseinstellung deutsch oder english gewählt werden, was den Austausch „punkt“ zu „komma“ betrifft.

2. Die Einstellung der Simulation

	A	B	C	D	E	F	G	H	I	J	K
1	Exel automatisierung for Itspice										
2	Versionsnummer 0.00.2										
3	Basic simulation setup										
4	num in var	10	num meas var	4	num sim runs	100				1	
5								write nominal to "param.inc"			
6											
7		create 'Monte Carlo Table'		simulate 'Monte Carlo Table'				write dataline to "param.inc"		analyse 'Monte Carlo Table'	
8											
9								<input checked="" type="checkbox"/> faster simulation of small files			
10	ItSpice input and output variables [the number of variables must correspond to the simulation setup]										

- Hier wird die Anzahl an Variablen angegeben die unten in der Liste aufgeführt sind
- Die Anzahl an measure Rückgaben die in der Liste aufgeführt sind
- Die Anzahl an Simulationsdurchläufen
- Der Button create „Monte Carlo“ füllt die Tabelle „Meas Result“ mit den Werten
- Simulate Monte Carlo startet die ItSpice Simulation
- Die Aktivierung der Checkbox führt dazu, dass sämtliche Werte aller Simulationsdurchläufe gleichzeitig in die Include-Datei geschrieben werden. Dies beschleunigt die Simulation. Bei komplexen Schaltungen mit umfangreichen „*.raw“-Dateien und vielen Durchläufen kann es jedoch sinnvoll sein, diese Option zu deaktivieren.
- „write nominal to ,param.inc“ schreibt die Nominalwerte in das Parameterfile
- „write dataline to ,param.inc“ schreibt eine vorgegebene Simulationszeile in das Parameterfile (der Name für ,param.inc“ entspricht der Einstellung in 1)
- „analyse ,Monte Carlo Table“ erstellt automatisch die Formeln auf dem Tabellenblatt „Meas-Result“

3. Die Definition der Schaltplan Parameter

ItSpice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
13	1	R1	10k	5,0%	1,33				gauss			
14	2	R2	100k	5,0%	1,33				gauss			
15	3	R3	1k	5,0%	1,33				gauss			
16	4	R4	1k	5%	1				gauss			
17	5	R5	220Ohm	5%	1				gauss			
18	6	R6	470hm	5%	1				gauss			
19	7	C1	1u	10%	1				gauss			
20	8	C2	47u	10,0%	1				uniform			
21	9	T25	25°C	100,0%	1				uniform			
22	10	U	9V	10%	1				beta		6	10
23	11											
24	12											
25	13											
26	14											
27	15											
28	16											
29	17											

Hier werden die Werte für die Simulation definiert, Parameter ist der Variablen Namen wie dieser in ItSpice in den geschweiften Klammern {} verwendet wird, value ist der Wert, hier werden alle Prefixes von 1p bis 1T unterstützt, es wird empfohlen statt ,μ' ein ,u' zu verwenden. Tolerance wird in % angegeben. Es werden relative und absolute Toleranz Angaben zu unterstützt, also R1 10K ±1% oder ofs 0V ±1mV (relative funktioniert hierbei nicht). Deshalb ist bei der relativen Toleranz die Formatierung in % anzuwenden, andernfalls wird es vom Programm als absolute Toleranz angesehen.

Anm.:ItSpice hat schon mal Probleme verursacht wenn ein Texteditor ein ,μ' schreibt, ItSpice kann beides μ und u wenn es im Schaltplan Editor eingegeben wird. Das Makro konvertiert μ zu u so dass es eigentlich keine Probleme geben sollte. Aber mit dem u gibt es keine Probleme

Hinweis: es können verschiedene Verteilungen ausgewählt werden, Gauss und gleichmäßige Verteilung sowie zwei verschiedene asymmetrische Verteilung. In dieser Anleitung wird die Anwendung dieser Verteilungen nur oberflächlich besprochen. Asymmetrische Verteilungen findet man z.B. in Datenblättern von OP für Eingangströme oder Eingangsoffsetspannungen, Angaben wie **min. 1uA – typ. 3uA – max. 15uA**.

Verteilungen

pos	parameter	value	tolerance	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	1k	5,0%	1,33				gauss			
2	R2	1k	5,0%	1,33				uniform			
3	V1	10m	100m	1,33				log			
4	V2	200m		1,33				log_limit	1m	1	
5	V3	300m		1,33				beta_limit	1m	1	

Es werden 4 verschiedene Verteilungen unterstützt, diese werden in Spalte „distribution“ durch die entsprechenden Schlüsselwörter definiert

- „gauss“ das Programm generiert Werte die einer gaußschen Normalverteilung entsprechen, dabei entspricht der Mittelwert, dem Wert aus der Spalte „Value“, die Standardabweichung der Gaussverteilung dem Wert Toleranz/(3*cp). Cp ist der Fähigkeitsindex und ist ein Maß für die Qualität (übliche Werte zwischen 1.33 und 2). (siehe Anhang)
- „uniform“ das Programm generiert Werte die gleichmäßig verteilt sind, dabei entspricht der Mittelwert, dem Wert aus der Spalte „Value“, der größte mögliche Wert ist Mittelwert + Tolerance, der kleinste mögliche Wert ist Mittelwert - Tolerance
- „log“ oder „log_limit“ das Programm generiert Werte die einer logarithmischen Normalverteilung entsprechen. Es gibt zwei Möglichkeiten der Parametereingabe, typischer Wert und max. Wert, sowie min Wert und max Wert. Der Wert in Spalte „value“ entspricht dem Median der Verteilung, also 50% der generierten Werte sind unterhalb und 50% oberhalb dieser Vorgabe.

Parameter	Symbol	Condictions	Min	Typ	Max	Unit
Input Offset Current	$I_{I(Ofs)}$	$T_A=25^{\circ}C$		5	10	pA
Input Offset Current	$I_{I(Ofs)}$	$T_A=25^{\circ}C$	2		10	pA

Für die log Verteilung wird im Eingabefeld „tolerance“ die oberen Grenze der Verteilung bzw. der MAX Wert aus dem Datenblatt eingegeben (das ist eigentlich nicht die Toleranz ist, Toleranz ist obere Grenze – Median, bzw. Datenblatt Max-Typ)

Für die „log“ - Verteilung werden „value“ und „tolerance“ verwendet, für die „log_limit“ werden die Werte in den Spalten „asym low“ und „asym high“ eingegeben

- „beta“ das Programm generiert Werte die einer Beta Verteilung entsprechen. Als Eingabe wird der Wert in Spalte „value“ benötigt, der auch bei dieser Verteilung dem Median entspricht (50%< und 50%>), sowie die Werte in den Spalten „asym low“ und „asym high“

4. die Messgrößen

ItSpice meas results					
pos	meas name	nom. expect	low spec limit	hi spec limit	description
1	tm02				
2	tm03				
3	tm01				
4	tm04				
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					

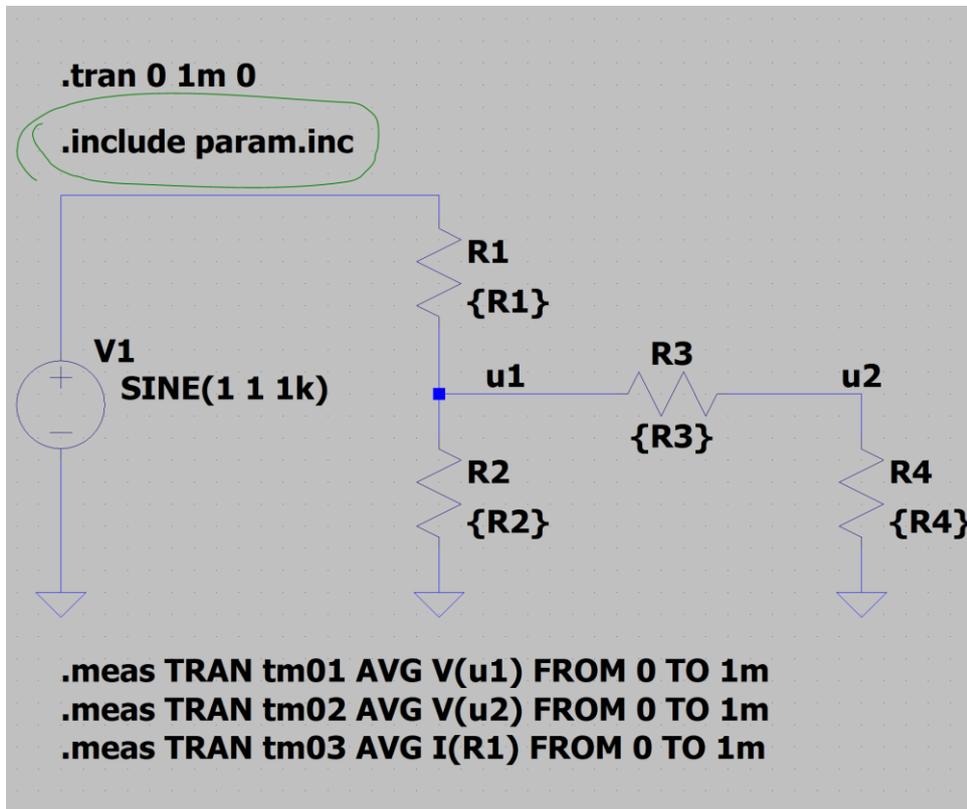
hier wird der Name wie in ItSpice im measure Befehl angegeben, zu empfehlen ist hier die Kleinschreibung, für die verschiedenen Auswertungen kann man dann den zu erwarteten Mittelwert angeben sowie die unter und obere Toleranzgrenze die das Ergebnis erfüllen soll. Die Analyse benutzt diese Werte um entsprechende statistische Analysen zu erstellen.

Anwendungs-Beispiele

Zum Abschluss dieser Anleitung möchte ich jetzt noch einige Beispiele mit Auswertung aufführen um die Funktionen des Programms näher zu erläutern

Beispiel 1

Der erste ItSpice Schaltplan dazu sieht dazu wie folgt aus



Wichtig ist hier das Einbinden des Parameter Files mittels dem Schlüsselwort „include“, das Excel Makro schreibt die Werte aus der Tabelle in dieses Textfile, sofern die Checkbox deaktiviert ist, für das erste Beispiel Projekt „ExcelTutor01“ würde das File mit aktivierter Checkbox „ExcelTutor01.inc“ heißen.

Die Vorbereitung in Excel sieht so aus

ItSpice input parameter										ItSpice meas results							
pos	parameter	value	tolerance	cp	cpk	meanofs	low limit	high limit	distribution	tolerance typ	asym low	asym high	pos	meas name	nom. expect	low spec	limit hi
1	R1	47k	5,0%		1,33				gauss				1	tm02			
2	R2	4,7k	5,0%		1,33				gauss				2	tm03			
3	R3	100k	5,0%		1,33				gauss				3	tm01			
4	R4	10k	5,0%		1,33				gauss				4				
5													5				
6													6				
7													7				
8													8				
9													9				
10													10				

Wichtig: die Wert für „num var“ und „num meas var“ müssen der tatsächlichen Anzahl in der Tabelle entsprechen

Im nächsten Schritt werden diese Daten wie in den Beispielen vorher analysiert. Hierbei gibt es jetzt aber weitere Unterstützung, das Makro Analyse fügt automatisch alle Formeln ein die wir vorher manuell eingegeben haben. Außerdem gibt es noch weitere statistische Analyse. Diese werde ich aber nur in der Anwendung der Ergebnisse erläutern und nicht die mathematischen Hintergründe.

sName	tm02	tm03	tm01
sNominalExp	7.95597m	19.416u	87.5156m
dNominalExp	0	0	0
dLowSpecLi	0	0	0
dHighSpecLi	0	0	0
Capability Cp			
Average	0,0079	1,94E-05	0,0873
SteDev	1,58E-04	2,24E-07	0,0012
lo Limit 4.5S	0,0072	1,84E-05	0,0818
up Limit 4.5S	0,0086	2,04E-05	0,0928
AD Test	0,2235	0,9788	0,8865

Das Ergebnis zeigt uns das wir für die Ausgangsspannung V(u2) einen Mittelwert von 7.9mV haben sowie einen untere Grenze loLimit 4.5S von 7.2mV und eine obere Grenze „upLimit 4.5S von 8.6mV.

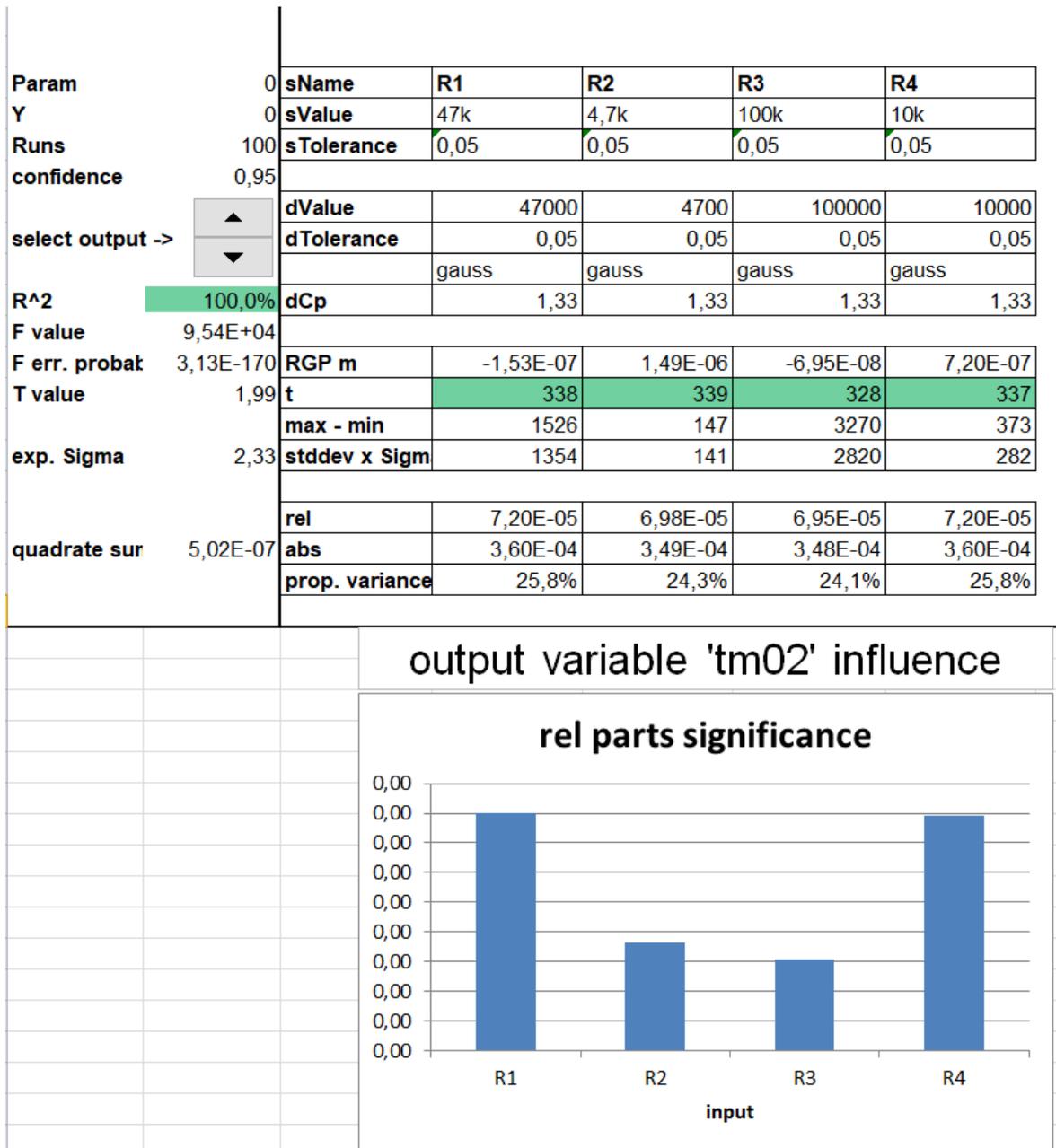
Die Grenzen werden mit ± 4.5 Sigma gerechnet, also 4.5 mal der ermittelten Standardabweichung, das bedeutet das nur 3.6ppm (parts per million) ausserhalb der Grenze liegen . Das entspricht durchaus der gängigen Praxis in der Industrie. Wenn jemand den Einwand hat, es heißt doch „Six Sigma“, ja das ist richtig, bei kurzzeitigen Prozess Analysen geht man davon aus, das dies langfristig sich verschiebt, um 1.5 Sigma. Die Datenblatt Angaben die wir verwenden sind aber nicht kurzfristig, deshalb rechnen wir mit 4.5 und nicht mit 6.0.

Die Auswertung der Einflüsse auf der nächsten Seite zeigt uns einige Informationen mehr als wir und in den manuellen Beispielen erarbeitet haben. Neben der Korrelation kann Excel noch einiges mehr berechnen, welches für unsere Analysen nützlich ist.

Dazu zählen in erster Linie:

- „R²“, dieser Wert zeigt uns an, wie gut die ermittelten Parameter das Modell beschreiben, wenn der Wert 0.5 ist., also 80%, dann bedeutet dies, das mit den Linear-Koeffizienten 80% des Verhaltens beschrieben werden und 20% nicht dadurch erklärt werden. (das könnten dann z.B. quadratische Faktoren sein)
- „F err porbability“ der Wert sollte auch immer mindestens kleiner 0.01 bzw 1% sein und zeigt die Wahrscheinlichkeit ob die Modellannahme falsch ist
- „T-Value“ und „t“ der Koeffizienten, jeder Koeffizient dessen t größer T hat ein systematischen Einfluss, ist das t kleiner, dann hat der koeffizient keinen Einfluss und wird nicht berücksichtigt in der weiteren Darstellung

Was uns aber wirklich interessiert, welche Bauteile beeinflussen meine Schaltung, das sehen wir auf der nächsten Seite und das Makro hat die oben aufgeführten Informationen so aufgearbeitet, das es wieder einfacher ist.

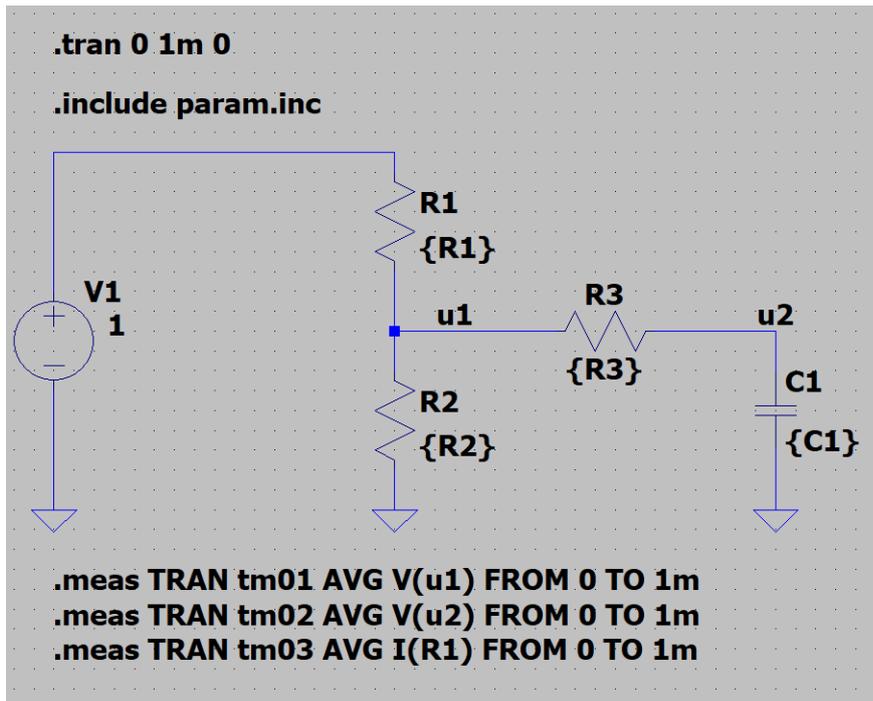


Was uns hier letztendlich interessiert sind die Zeilen 17, 18 und 19, die den Einfluss der Bauteile auf die Toleranz (hier Spannung netz $V(u_2)$) darstellen. In den meisten Fällen ist die Zeile 19, das Diagramm „parts proportion to tolerance“ ausreichend, da sie den jeweils prozentualen Anteil auf das Ergebnis darstellt. Hier ist der Anteil von allen Bauteilen gleich groß.

Mit diesem Ergebnis können wir dann die Eigenschaften einer Schaltung komfortabel untersuchen und optimieren. In diesem Fall bedeutet dies das alle Widerstände gleichermaßen genauer sein müssen, weil alle den gleichen Einfluss aufweisen.

Beispiel 2

Das zweite Beispiel sieht in LtSpice so aus

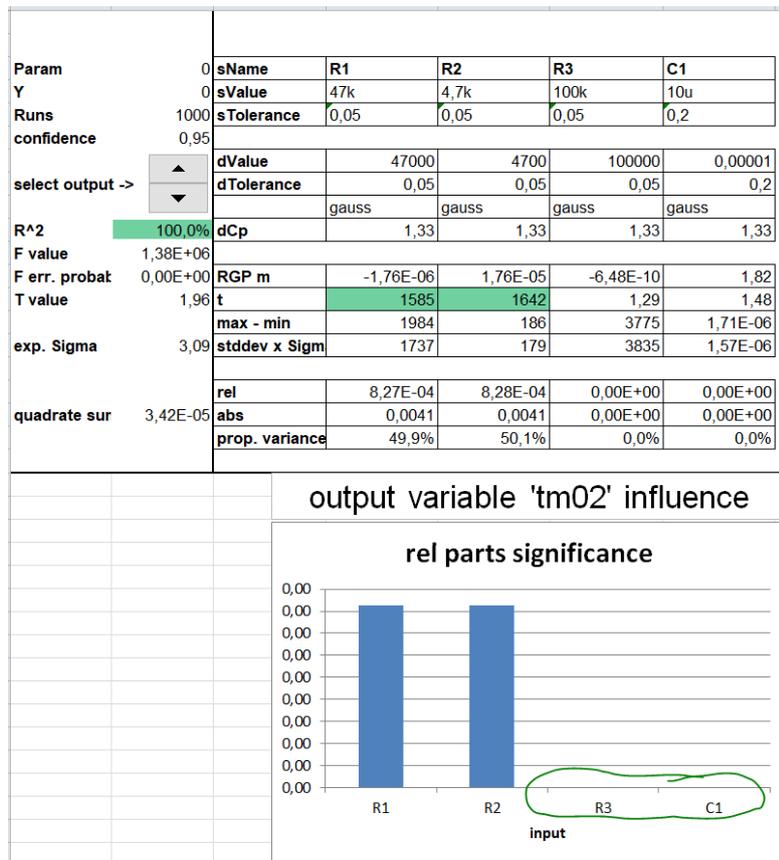


Wir erkennen das R3 und C1 auf das Gleichspannungsverhalten keinen Einfluss haben, dies zeigt uns auch die Auswertung in Excel

	A	B	C	D	E	F	G	H	I
1									
2									
3	Param	0	sName	R1	R2	R3	C1		
4	Y	0	sValue	47k	4,7k	100k	10u		
5	Runs	100	sTolerance	0,05	0,05	0,05	0,2		
6	confidence	95%							
7			dValue	47000	4700	100000	0,00001		
8			dTolerance	0,05	0,05	0,05	0,2		
9				gauss	gauss	gauss	gauss		
10	R^2	100,0%	dCp	1,33	1,33	1,33	1,33		
11	F value	2,76E+05							
12	F err. probabi	3,68E-192	RGP m	-1,76E-06	1,76E-05	3,36E-09	-1,11		
13	T value	1,99	t	669	718	2,79	0,3811		
14			max - min	1108	143	3803	1,10E-06		
15	exp. Sigma	2,33	stddev x Sigma	1192	131	2720	1,13E-06		
16			rel	8,27E-04	8,26E-04	3,36E-06	0,00E+00		
18	qudrate sum	3,41E-05	abs	0,0041	0,0041	1,68E-05	0,00E+00		
19			prop. variance	50,1%	49,9%	0,0%	0,0%		
20									
21				46745,12867	4741,437933	100972,3004	1,03061E-05		
22				46028,33402	4658,318681	99135,9832	9,72047E-06		
23									
24									
25									
26									
27									
28									
29									
30									
31									
32									
33									
34									
35									
36				47194,78352	4691,869765	100784,839	9,69962E-06		
37									
38									

Wobei hier für R3 noch ein minimaler Einfluss gezeigt, welcher der numerischen Genauigkeit in ItSpice geschuldet ist, aber da der t-Wert Faktor 200 kleiner ist und auch die der Koeffizient

entsprechend klein ist, ist die Gefahr einer falschen Beurteilung der Signifikanz trotzdem nicht gegeben. Wenn man die Anzahl der Simulationen auf 1000 erhöht ist das Ergebnis wie erwartet



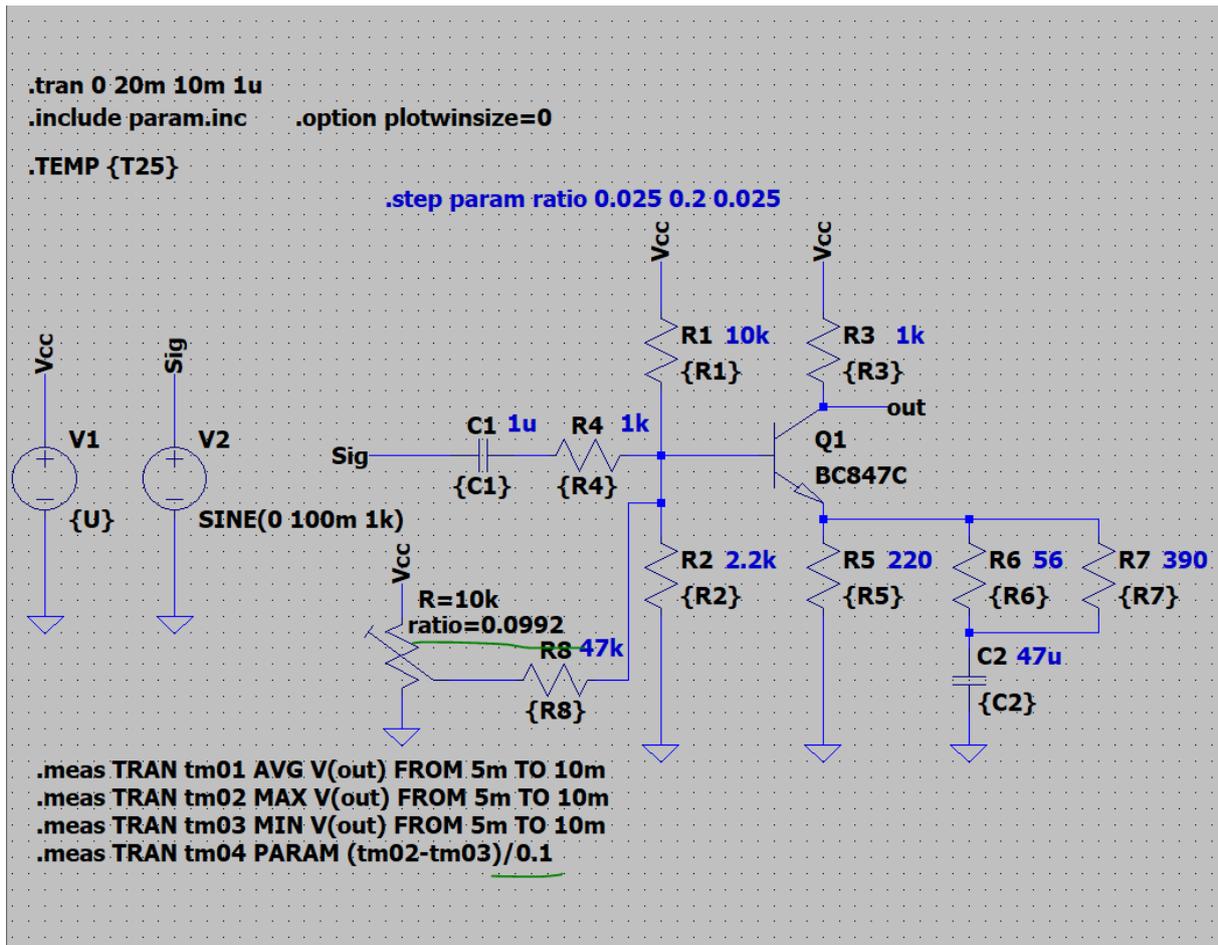
Beispiel 3

Nachdem ich mit all den ganzen Beispielen mit Widerständen und Kondensatoren hoffentlich genug Vertrauensbildende Maßnahmen getroffen habe, das die Statistik sehr gut Schaltungen bewerten kann, möchte ich als nächstes eine Schaltung mit einem Transistor berechnen.

Es soll eine Transistorverstärker in Emitterschaltung simuliert werden. Die Spannungsversorgung soll von einer 9V Blockbatterie übernommen werden, die AC Verstärkung soll 25 betragen. Es soll in einem Temperaturbereich von 0°C bis 50°C betrieben werden.

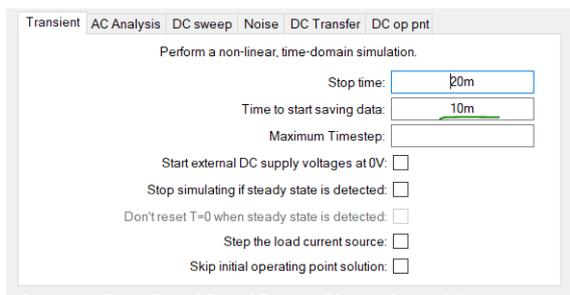
Zur Arbeitspunktstabilisierung ist die Aufgabe des Emitterwiderstand (R5) soweit als bekannt vorausgesetzt, dieser wurde hier mit einem Fünftel des Kollektorwiderstandes gewählt. Da die AC Verstärkung dem Verhältnis Kollektorwiderstand (R3) zu Gesamt Emitterwiderstand entspricht, muss zu R5 noch mit einem Kondensator in Reihe ein Widerstand geschaltet werden, so dass die Parallelschaltung $1k/25=40$ entspricht. Bei der Auslegung der Bauteile habe ich aus der E12 Reihe noch 56 Ohm und 390 Ohm gewählt, damit ist die Parallelschaltung für AC $1/(1/220+1/56+1/390)=40.05$ Ohm, damit beträgt die erwartete Verstärkung 24.97, die Simulation ergibt 24.89, das ist richtig so, da die idealisierte Berechnung der Verstärkung für $\beta \rightarrow \infty$ gilt.

Für den Arbeitspunkt habe ich mit einem „Trimmer“ den Arbeitspunkt für Uout bei 9V exakt auf $U_{ap}=(U_{vcc}+U_{re})/2=4.94684V$ eingestellt.



Um Effekt durch Einschwingverhalten auszuschließen werden die ersten 10ms nicht ausgewertet

Frage: mit dem Transistor BC547C ergibt sich in ItSpice eine Verstärkung die größer als R_c/R_e ist, wenn jemand weiß welche parasitären Effekt das bewirken können, wäre für Feedback sehr dankbar

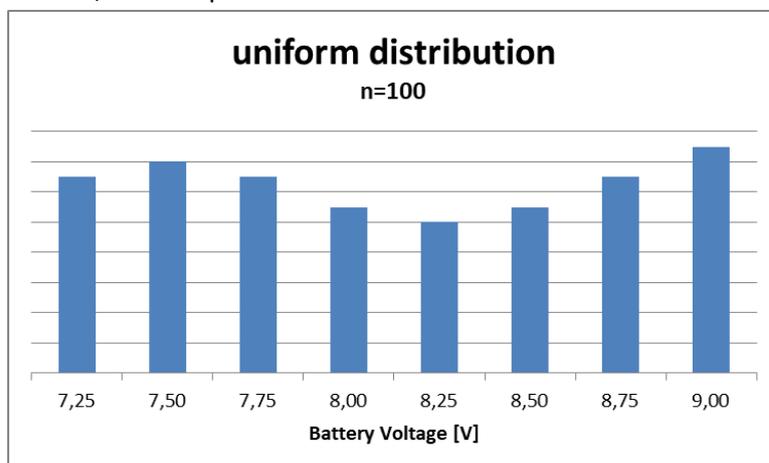


Die Messgröße tm04 entspricht die Verstärkung, weil die Eingangsspannung mit $0.1V_{pp}$ eingestellt ist.

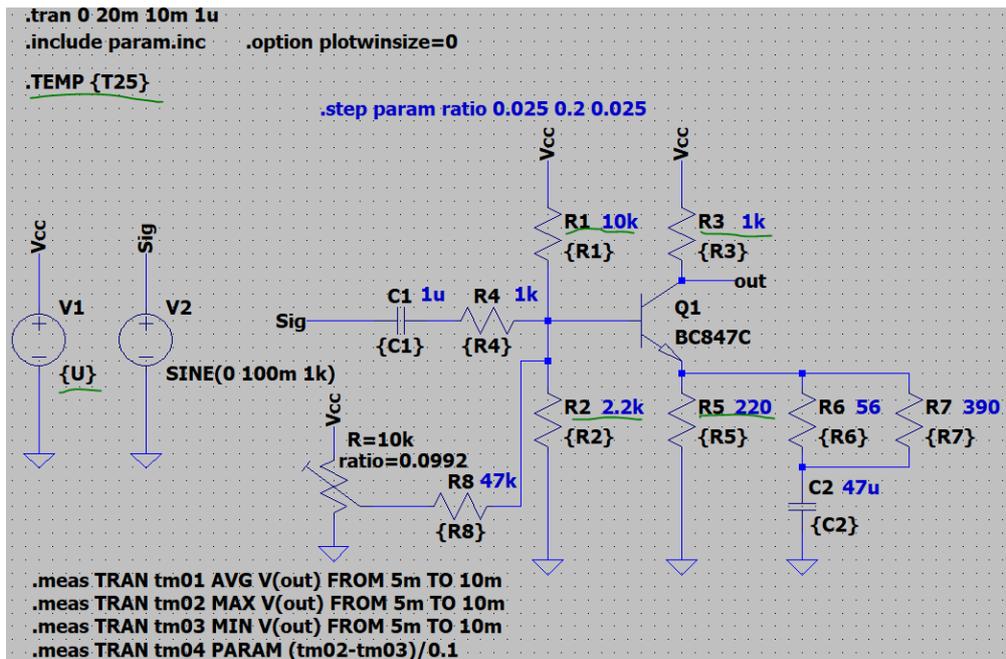
In Excel machen wir folgende Angaben für die Simulation

Excel automatisierung for ItSpice																
Versionsnummer 0.00.2											ItSpice 'path+file' without '.exe' C:\Program File					
Basic simulation setup																
2		12 num meas var		4 num sim runs		1000				1		proj. Path ("lw:\xxx\l") c:\temp\				
											proj. Name ("name" without ext) ExcelTutor03					
											<input type="checkbox"/> use project nam					
write nominal to "param.inc"																
create 'Monte Carlo Table'			simulate 'Monte Carlo Table'			write dataline to "param.inc"			analyse 'Monte Carlo Table'							
<input checked="" type="checkbox"/> faster simulation of small files																
ItSpice input and output variables [the number of variables must correspond to the simulation setup]																
ItSpice input parameter																
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high	pos	meas name	r	
13	1	R1	10k	5,0%	1,33				gauss				1	tm01		
14	2	R2	2,2k	5,0%	1,33				gauss				2	tm02		
15	3	R3	1k	5,0%	1,33				gauss				3	tm03		
16	4	R4	1k	5%	1,33				gauss				4	tm04		
17	5	R5	220Ohm	5%	1,33				gauss				5			
18	6	R6	56Ohm	5%	1,33				gauss				6			
19	7	R7	390Ohm	5%	1,33				gauss				7			
20	8	R8	22k	5%	1,33				gauss				8			
21	9	C1	1u	10%	1,33				gauss				9			
22	10	C2	47u	10,0%	1,33				gauss				10			
23	11	T25	25°C	100,0%	1				uniform				11			
24	12	U	8V	13%	1				uniform				12			
25	13												13			
26	14												14			
27	15												15			
28	16												16			
29	17												17			

- Alle Widerstandswerte weisen eine Toleranz von 5% auf, mit einer Fähigkeit von 1.33 und sind alle gauss verteilt
- Für die Kondensatoren nehmen wir 10% und auch gauss verteilt
- Für die Temperatur wollen wir den Bereich 0°C bis 50°C darstellen, das wäre dann 25°C ±100%, da alle Temperaturen gleichermaßen vorkommen nehmen wir die gleichmäßige „uniform“ Verteilung (wenn es ein Gerät für den Ausseneinsatz ist)
- Für Die Spannungstoleranz nehmen ich eine ebenfalls ein gleichmäßige Verteilung an mit 8V ±12.5%, das entspricht 7V bis 9V



Die Batteriespannung wird hier gleichmäßig verteilt angenommen, weil diese bei 9V startet und beim Erreichen von 7V wird diese ausgewechselt, d.h. über die Betriebsdauer sinkt die Spannung kontinuierlich.



sName	tm01	tm02	tm03	tm04
sNominalExpected				
dNominalExp	0	0	0	0
dLowSpecLir	0	0	0	0
dHighSpecLi	0	0	0	0
Capability Cp				
Average	4,76	5,95	3,54	24,10
SteDev	0,2039	0,2219	0,1957	0,6976
lo Limit 4.5S	3,84	4,95	2,65	20,96
up Limit 4.5S	5,68	6,94	4,42	27,24
AD Test	0,8309	0,7615	0,9227	0,0679

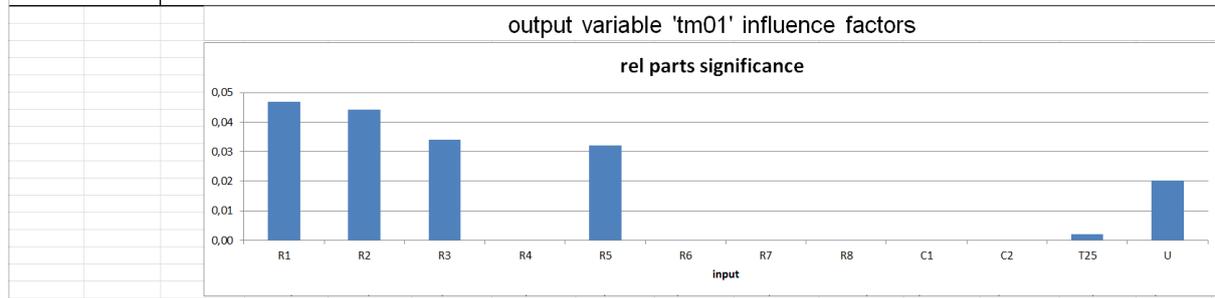
Schauen wir auf die Ergebnisse, als erstes schauen wir die Einflussfaktoren auf den Arbeitspunkt von U(out) tm01 und der Verstärkung tm04, der Arbeitspunkt variiert von 3.8V bis 5.8V. Die Verstärkung variiert von 21 bis 27.

Diese Ergebnisse sind jetzt ernüchternd wenn man den Aufwand bedenkt den Wert so optimal einzustellen. Sicher gilt hier auch zu bedenken das der Arbeitspunkt ja für jede Toleranzsituation individuell angepasst wird und somit die Streuung kleiner ausfällt. Auf der anderen Seite werden die Widerstände aber auch durch Temperatur und Alterung beeinflusst. Ein Widerstand mit einer Toleranz von 1% bei 21°C kommt im Temperaturbereich -20°C bis 70°C und Alterung aber auch auf über 3%, je nach Qualität mehr oder weniger. Insofern ist die Aussage auch wiederum nicht ganz falsch.

Wenn man es genau wissen möchte muss man aber 2 Simulationen aufbauen, eine kalibrierte für den Arbeitspunkt und eine weitere für die Verstärkung, es gibt zwar noch die Möglichkeit die Kalibrierung in der Simulation mittels einer Übertragungsfunktion einfließen zu lassen, aber das Thema würde hier den Rahmen sprengen.

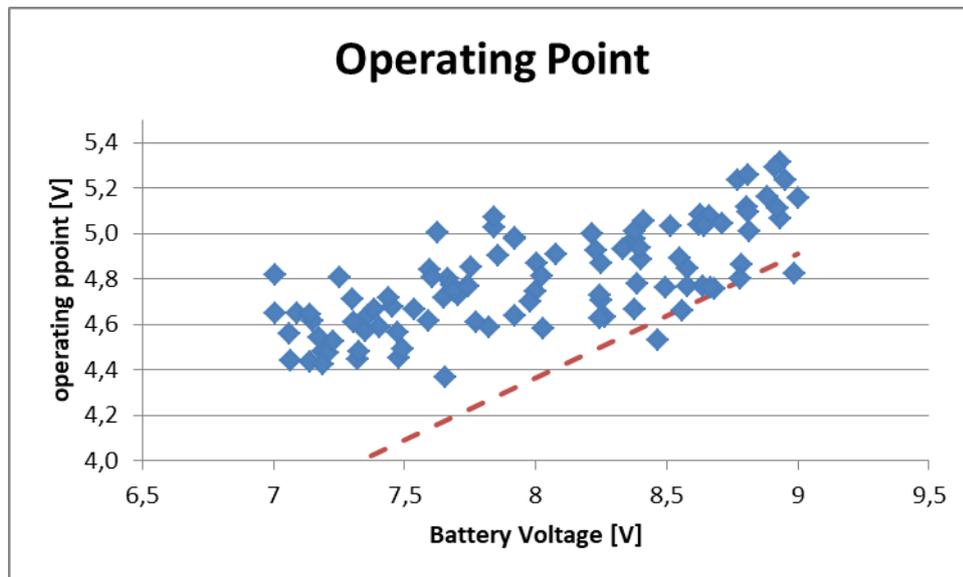
Schauen wir mal auf die Einflussfaktoren für den Arbeitspunkt, das sind die Toleranz des Spannungsteilers R1 und R2 sowie die Betriebsspannung und Temperatur, auf letzteres haben wir keinen Einfluss und können nur Schaltungskonzepte entwickeln die diesbezüglich robuster sind.

Param	sName	R1	R2	R3	R4	R5	R6	R7	R8	C1	C2	T25	U
Y	sValue	10k	2.2k	1k	1k	2200hm	560hm	3900hm	22k	1u	47u	25°C	8V
Runs	sTolerance	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,1	0,1	1	0,125
confidence	dValue	10000	2200	1000	1000	220	56	390	22000	0,000001	0,000047	25	8
select output ->	dTolerance	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,1	0,1	1	0,125
R^2	dCp	gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss	uniform	uniform
F value		1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1	1
F err. probab	RGP m	4,69E-04	-0,0020	-0,0034	4,45E-05	0,0145	0,0010	-2,66E-04	-6,18E-06	3,78E+04	-776	-0,0077	0,2522
T value	t	56,41	44,17	34,87	0,5131	37,23	0,6799	1,11	1,48	0,8334	0,9547	92,44	132
exp. Sigma	max - min	308	71,88	28,05	32,91	7,11	1,79	11,84	696	6,29E-08	3,95E-06	24,48	0,9795
quadrate sur	stddev x Sigm	315	60,49	27,18	30,04	6,63	1,68	11,27	656	5,78E-08	3,18E-06	30,81	1,35
	rel	0,0469	0,0443	0,0341	0,00E+00	0,0320	0,00E+00	0,00E+00	0,00E+00	0,00E+00	0,00E+00	0,0019	0,0202
	abs	0,2343	0,2213	0,1703	0,00E+00	0,1598	0,00E+00	0,00E+00	0,00E+00	0,00E+00	0,00E+00	0,1937	0,2522
	prop. variance	21,2%	18,9%	11,2%	0,0%	9,8%	0,0%	0,0%	0,0%	0,0%	0,0%	14,5%	24,5%



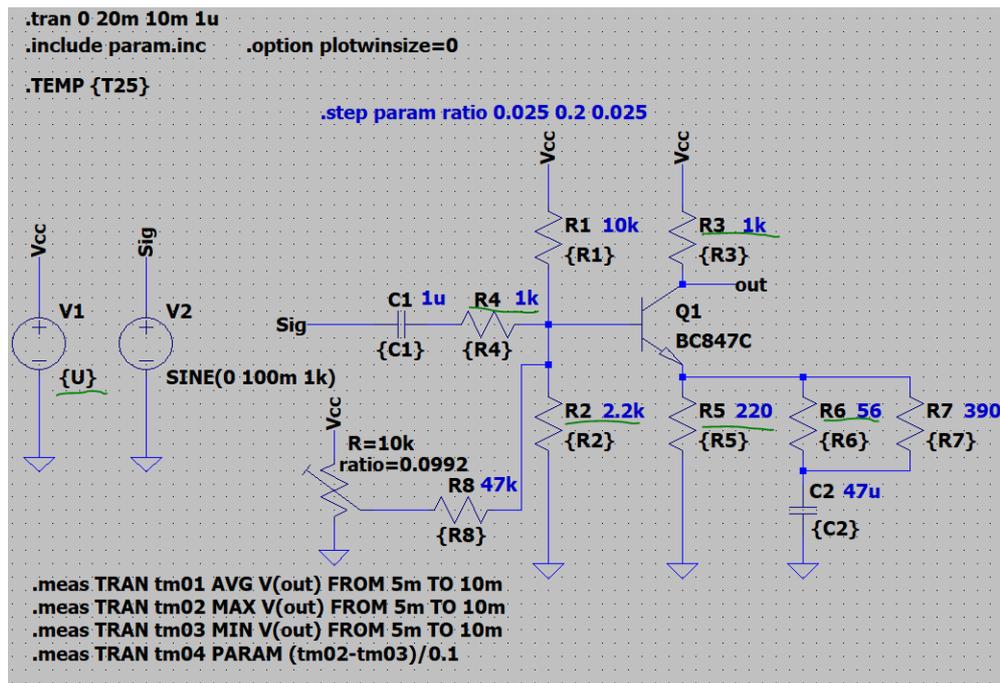
Auf der anderen Seite ist es unser Bestreben das der Arbeitspunkt in etwa $U_{ap} = (U_{vcc} + U_{re}) / 2$ entspricht.

Deshalb können wir im Excel Sheet den Messwert tm01 als XY Plot über die Spannung U darstellen inclusive dem angestrebten Arbeitspunkt.

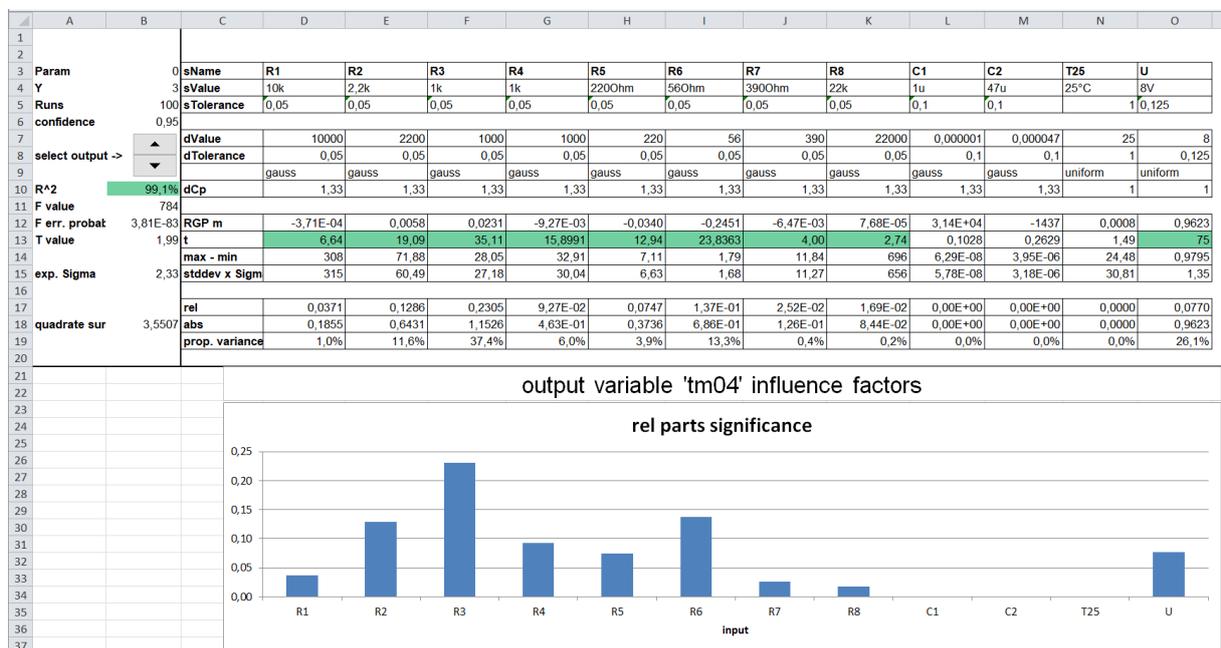


Wir sehen das unsere Variation des Arbeitspunktes unserem Ziel eher entgegen kommt. Wenn wir hier optimieren wollen müssen wir uns Gedanken über die Definition des Zielwertes machen. Da das Ausgangssignal aber unseren Erwartungen entspricht lassen wir es wie es ist.

Schauen wir auf die Verstärkung

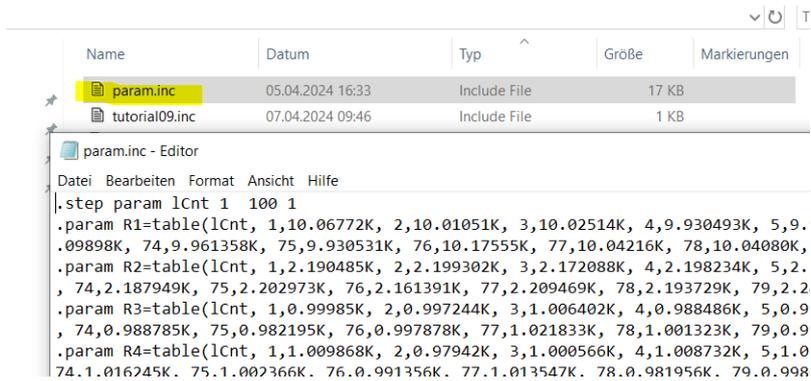


Die grün unterstrichenen Parameter im Plan bzw. die größten blauen Balken haben den größten Einfluss.

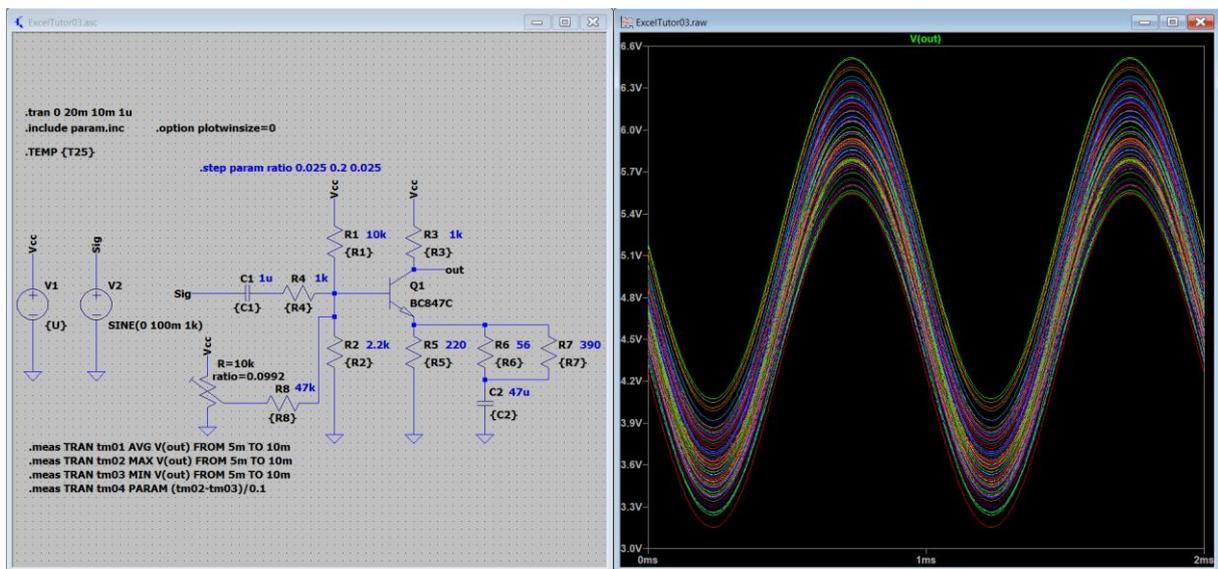


Wir sehen hier in diesem Bild den Einfluss der Widerstände auf die Verstärkung, die Kondensatoren haben keinen Einfluss da die Grenzfrequenz weit genug entfernt ist. Wenn wir die Variation der Verstärkung einschränken müssen wird die Toleranz der Bauteile mit den größten Balken reduzieren, das sind hier R3 und R6.

Nachdem wir die Simulation aus Excel gestartet haben existiert noch das Parameterfile für diese 100 Simulation.



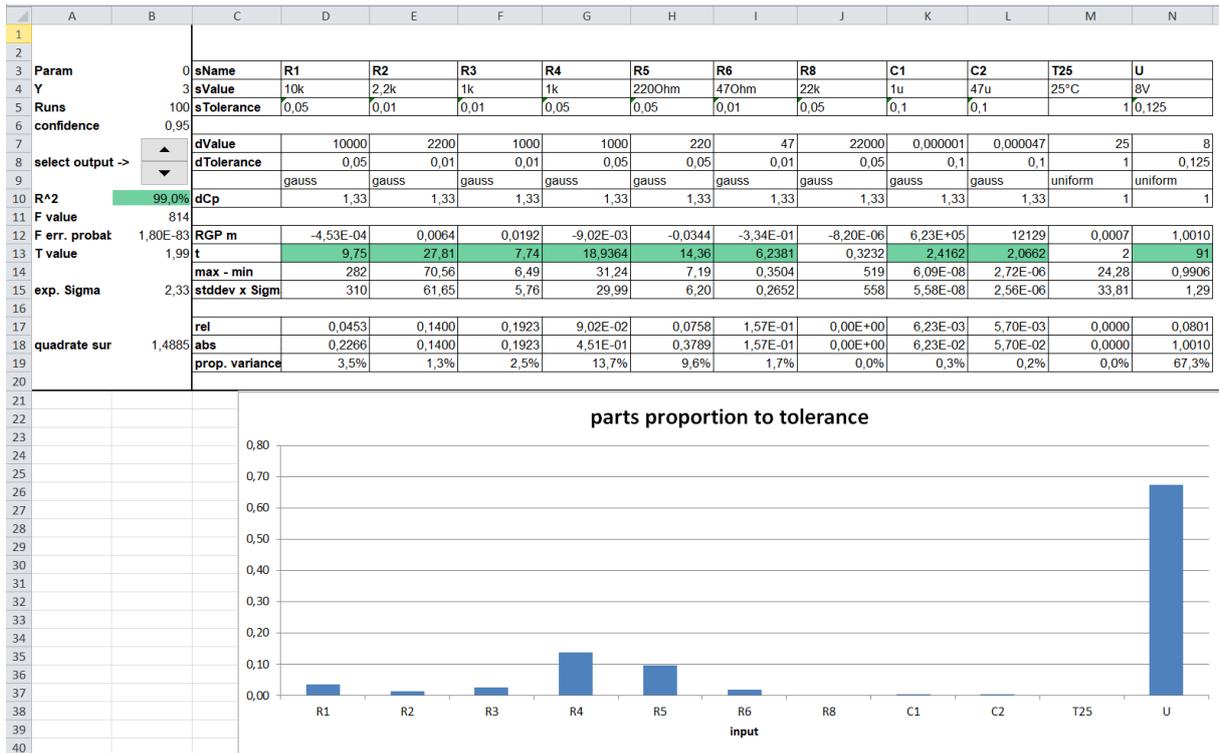
Wir können ItSpice starten und erhalten die grafische Ausgabe



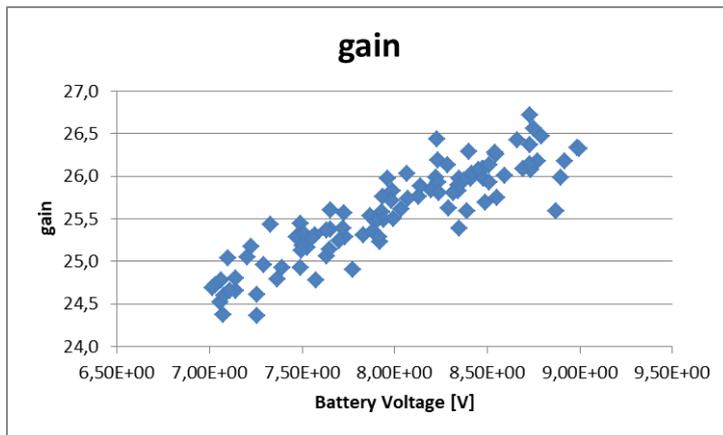
Den Zeitbereich habe ich auf 2ms begrenzt, wir sehen das alle Ausgangssignale einem Sinus entsprechen und genügend Reserve zur Betriebsspannungsgrenze aufweisen, deshalb ist eine Stabilisierung vom Arbeitspunkt nicht notwendig. Wir möchte aber die Verstärkung präziser haben.

Optimierungsgedanken zur Verstärkung, wir möchten für den Emitterwiderstand R6//R7 48.9 Ohm haben, um eine Verstärkung von genau 25 zu erreichen, aber unsere Toleranz reicht von 21 bis 27, also zu viel der Mühe, dann können wir auch 47 Ohm nehmen, dafür aber 1%. Das gleiche für R3, wir wählen wir auch 1k 1%. Das bedeutet wir haben 3 billige gegen 2 geringfügig teurere Widerstände ersetzt. Schauen wir uns das Ergebnis an.

sName	tm01	tm02	tm03	tm04
sNominalExpected				
dNominalExp	0	0	0	0
dLowSpecLi	0	0	0	0
dHighSpecLi	0	0	0	0
Capability Cp				
Average	4,59	5,85	3,29	25,58
SteDev	0,2224	0,2403	0,2118	0,5490
lo Limit 4.5S	3,59	4,77	2,34	23,11
up Limit 4.5S	5,59	6,93	4,24	28,05
AD Test	0,8911	0,6932	0,8707	0,0916



Es verbleibt ein wesentlicher Einfluss der Betriebsspannung, auch hier schauen wir uns im XY-Plot die Abhängigkeit der Verstärkung von der Betriebsspannung an



Das ist die Eigenschaft der Schaltung, mit weiteren Toleranzeinschränkungen der Bauteile werden wir keine Erfolge erzielen

Wir modifizieren die Schaltung in dem wir für die Temperturkompensation der Basis Emitter Spannung für die Basisvorspannung auch eine Diode einsetzen statt nur einem ohmschen Spannungsteiler. Desweiteren nutzen wir noch die Spannungsgegenkopplung, indem wir die Ausgangsspannung zurückführen. Die Nominalwerte der Widerstände kann man mittels Parametervariation „step“ in ItSpice ermitteln. Beim Spannungsteiler R2 zu R7 im Schlatplan kann man den Gesamtwiderstand mit `.step param n 1 10 0.5` variieren indem man $R2=\{R2*n\}$ und $R7=\{R7*n\}$ einsetzt, zudem kann man R2 oder R7 variieren für das Verhältnis. Im Log File sucht man sich dann die passende Kombination für den Arbeitspunkt und die Verstärkung.

Wer dabei nur in der E12 oder E24 Reihe suchen will definiert sich am besten eine Funktion, das kann man dann alles in ein Include File speichern und kann es so jederzeit einfach einfügen

.func E12(n,m) {table(n,1,1.2,1.2,3,1.5,4,1.8,5,2.2,6,2.7,7,3.3,8,3.9,9,4.7,10,5.6,11,6.8,12,8.2)*m}

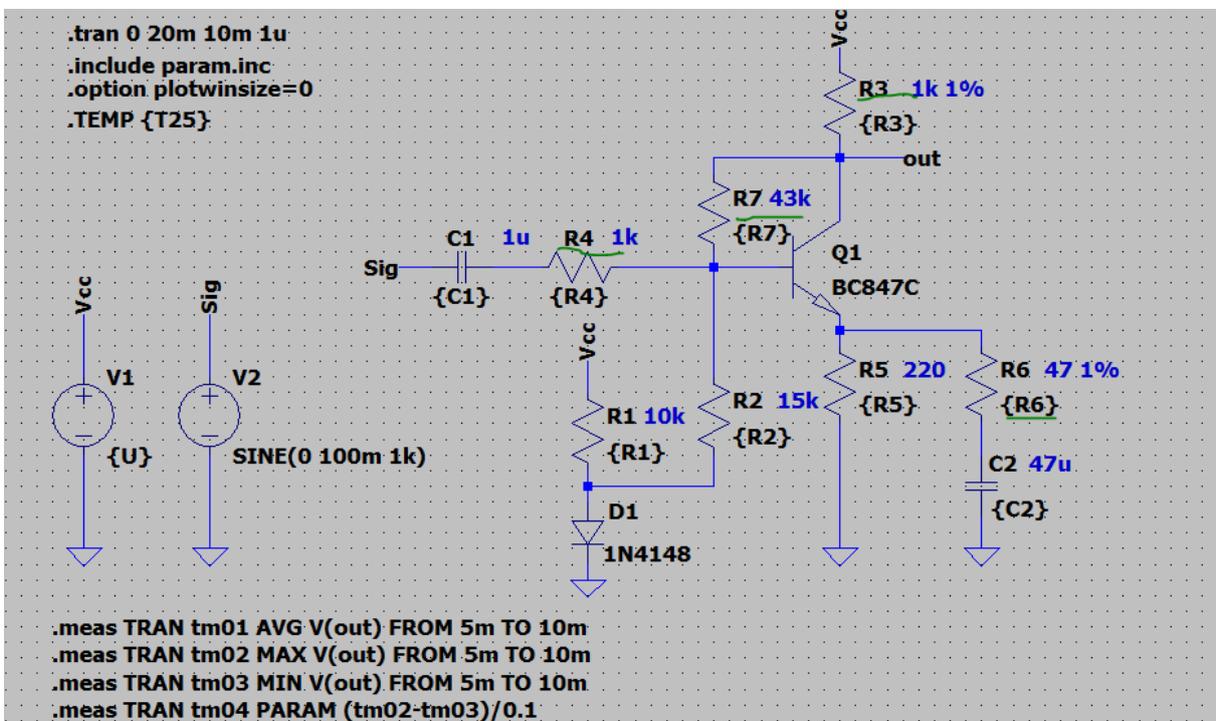
```
.step param i 1 12 1
R8
{E12(i,10k)}
```

Das Beispiel variiert 10KΩ bis 82KΩ

Für die E12 Reihe sollte man die Funktion mit n=-11 starten und bis 24 definieren, -11 wäre dann 0.1 und für 13 ist der Wert 10, dann kann man über drei Dekaden variieren.

Beispiel 4

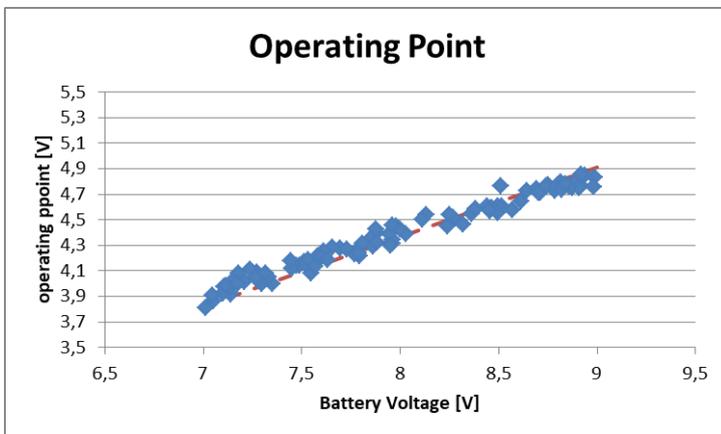
Das Ergebnis der Optimierung ist dieser Schaltplan



R2 mit R7 bilden neben der Basis Spannung auch eine AC Gegenkopplung, weshalb der Widerstand und das Spannungsteilerverhältnis angepasst werden muss um Arbeitspunkt und Verstärkung einzustellen. Die Ergebnisse sind hiermit deutlich besser, alle Widerstände haben noch 5% Toleranz.

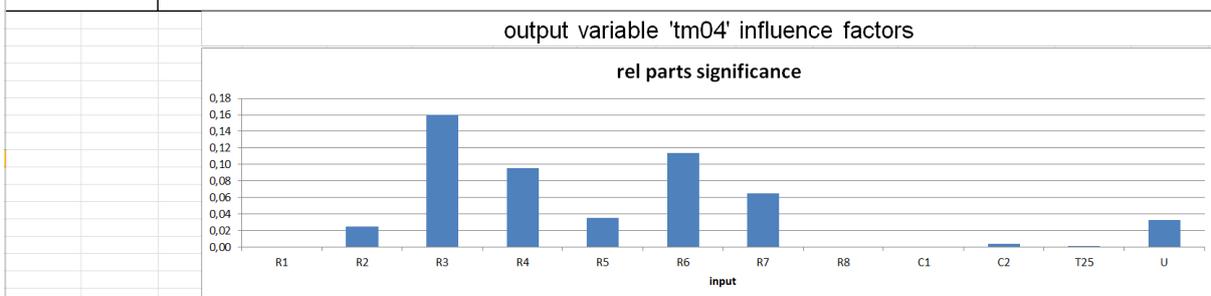
sName	tm01	tm02	tm03	tm04
sNominalExpected				
dNominalExp	0	0	0	0
dLowSpecLi	0	0	0	0
dHighSpecLi	0	0	0	0
Capability Cp				
Average	4,40	5,63	3,14	24,96
SteDev	0,2965	0,3108	0,2868	0,3584
lo Limit 4.5S	3,06	4,23	1,84	23,35
up Limit 4.5S	5,73	7,03	4,43	26,57
AD Test	0,0024	0,0012	0,0053	0,7779

Für den Arbeitspunkt haben wir einen Bereich von 1.7 Volt, aber im Mittel folgt dieser der Vorgabe, dass die Spannung zwischen Transistor CE Strecke und R3 gleich verteilt sind, das erklärt das es nicht Gauss verteilt ist. (AD Test Ergebnis <0.05)



Einflussfaktoren auf die Verstärkung

Param	sName	R1	R2	R3	R4	R5	R6	R7	R8	C1	C2	T25	U	
Y	sValue	10k	15k	1k	1k	220ohm	47Ohm	43k	22k	1u	47u	25°C	8V	
Runs	sTolerance	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,1	0,1		1 0,125	
confidence														
	dValue	10000	15000	1000	1000	220	47	43000	22000	0,000001	0,000047		25	8
select output ->	dTolerance	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,05	0,1	0,1		1	0,125
		gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss	gauss		uniform	uniform
R*2	dCp	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33		1	1
F value		2,66E+03												
F err. probat	RGP m	-3,05E-05	1,62E-04	0,0160	-9,58E-03	-0,0159	-2,41E-01	1,51E-04	1,39E-05	8,86E+04	7105	-0,0030	0,4066	
T value	t	1,90	14,62	80,61	53,66	19,65	64,7784	31,99	1,71	0,96	3,75	20,47	118	
exp. Sigma	max - min	341	584	27,15	31,06	7,01	1,33	1053	673	5,98E-08	2,70E-06	23,78	0,9964	
	stddev x Sigma	301	437	25,12	27,45	6,47	1,34	1047	615	5,32E-08	2,74E-06	34,26	1,40	
quadrate sur	rel	0,00E+00	0,0243	0,1596	9,58E-02	0,0351	1,13E-01	0,0650	0,00E+00	0,00E+00	3,34E-03	7,57E-04	0,0325	
	abs	0,0000	0,1216	0,7980	0,4792	0,1753	5,67E-01	0,3248	0,0000	0,00E+00	3,34E-02	0,0757	0,4066	
	prop. variance	0,0%	1,0%	42,1%	15,2%	2,0%	21,3%	7,0%	0,0%	0,0%	0,1%	0,4%	10,9%	

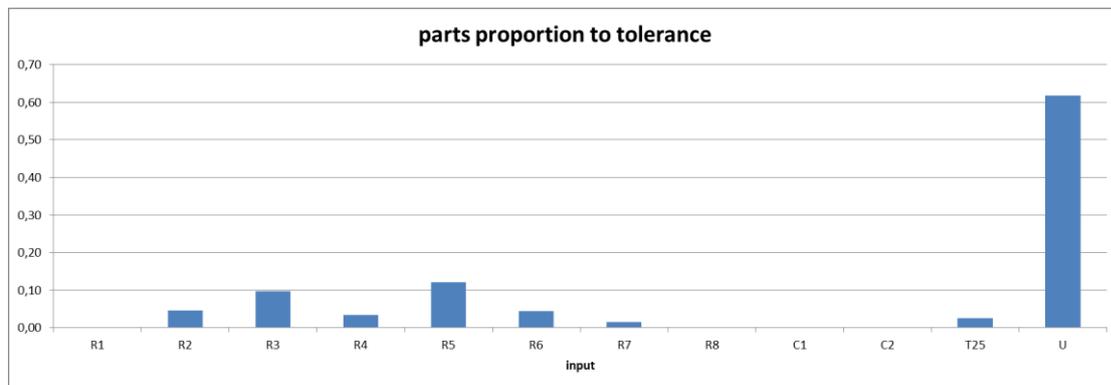


Lassen wir uns noch die Ergebnisse berechnen wenn wir die Widerstände R3 R4 R6 und R7 mit 1% auswählen

sName	tm01	tm02	tm03	tm04
sNominalExpected				
dNominalExp	0	0	0	0
dLowSpecLi	0	0	0	0
dHighSpecLi	0	0	0	0
Capability Cp				
Average	4,35	5,59	3,09	24,96
SteDev	0,2924	0,3063	0,2826	0,2591
lo Limit 4.5S	3,04	4,21	1,82	23,79
up Limit 4.5S	5,67	6,97	4,36	26,12
AD Test	0,00E+00	0,00E+00	0,00E+00	0,0183

Jetzt sind die Ergebnisse aber nichtmehr Normal (Gauss) verteilt. Das muss betrachtet werden.

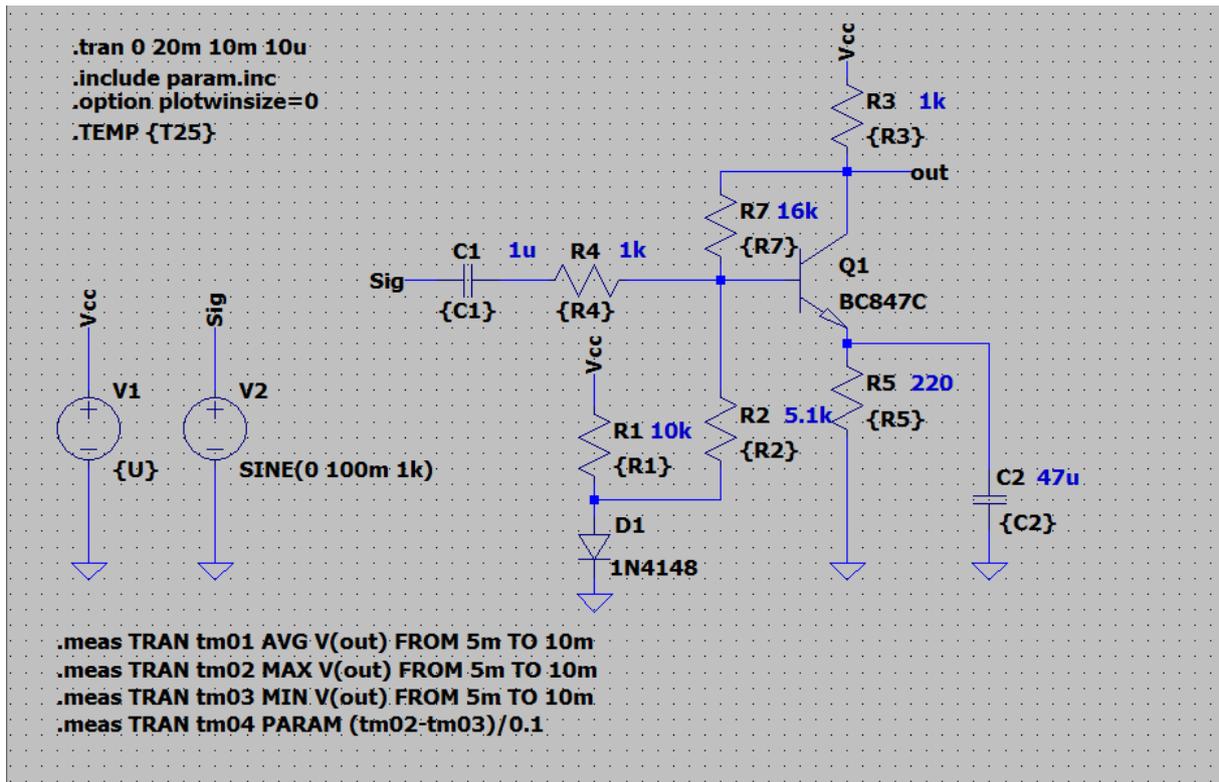
Durch gezielte Reduzierung der Toleranz und eine Anpassung der Schaltung haben wir jetzt die Verstärkung von 21-27 reduziert auf 23.8-26.1 und einen Arbeitspunkt der unserer Vorgabe folgt so dass eine optimale Ausnutzung der Betriebsspannung gegeben, indem der Arbeitspunkt mittig im nutzbaren Bereich liegt.



Als einzig verbliebener Einflussfaktor haben wir die Betriebsspannung, das bedeutet zwei Dinge:

- Die Betriebsspannung ist eine gleichmäßige Verteilung, wenn dies der einzige Einflussfaktor ist, wird das Ergebnis entsprechend eine gleiche verteilungsform aufweisen
- Ein weitere Optimierung ist nur mittels Designanpassung zu realisieren. Aber auch das ist eine wichtige Erkenntnis, das eine Eingrenzung der Toleranzen das Ergebnis nicht besser macht, nur die Schaltung wird teurer.

Als weitere Übung kann diese Schaltung untersucht werden



Die Ergebnisse sind ähnlich, Vorteil dieser Schaltung, es entfällt $R_6=47\Omega$, der in Reihe zu C_2 geschaltet war und es gibt nur noch zwei Widerstände mit wesentlichen Einfluss auf die Verstärkung statt vorher 4 Stück, was natürlich günstiger ist.

Ergebnisse mit der Beta Verteilung

Das ist jetzt schon sehr extrem und nur für die die Interesse daran haben, ich wollte das Thema aber nicht unerwähnt lassen

Excel automatisaton for Itspice

Versionsnummer 0.00.2 ItSpice 'path+file'

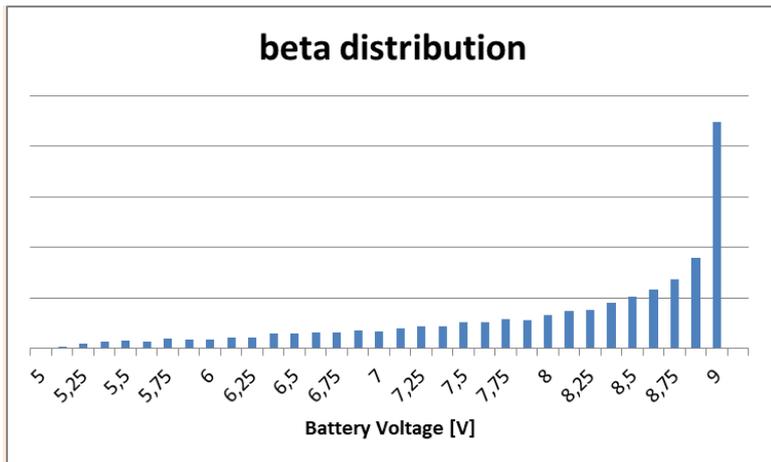
Basic simulation setup

2	12	num meas var	4	num sim runs	100	1	proj. Pat
							proj. Name ("name

faster simulation of small files

ItSpice input and output variables [the number of variables must correspond to the simulation setup]

ItSpice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	10k	5,0%		1,33				gauss			
2	R2	15k	5,0%		1,33				gauss			
3	R3	1k	5,0%		1,33				gauss			
4	R4	1k	5,0%		1,33				gauss			
5	R5	220Ohm	5,0%		1,33				gauss			
6	R6	470hm	5,0%		1,33				gauss			
7	R7	43k	5,0%		1,33				gauss			
8	R8	22k	5,0%		1,33				gauss			
9	C1	1u	10,0%		1,33				gauss			
10	C2	47u	10,0%		1,33				gauss			
11	T25	25°C	100,0%		1				uniform			
12	U	8V			1				beta		5	9
13												

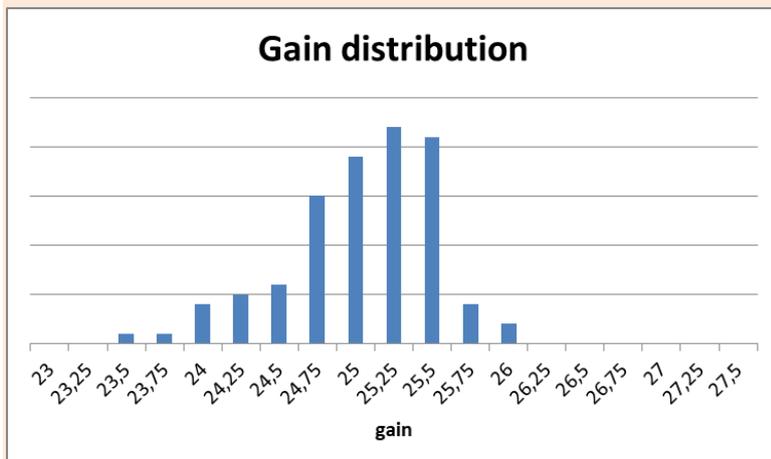


Die Batterie startet mit 9V und je nach Nutzerverhalten wird sie nach einer gewissen Zeit gegen eine neue Batterie ausgetauscht, wobei hier $\frac{1}{4} > 7.5V$ liegen. Alle Widerstände haben 5%.

Ergebnis für 100 Simulationsdurchläufe

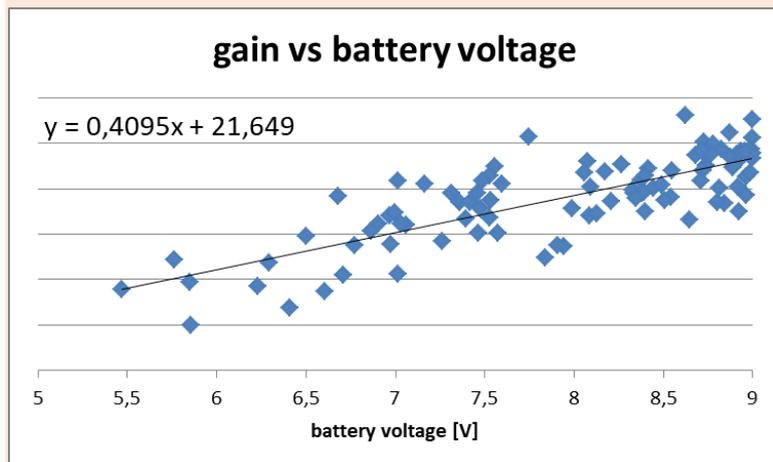
sName	tm01	tm02	tm03	tm04
sNominalExpected				
dNominalExp	0	0	0	0
dLowSpecLi	0	0	0	0
dHighSpecLi	0	0	0	0
Capability Cp				
Average	4,39	5,62	3,13	24,92
SteDev	0,4516	0,4736	0,4374	0,4701
lo Limit 4.5S	2,35	3,49	1,16	22,80
up Limit 4.5S	6,42	7,75	5,10	27,03
AD Test	0,00E+00	0,00E+00	0,00E+00	0,0033

Das sieht deutlich schlechter aus, das liegt daran das jetzt einzig die Spannung einen wesentlichen Einfluss hat, entsprechend folgt auch die Verstärkung dieser beta Verteilung



In diesem Fall ist wichtig, sich die Verteilung genauer anzuschauen, die Werte sehen dort besser aus als die statistische Rechnung zeigt, das liegt daran, dass diese nur für Gauss Verteilungen gültig sind. Bei einem Wert vom AD Test = 0, sind diese Werte ungültig.

Und hier noch mal die Abhängigkeit zur Betriebsspannung



Man kann in diesem Fall in einer weiteren Spalte die aufgrund der Betriebsspannung erwartete Verstärkung darstellen und dann die Abweichung zu diesem Erwartungswert berechnen, das sieht dann so aus

S	T	U	V	W	X	Y
sName		tm01	tm02	tm03	tm04	
sNominalExpected						
dNominalExp		0	0	0	0	
dLowSpecLi		0	0	0	0	
dHighSpecLi		0	0	0	0	
Capability Cp						
Average		4,39	5,62	3,13	24,92	0,00
SteDev		0,4516	0,4736	0,4374	0,4701	0,2852
lo Limit 4.5S		2,35	3,49	1,16	22,80	-1,28
up Limit 4.5S		6,42	7,75	5,10	27,03	1,28
AD Test		0,00E+00	0,00E+00	0,00E+00	0,0033	0,3382
		4,78E+00	6,00E+00	3,53519	24,6598	0,52968122
		4,85E+00	6,10E+00	3,56928	25,3367	-0,002281
		4,20E+00	5,44E+00	2,9306	25,049	-0,28967775
		4,77E+00	6,03E+00	3,48873	25,3658	-0,16289571

In der grün unterstrichenen Zelle habe ich die Formel unten eingetragen, und dann für bis zum Ende der Daten nach unten kopiert. In den gelb markiert Zellen habe ich die Formel links daneben rüber kopiert und den Datenbereich angepasst (F2 und dann den Rahmen nach rechts ziehen)

Wir haben jetzt ein Verstärkung die um ± 1.3 variiert sofern die Betriebsspannung stabil ist. Hier zeigt der AD Test auch das die Daten Normalverteilt sind, somit unsere statistischen Rechnung Gültigkeit haben. Für den Mittelwert können wir die Abhängigkeit der Verstärkung von der Spannung eintragen, siehe XY Plot, $gain=0.4095 \times U + 21.649$

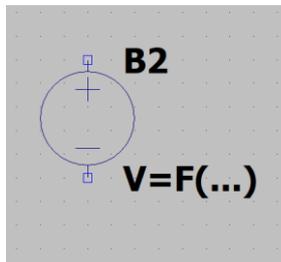
519	<u>$=0,4095 * O21 + 21,649 - X21$</u>	
928	25,3367	23,

Beispiel 5

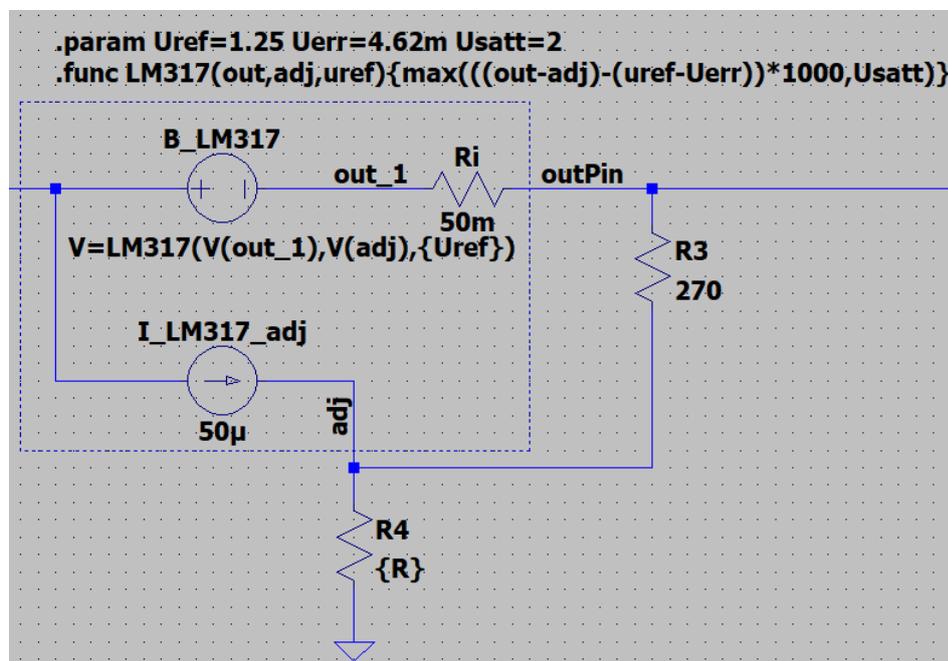
Jetzt noch ein paar weitere Schaltungen deren Analyse interessant ist. Als nächstes probieren wir mal ein einfaches günstiges Messgerät zu entwickeln, ein Ohm-Meter für niederohmige Widerstände. Als Basis starten wir mit einer bekannten Schaltung die vielfach im Netz zu finden ist. Statt einer fertigen Anzeige benutzen wir aber einen Arduino oder STM Bluepill, das bedeutet wir müssen die Signale verstärken.

Wir werden nicht alle Bauteile 1:1 aus der ItSpice Bibliothek verwenden können, weil diese keine Eigenschaften haben, mit denen wir die Toleranzen einfach variieren können.

Als erstes bauen wir uns einen Ersatz für den LM317, wir starten mit einer Spannungsquelle v



Die Basis ist ein mathematisches Modell für die gesteuerte Spannungsquelle, wir definieren dazu die regelabweichung, Basis Istwert-Sollwert, mit Istwert = $V(\text{out}) - V(\text{adj})$, dieser soll 1.25V betragen, als Verstärkung wählen wir 1000 und als Sättigungsspannung 2V.

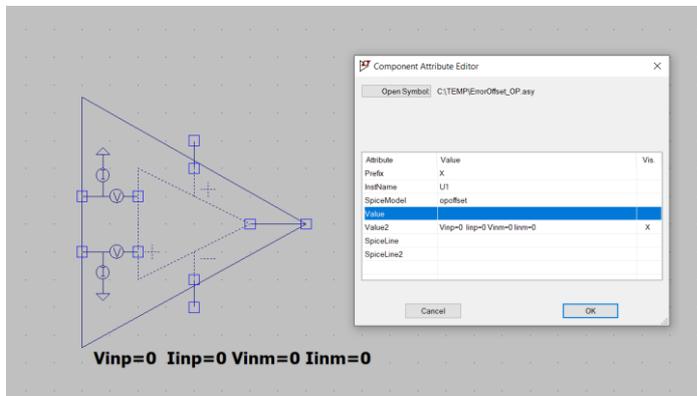


Das ist ein „idealer“ LM317, R_i bewirkt die nicht ideale Lastausregelung, diesen könnten wir in der Simulation ebenfalls variieren. Die 50m repräsentieren den Lastregelfehler von 1.5% und für U_{ref} gilt bei 25°C $\pm 4\%$ zusätzlich $\pm 0.7\%$ über Temperaturbereich.

I_LM317_adj ist der Strom aus dem Adj Pin, den wir berücksichtigen wollen, typ 50uA und max 100uA.

Als nächstes wählen wir einen Operationsverstärker aus um die Spannung die über das Testobjekt abfällt zu messen. Wir bleiben im low budget bereichn und wählen einen TL072.

Hier wollen wir aber die Eingangsoffsetspannung und den Eingangsoffsetstrom berücksichtigen, das geht bei dem Modell in der Form aber nicht. Aus diesem Grund wird das Modell „gekapselt“, ähnlich dem Temperatur Modell für Mosfets, (das Modell ist in der zip Datei)



Hier können wir für die statistische Simulation diese Offsetwerte für Strom und Spannung eingeben und platzieren innerhalb den OP aus der ItSpice Bibliothek

Das Datenblatt gibt für Input Offset Voltage typ $\pm 1\text{mV}$ an und max $\pm 5\text{mV}$, für den Offset-Strom sind es typ 0 und max 300pA.

Da zu diesem Zeitpunkt noch keine Verteilungsform für die Spannung und Strom mit zufälligen Vorzeichen zur Verfügung steht wählen wir hier die die für den Offset ungünstige Variante, das der ein Eingang immer positive Offset Spannung hat und der andere Eingang immer negative Offsetspannung.

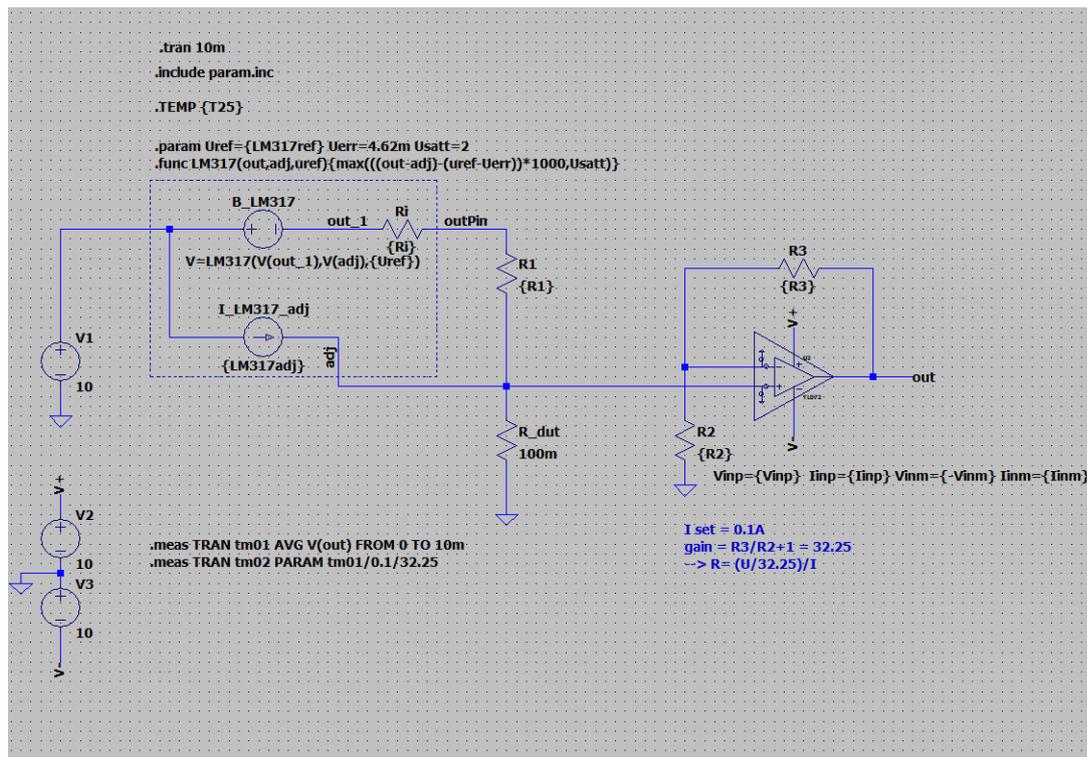
Anm: Der Einfachheit halber habe ich in den Simulationen I_{Bias} weggelassen und nur I_{Offset} berücksichtigt, ganz am Ende sehen wir, dass wegen der Optimierung diese Werte kaum noch Einfluss haben. Aber im Anhang „Übersicht Verteilungen für Bauteile“ ist die korrekte Form dargestellt.

So sieht die ausgefüllte Excel Tabelle aus

ItSpice input and output variables [the number of variables must correspond to the simulation setup]												
Itspice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	12.50hm	5,0%		1,33				gauss			
2	R2	32k	5,0%		1,33				gauss			
3	R3	1Meg	1,0%		1,33				gauss			
4	Vinp	1m	5m		1,33				log			
5	Vinm	1m	5m		1,33				log			
6	Iinp		0 300p		1,33				gauss			
7	Iinm		0 300p		1,33				gauss			
8	LM317ref	1.25V	4,7%		1,33				gauss			
9	Ri	37.5m	50,0%		1				uniform			
10	LM317adj	75u	33,0%		1				uniform			
11	T25		25		1				uniform			
12												
13												
14												

Hier hat sich ein Fehler eingeschlichen, 32k ist falsch, hätte 33k sein müssen, aber alles mit 32k gerechnet, will ich jetzt nicht neu machen

Und der dazugehörige Schaltplan



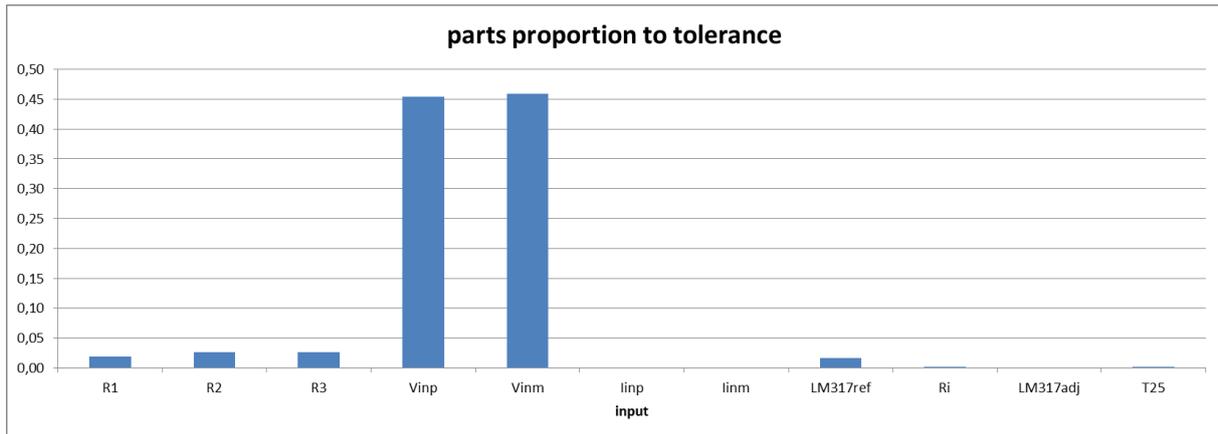
Der erste Messbefehl ermittelt die Spannung am OP Ausgang und der zweite Messbefehl berechnet den Widerstandswert unter Berücksichtigung des definierten Messstromes und des nominellen Verstärkungsfaktors von 32.5

$$R = \frac{U}{I} \text{ with } U_{out} = U_R \cdot gain \rightarrow U_R = \frac{U_{out}}{gain} \rightarrow R = \frac{U_{out}}{I \cdot gain}$$

sName	tm01	tm02
sNominalExpected		
dNominalExp	0	0
dLowSpecLim	0	0
dHighSpecLim	0	0
Capability Cp		
Average	0,3952	0,1225
SteDev	0,0262	0,0081
lo Limit 4.5S	0,2772	0,0860
up Limit 4.5S	0,5131	0,1591
AD Test	0,00	0,00

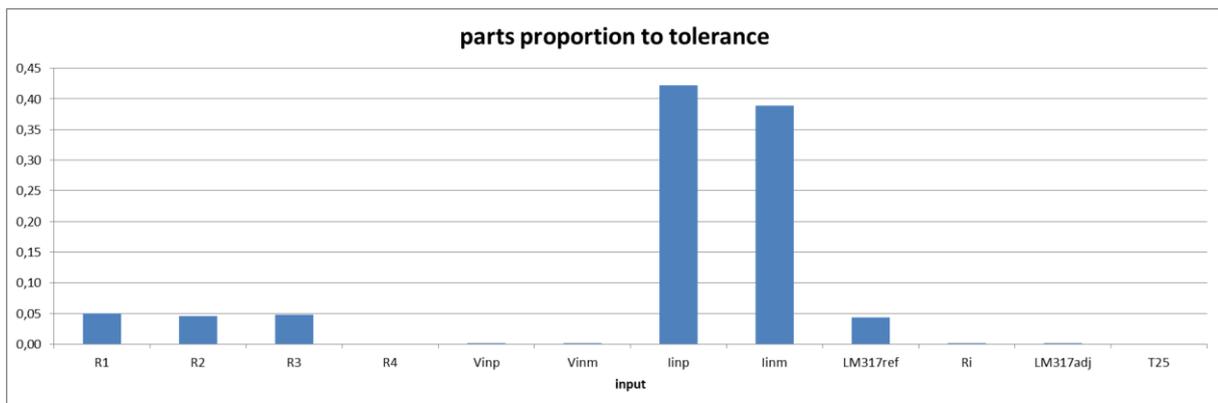
Diese Ergebnisse sind sehr schlecht, der Messfehler am beträgt -14% und bis zu +50%, der nächste Plot zeigt uns die Ursachen?

Param	0	sName	R1	R2	R3	Vinp	Vinn	linp	linm	LM317ref	Ri	LM317adj	T25
Y	0	sValue	12.50hm	32k	1Meg	1m	1m		0	1.25V	37.5m	75u	25
Runs	100	sTolerance	0,05	0,05	0,05	5m	5m	300p	300p	0,047	0,5	0,33	1
confidence	0,95												
select output ->		dValue	12,5	32000	1000000	0,001	0,001	0	0	1,25	0,0375	0,000075	25
		dTolerance	0,05	0,05	0,05	0,005	0,005	3E-10	3E-10	0,047	0,5	0,33	1
			gauss	gauss	gauss	log	log	gauss	gauss	gauss	uniform	uniform	uniform
R^2	100,0%	dCp	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1	1	1
F value	2,53E+04												
F err. probat	6,22E-149	RGP m	-0,0257	-1,21E-05	3,89E-07	32,43	32,48	-3,16E+05	2,67E+05	0,2613	-0,0310	1,03	1,81E-05
T value	1,99	t	80,79	94,48	79,73	294	320	0,4423	0,3197	71,07	5,95	0,2708	4,96
exp. Sigma	2,33	max - min	0,4263	1246	2,71E+04	0,0012	0,0013	1,70E-10	1,47E-10	0,0359	0,0184	2,39E-05	23,78
		stddev x Sigma	0,3767	933	2,51E+04	0,0011	0,0013	1,71E-10	1,46E-10	0,0329	0,0233	3,39E-05	34,26
quadrate sur	0,0144	rel	0,0032	0,0039	0,0039	3,24E-04	3,25E-04	0,00E+00	0,00E+00	0,0033	1,16E-05	0,00E+00	4,53E-06
		abs	0,0160	0,0193	0,0195	0,0811	0,9812	0,00E+00	0,00E+00	0,0154	5,82E-04	0,00E+00	4,53E-04
		prop. variance	1,8%	2,6%	2,6%	45,6%	45,7%	0,0%	0,0%	1,6%	0,0%	0,0%	0,0%



Die Offset Spannung des Operationsverstärkers ist die wesentliche Ursache. Die gleiche Ausrichtung der Polarität halbiert die Streuung und verschiebt den Offset, das ist aber erst beim Feintuning der Schaltung zu beachten

Wir untersuchen die gleiche Schaltung mit einem OP mit geringerer Offsetspannung, wir wählen einen OP mit Uofs typ 20uV 100uV max und Iofs 6n typ 90nA max, (wir lassen erst mal den tl072 drin und simulieren ihn besser als er ist).



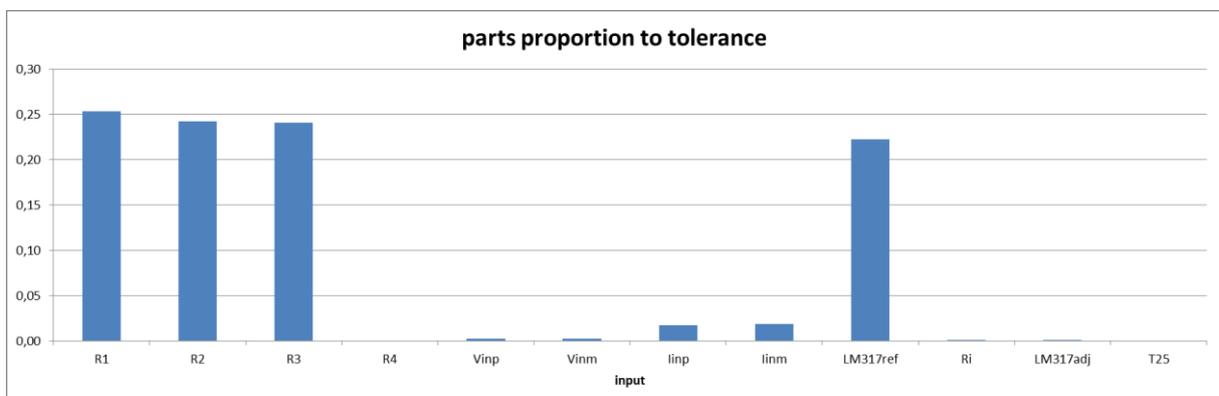
Jetzt verursachen die Eingangsströme den wesentlichen Anteil am Messfehler, wir haben aber die Option Widerstände mit niedrigeren Werte auszuwählen, weil beim Eingangsoffsetstrom ist es die Kombination aus Widerstandswert und Eingangsstrom welche den Offsetfehler am Ausgangs bewirkt.

Itspice input and output variables [the number of variables must correspond to the simulation setup]												
Itspice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	12.5Ohm	5,0%	1,33					gauss			
2	R2	3,2k	5,0%	1,33					gauss			
3	R3	100k	5,0%	1,33					gauss			
4	R4	3,2k	5,0%	1,33					log			
5	Vinp	20u	100u	1,33					log			
6	Vinm	20u	100u	1,33					log			
7	linp	6n	90n	1,33					log			
8	linm	6n	90n	1,33					log			
9	LM317ref	1.25V	4,7%	1,33					gauss			
10	Ri	37.5m	50,0%	1					uniform			
11	LM317adj	75u	33,0%	1					uniform			
12	T25	25	100,0%	1					uniform			
13												
14												
15												
16												

sName	tm01	tm02
sNominalExpected		
dNominalExp	0	0
dLowSpecLir	0	0
dHighSpecLir	0	0
Capability Cp		
Average	0,3238	0,1004
SteDev	0,0078	0,0024
lo Limit 4.5S	0,2888	0,0895
up Limit 4.5S	0,3588	0,1112
AD Test	0,09	0,09

Jetzt haben wir ein Ergebnis mit ca. ±10% Messfehler und der Blick auf die Ursachen zeigt uns jetzt den Einfluss der Widerstandstoleranzen, und der Referenzspannung des LM317

Param	0	sName	R1	R2	R3	R4	Vinp	Vinm	linp	linm	LM317ref	Ri	LM317adj	T25
Y	0	sValue	12,5Ohm	3,2k	100k	3,2k	20u	20u	6n	6n	1,25V	37,5m	75u	25
Runs	100	sTolerance	0,05	0,05	0,05	0,05	100u	100u	90n	90n	0,047	0,5	0,33	1
confidence	0,95													
select output ->		dValue	12,5	3200	100000	3200	0,00002	0,00002	0,000000006	0,000000006	1,25	0,0375	0,000075	25
		dTolerance	0,05	0,05	0,05	0,05	0,0001	0,0001	0,000000009	0,000000009	0,047	0,5	0,33	1
R*2	100,0%	dCp	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1,33	1	1
F value	3,67E+04													
F err. probat	1,36E-155	RGP m	-0,0258	-9,87E-05	3,15E-06	3,09E-04	32,74	-30,50	-9,42E+04	9,71E+04	0,2573	-0,0255	2,73	-1,24E-06
T value	1,99	t	300	320	296	3,0061	25,91	25,71	65,00	46,64	339	22,74	3,25	1,44
exp. Sigma	2,33	max - min	0,3135	95,05	2803	0,0296	2,26E-05	2,96E-05	4,10E-08	1,54E-08	0,0468	0,0185	2,47E-05	24,95
		stddev x Sigma	0,3379	95,80	2626	0,0278	2,22E-05	2,45E-05	2,01E-08	1,35E-08	0,0373	0,0253	3,36E-05	33,01
quadrat sur	0,0010	rel	0,0032	0,0032	0,0031	0,00E+00	6,55E-06	6,10E-06	5,65E-06	5,83E-06	0,0032	9,54E-06	2,05E-06	0,00E+00
		abs	0,0161	0,0158	0,0157	0,00E+00	0,0016	0,0015	0,0042	0,0044	0,0151	4,77E-04	6,76E-05	0,00E+00
		prop. variance	25,3%	24,3%	24,1%	0,0%	0,3%	0,2%	1,7%	1,9%	22,2%	0,0%	0,0%	0,0%



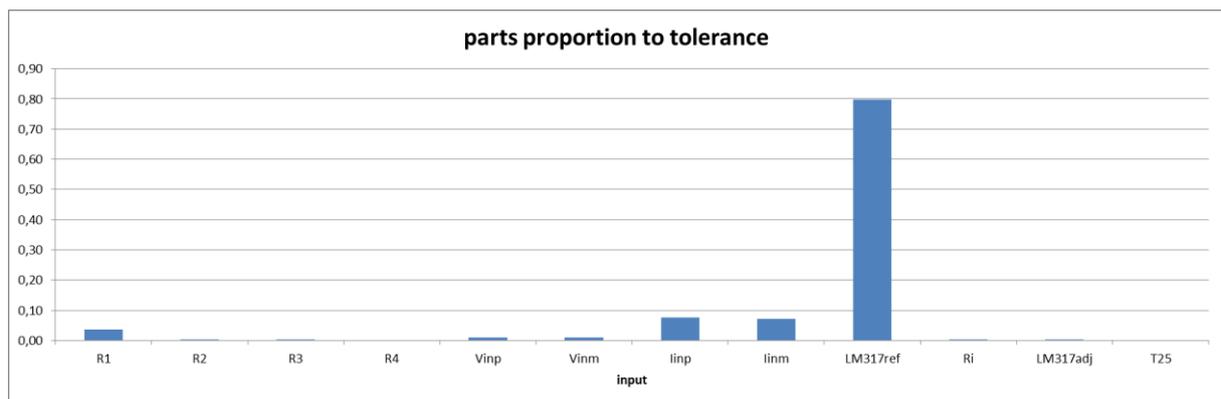
Wir reduzieren die Toleranzen, bei dem Widerstandspaar R3/R3, das die Verstärkung definiert reicht es eigentlich aus, das Verhältnis genau einzustellen, wie das gehen kann habe ich in einem anderen Tutorial beschrieben, wir definieren dies in der Simulation mit jeweils 0.1%. Für den Absolut Wert von R1 definieren wir eine Toleranz von 1%.

Mit dieser Anpassung erhalten wir jetzt diese Ergebnisse

ItSpice input and output variables [the number of variables must correspond to the simulation setup]												
ItSpice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	12.50hm	1,0%	1,33					gauss			
2	R2	3.2k	0,1%	1,33					gauss			
3	R3	100k	0,1%	1,33					gauss			
4	R4	3.2k	5,0%	1,33					log			
5	Vinp	20u	100u	1,33					log			
6	Vinm	20u	100u	1,33					log			
7	linp	6n	90n	1,33					log			
8	linm	6n	90n	1,33					log			
9	LM317ref	1.25V	4,7%	1,33					gauss			
10	Ri	37.5m	50,0%	1					uniform			
11	LM317adj	75u	33,0%	1					uniform			
12	T25	25	100,0%	1					uniform			
13												
14												

sName	tm01	tm02
sNominalExpected		
dNominalExp	0	0
dLowSpecLir	0	0
dHighSpecLi	0	0
Capability Cp		
Average	0,3231	0,1002
SteDev	0,0041	0,0013
lo Limit 4.5S	0,3047	0,0945
up Limit 4.5S	0,3415	0,1059
AD Test	0,46	0,46

Der Messfehler beträgt jetzt nur noch $\pm 6\%$, was können wir jetzt machen um noch besser zu werden. Die Einflussgröße ist jetzt fast nur noch der LM317, dieser weißt eine Toleranz von $\pm 4.7\%$ auf. Wir können jetzt eine genauere Referenz wählen. Ein andere Alternative ist es diese Spannung zu messen.



Wenn wir jetzt die Schaltung soweit abändern das wir die LM317 Ref Spannung ebenfalls messen , dann benötigen wir einen Differenzverstärker, als Verstärkungsfaktor wählen wir den Faktor 2, damit liegen wir im oberen Drittel eines AD Wandler der 3.3V als max. Eingangsspannung verarbeitet, wie beim STM32 bluepill.

An der Stelle möchte ich den Punkt Referenzspannung des STM32 analytisch betrachten und nicht die AD Wandler als Spice Modell erstellen.

Der Messstrom I ist abhängig von U_{LM317} , wenn wir messen repräsentiert das Ergebnis $U_{ADLM317}$ einen Messwert relativ U_{Ref} ,

$$U_{ADref} = \frac{digitsU}{4096} \cdot U_{Ref}$$

daraus folgt

$$I_{meas} = \frac{\frac{digitsI}{4096} \cdot U_{Ref}}{R_1}$$

für die Spannung am Testobjekt gilt

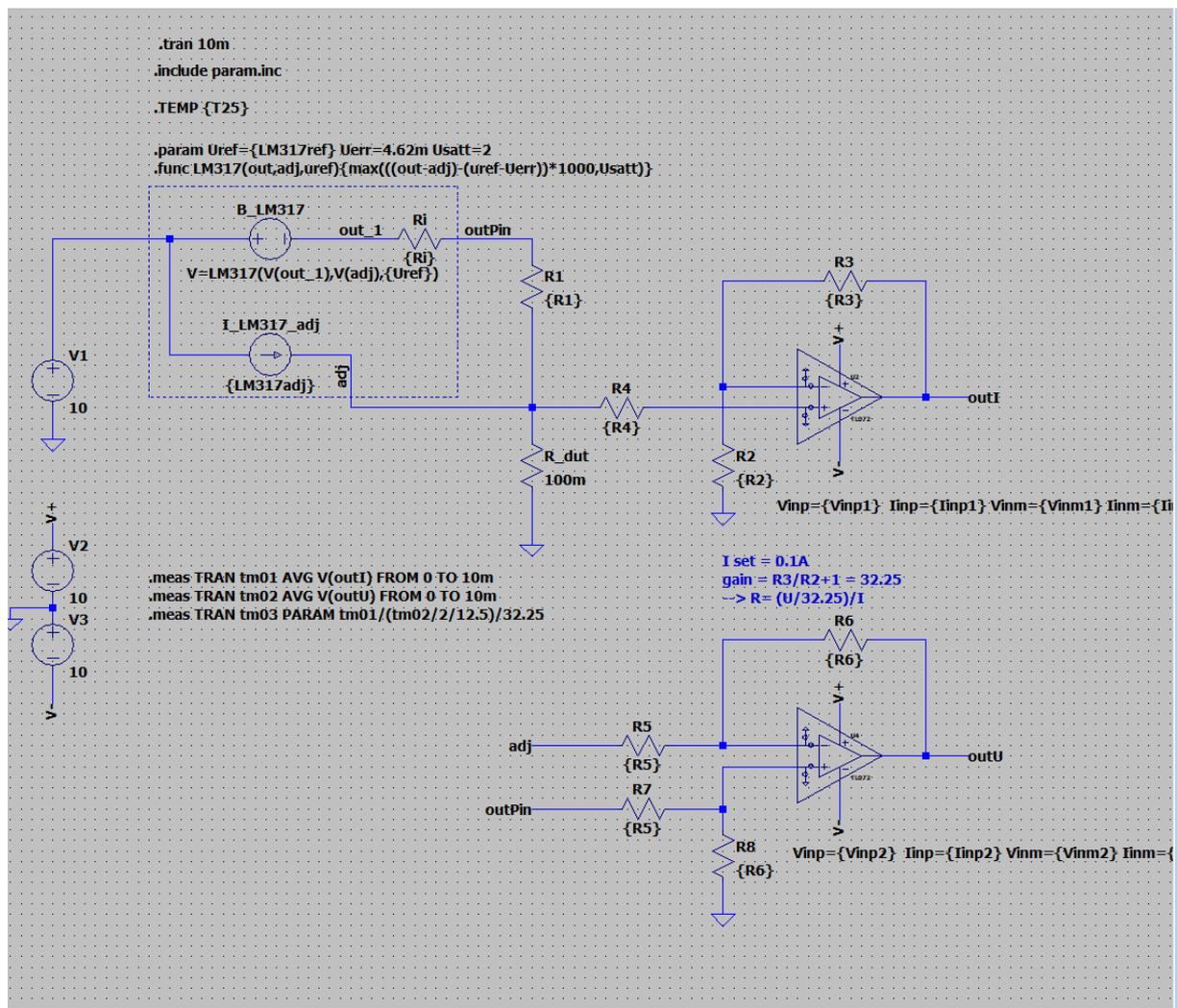
$$U_{DUT} = \frac{digitsU}{4096} \cdot U_{Ref}$$

und für den Widerstand gilt

$$R = \frac{U}{I} = \frac{\frac{digitsU}{4096} \cdot U_{Ref}}{\frac{\frac{digitsI}{4096} \cdot U_{Ref}}{R_1}} = \frac{digitsU}{digitsI} \cdot R_1$$

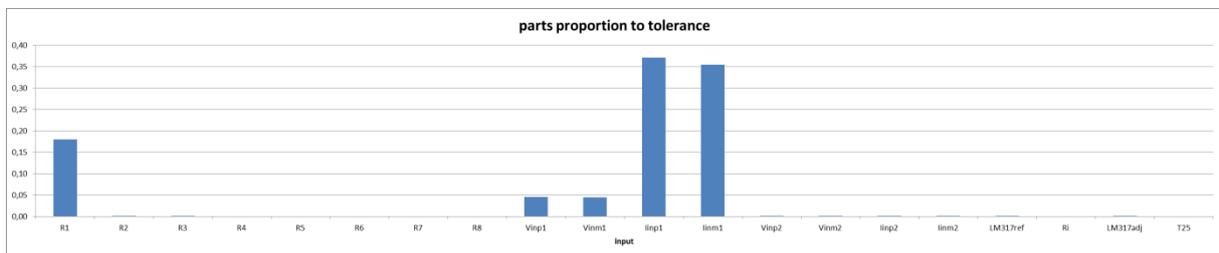
oder einfach ausgedrückt, fließt durch beide Widerstände der gleiche Strom, dann teilt sich die Spannung relativ zu den Widerstandswerten auf und wenn R_1 bekannt ist kann man R_{DUT} berechnen, darauf hat die absolute Genauigkeit der Spannungsmessung keinen Einfluss.

Da ist nun der nächste Schritt unserer Toleranzbetrachtung, die Spannung des LM317 wird gemessen.



sName	tm01	tm02	
sNominalExpected			
dNominalExp	0	0	0
dLowSpecLi	0	0	0
dHighSpecLi	0	0	0
Capability Cp			
Average	0,3230	2,50	0,1003
SteDev	0,0041	0,0307	3,87E-04
lo Limit 4.5S	0,3044	2,36	0,0985
up Limit 4.5S	0,3416	2,63	0,1020
AD Test	0,28	0,37	0,32

Das Ergebnis ist jetzt mit $\pm 2\%$ schon wirklich gut, wir betrachten jetzt noch einmal die Einflussfaktoren.

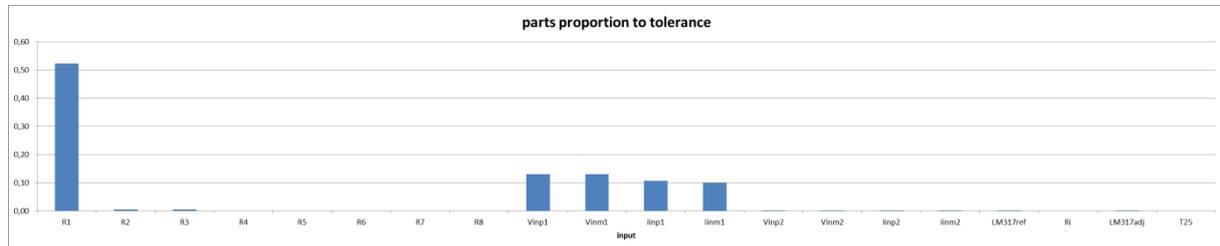


Es ist nochmals der Eingangsoffsetstrom der einen Einfluss hat sowie der Widerstand der den Strom mit definiert, wie wir in der analytischen Rechnung gesehen haben. Deswegen werden wir im letzten Schritt dieser Toleranzbetrachtung nochmals die Schaltung niederohmiger auslegen.

Itspice input and output variables [the number of variables must correspond to the simulation setup]												
Itspice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	12.50hm	1,0%	1,33					gauss			
2	R2	1k	0,1%	1,33					gauss			
3	R3	33k	0,1%	1,33					gauss			
4	R4	1k	5,0%	1,33					log			
5	R5	10k	0,1%	1,33					log			
6	R6	20k	0,1%	1,33					log			
7	R7	10k	0,1%	1,33					log			
8	R8	20k	0,1%	1,33					log			
9	Vinp1	20u	100u	1,33					log			
10	Vinm1	20u	100u	1,33					log			
11	linp1	6n	90n	1,33					log			
12	linm1	6n	90n	1,33					log			
13	Vinp2	20u	100u	1,33					log			
14	Vinm2	20u	100u	1,33					log			
15	linp2	6n	90n	1,33					log			
16	linm2	6n	90n	1,33					log			
17	LM317ref	1.25V	4,7%	1,33					gauss			
18	Ri	37.5m	50,0%	1					uniform			
19	LM317adj	75u	33,0%	1					uniform			
20	T25		25	100,0%	1				uniform			
21												

sName	tm01	tm02	
sNominalExpected			
dNominalExp	0	0	0
dLowSpecLi	0	0	0
dHighSpecLi	0	0	0
Capability Cp			
Average	0,3413	2,50	0,1057
SteDev	0,0045	0,0307	2,90E-04
lo Limit 4.5S	0,3212	2,36	0,1044
up Limit 4.5S	0,3613	2,64	0,1070
AD Test	0,26	0,25	0,38

Die Streuung ist mit $\pm 1.3\%$ wirklich sehr gut und es ist jetzt im wesentlichen nur noch unser Widerstand R1. Man könnten hier für diesen Wert mehrere niederohmige Messwiderstände parallel schalten um den Strom und die Eigenerwärmung in den Griff zu bekommen und damit die Toleranz nochmals reduzieren. Wir würden damit wahrscheinlich unterhalb 1% Fehler kommen



Mit dieser Basis schauen wir auch noch mal auf die Ursache für die absolute Abweichung des Messwertes. Wir messen 5% zu hoch.

Das hat zwei Gründe, zum einen messen wir den Widerstand mit dem Strom durch R1 plus dem Strom Adj des LM317, zum anderen haben wir für Offsetströme und Offsetspannungen nur positive Werte.

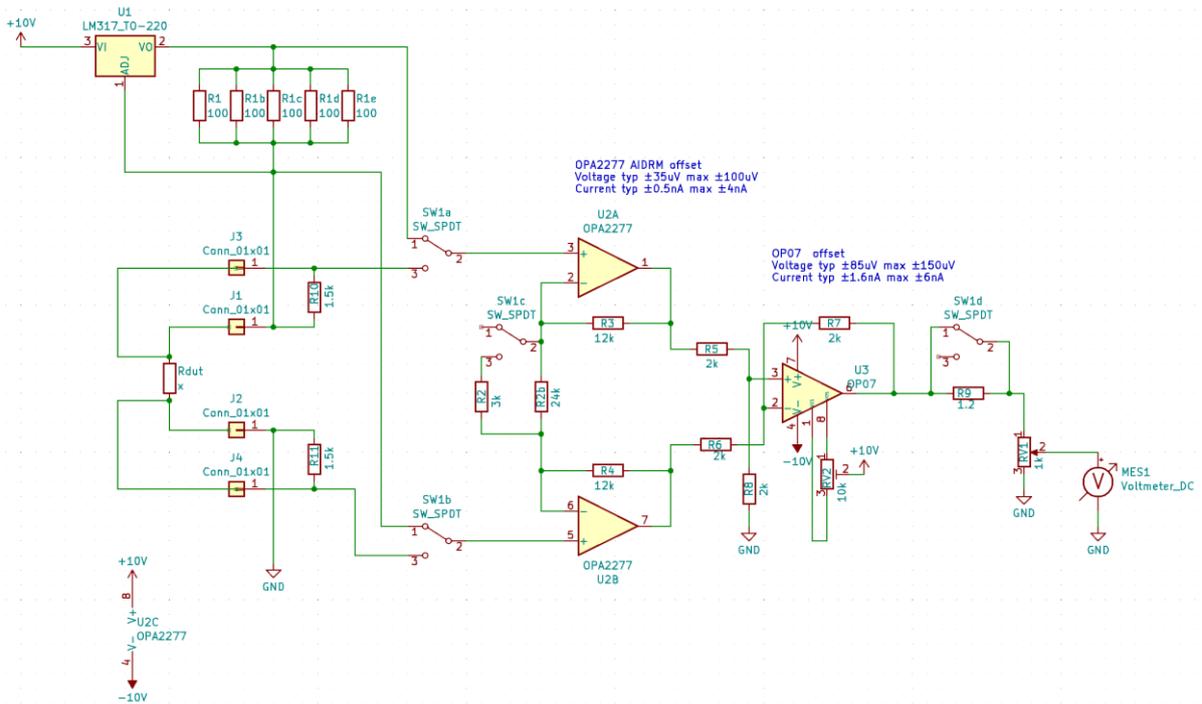
Um dies genauer zu betrachten habe ich jetzt noch eine weitere Verteilung für Messwerte definiert, diese hat aber soweit mir bekannt keinen Namen, es ist eine asymmetrische Verteilung mit zufälligen Vorzeichen. An dieser Stelle ist es sinnvoll diese zu benutzen. Da wir damit aber weitere Fehler erhalten möchte ich eine weitere Idee in das Schaltungsdesign einfließen lassen. Wir nutzen nicht zwei Verstärker Schaltungen mit jeweils individueller Toleranz, einmal um die Spannung am LM317 zu messen und einmal um die Spannung über das Testobjekt zu messen, sondern wir nehmen ein Verstärker und schalten diesen mittels Schalter bzw. Relais jeweils auf die Messgröße.

In der ItSpice Simulation werden wir aber keine Umschaltung simulieren, es geht einfacher, wir zeichnen beide Schaltungsteile, weisen aber den gleichen Bauteilen dieselben Wert zu. Damit wird dann in der Spannungsmessung derselbe Schaltungsteil simuliert wie in der Strommessung. Das ist der Vorteil der Simulation, in der Praxis würde das nicht gehen.

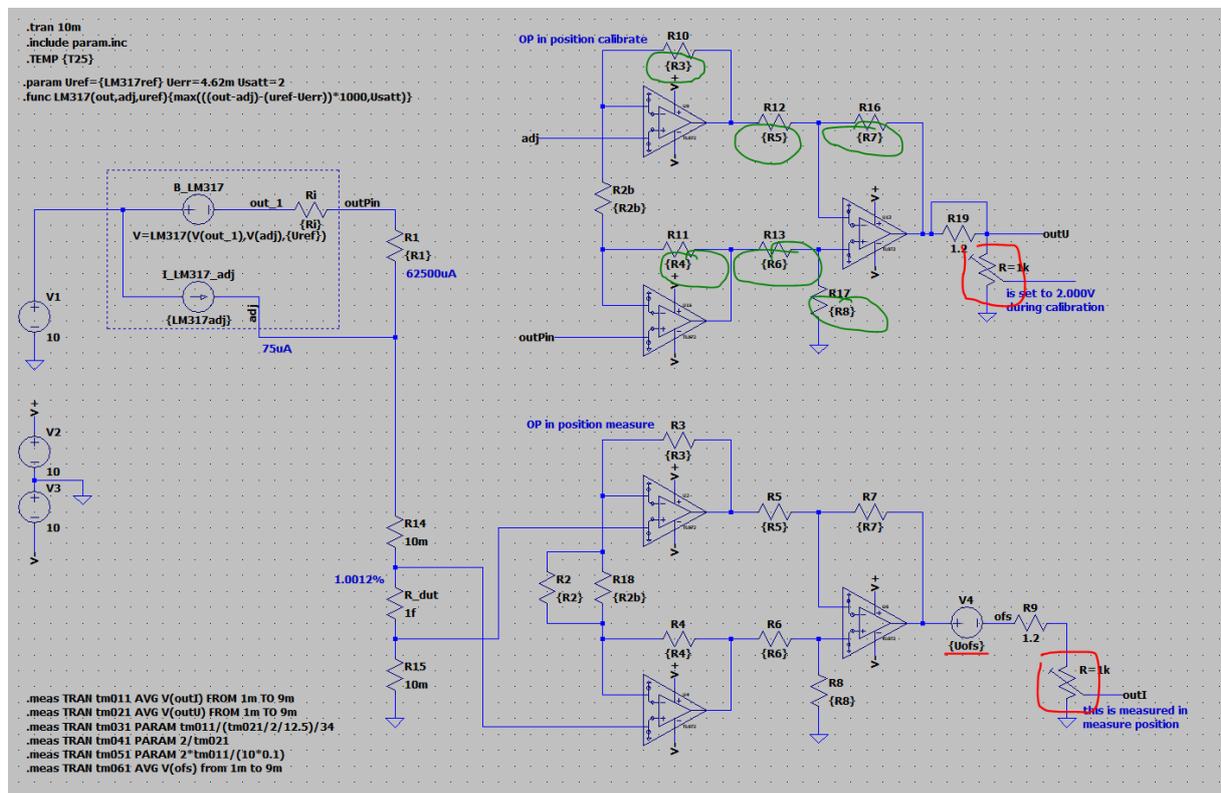
Die nächste Abbildung zeigt die Schaltung wie sie aufgebaut wird, in der Position 1-2 des 4-fach Wechselschalters wird die Anzeige auf 2.000 kalibriert mittels des Potentiometers RV1 und in der Position 3-2 wird gemessen.

Mit dem Schalter SW1-c wird die Verstärkung umgeschaltet, zwischen $\text{gain} = 2$ für die LM317 Spannungsmessung und $\text{gain} = 10$ für die Spannung am Testobjekt.

Um einen groben Abgleich des Adjust Stromes, der mit $75\mu\text{A}$ 0.12% Fehler bewirkt zu kompensieren wird in der Messposition der Widerstand R9 mit 1.2 Ohm wirksam geschaltet um diesen Faktor zu berücksichtigen



Und das ist die Version die wir dazu simulieren, die markierten Widerstände haben den gleichen Wert wie in der Messposition, damit können wir die Spannung outU ermitteln und damit die Einstellung des Trimmers, auch diese sind beide identisch bedatet und haben das gleiche Teilverhältnis. Wenn wir den notwendigen Einstellbereich in der Simulation ermittelt haben können wir den Bereich mit Festwiderständen sinnvoll einschränken um den Bereich des Potentiometers optimal auszunutzen.



Als erstes befüllen wir die Excel Tabelle mit den Werten

ltspice input and output variables [the number of variables must correspond to the simulation setup]												
ltspice input parameter												
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ	asym low	asym high
1	R1	200hm	0,2%	1,33					gauss			
2	R2	3k	0,1%	1,33					gauss			
3	R2b	24k	0,1%	1,33					gauss			
4	R3	12k	1,0%	1,33					gauss			
5	R4	12k	1,0%	1,33					gauss			
6	R5	2k	1,0%	1,33					gauss			
7	R6	2k	1,0%	1,33					gauss			
8	R7	2k	1,0%	1,33					gauss			
9	R8	2k	1,0%	1,33					gauss			
10	Vinp1	±35u	±100u	1,33					log			
11	Vinm1	±35u	±100u	1,33					log			
12	linp1	±0.5n	±6n	1,33					log			
13	linm1	±0.5n	±6n	1,33					log			
14	Vinp2	±35u	±100u	1,33					log			
15	Vinm2	±35u	±100u	1,33					log			
16	linp2	±0.5n	±6n	1,33					log			
17	linm2	±0.5n	±6n	1,33					log			
18	Vinp3	±85u	±150u	1,33					log			
19	Vinm3	±85u	±150u	1,33					log			
20	linp3	±1.6n	±6n	1,33					log			
21	linm3	±1.6n	±6n	1,33					log			
22	LM317ref	1.25V	4,7%	1,33					gauss			
23	Ri	37.5m	50,0%	1					uniform			
24	LM317adj	75u	33,0%	1					uniform			
25	T25	25	100,0%	1					uniform			
26	Tratio	0.812381	0%	1					gauss			
27	Uofs	0	0%	1					gauss			

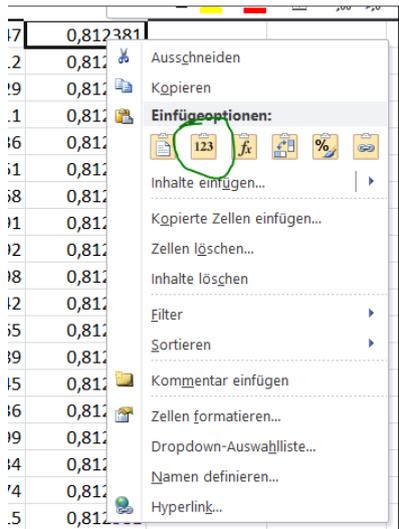
T_{ratio} ist das Verhältnis womit der Spannungsteiler in beiden Positionen eingestellt ist, zu diesem Zeitpunkt können wir erst mal einen beliebigen Wert zwischen >0 und <1 eintragen. Wir schreiben mit „write nominal to param.inc“ diese Werte in die Include Datei und messen einmal Uout, Der Wert beträgt 2.461₈₉₉V, damit am Potentiometer Abgriff 2V gemessen werden muss das Tratio auf 2.461₈₉₉V/2.00000V=0.812381 eingestellt werden, diesen Wert übertragen wir jetzt noch in unsere Tabelle und erstellen zur Kontrolle nochmals das Parameter File und starten noch mal die Simulation zur Kontrolle.

Jetzt sollte der Messbefehl tm051 für den Widerstand einen Wert nahe den 100mΩ ausgeben, es sind 99.9232mΩ

Wir erstellen die Monte Carlo Tabelle und simulieren 100 Durchläufe, das Ergebnis in der Tabelle sieht wie folget aus. In Spalte AC steht unser vorgegebenes T_{ratio} mit dem Nominalwert von 0.812381, in der Spalte AK steht unsere „gemessene Spannung“ in der „Kalibrierposition“, wir berechnen jetzt in der Spalte AQ für jede Position unseren Kalibrierwert indem wir 2/AK21 in der ersten Zelle eintragen und diese Formel bis zum Ende der Tabelle kopieren (dabei wird der Index AK21 automatisch hochgezählt, in der 100-ten Zelle steht dann =2/AK121)

AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ
0,812381	0						-2,72E-04	2,51588	-7,94E-05	0,794949	-5,43E-04	-3,35E-04		=2/AK21
0,812381	0						1,24E-04	2,51299	3,62E-05	0,795865	2,47E-04	1,52E-04		0,79586469
0,812381	0						9,31E-06	2,52045	2,72E-06	0,793508	1,86E-05	1,15E-05		0,7935091
0,812381	0						-7,04E-04	2,54187	-2,04E-04	0,786821	-1,41E-03	-8,68E-04		0,7868223
0,812381	0						-3,46E-04	2,52765	-1,01E-04	0,79125	-6,92E-04	-4,26E-04		0,79124879
0,812381	0						-8,38E-04	2,51251	-2,45E-04	0,796017	-1,68E-03	-1,03E-03		0,79601673
0,812381	0						-2,47E-04	2,53482	-7,16E-05	0,78901	-4,94E-04	-3,04E-04		0,78901066
0,812381	0						-2,39E-04	2,48284	-7,07E-05	0,805529	-4,78E-04	-2,94E-04		0,80552915
0,812381	0						1,26E-03	2,52795	3,67E-04	0,791155	2,52E-03	1,55E-03		0,79115489
0,812381	0						-1,60E-04	2,53961	-4,63E-05	0,787524	-3,20E-04	-1,97E-04		0,78752449
0,812381	0						6,02E-04	2,47694	1,79E-04	0,807449	1,20E-03	7,42E-04		0,8074479
0,812381	0						1,88E-04	2,55457	5,41E-05	0,782912	3,76E-04	2,31E-04		0,78291063

Jetzt „kalibrieren“ wir die Simulation indem wir die Werte in Spalte AQ markieren und kopieren und dann anschließend in Spalte AC als Werte einfügen, rechten Mausclick auf die zelle AC21 und Werte einfügen, siehe Abbildung



Damit erhalten wir für jedes Simulationssetup ein individuell eingestellte Kalibrierung

AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ
0,794950475	-3,35E-04													
0,795864687	1,52E-04													
0,793509096	1,15E-05													
0,7868223	-8,68E-04													
0,791248788	-4,26E-04													
0,796016738	-1,03E-03													
0,78901066	-3,04E-04													
0,805529152	-2,94E-04													
0,791154888	1,55E-03													
0,787522494	-1,97E-04													
0,807447899	7,42E-04													
0,782910627	2,31E-04													
0,799734488	8,69E-04													
0,793921735	-8,56E-04													

Das gleiche machen wir für die Spannungsoffset Kalibrierung, dazu stellen wir in der Simulation die Offsetspannung mit $R_{dut} = 0\Omega$ fest und reduzieren die Ausgangsspannung mittel V4 um diesen Wert. Hier müssen wir nur die Daten aus Spalte A= ind Spalte AD kopieren!

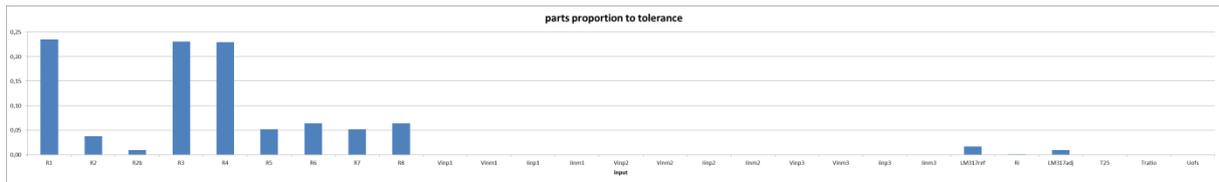
Wir wiederholen jetzt einfach die Simulation ohne die Monte Carlo Tabelle neu zu erstellen, sonst würden wir die „Kalibrierwerte“ überschreiben.

In der realen Schaltung stellen wir den Trimmer RV2 bei kurzgeschlossenen Klemmen so ein, dass wir die Anzeige 0.000V erhalten.

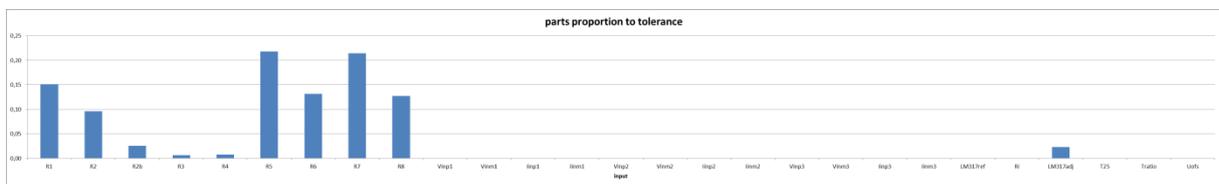
sName	tm011	tm021	tm031	tm041	tm051
sNominalExpected					
dNominalExp	0	0	0	0	0
dLowSpecLir	0	0	0	0	0
dHighSpecLi	0	0	0	0	0
Capability Cp					
Average	0,0500	2,50	0,0147	0,7990	0,1000
SteDev	6,03E-05	0,0323	1,87E-04	0,0103	1,21E-04
lo Limit 4.5S	0,0497	2,36	0,0138	0,7529	0,0994
up Limit 4.5S	0,0503	2,65	0,0155	0,8451	0,1005
AD Test	0,59	0,41	0,71	0,54	0,59

sName	tm011	tm021	tm031	tm041	tm051
sNominalExpected					
dNominalExp	0	0	0	0	0
dLowSpecLir	0	0	0	0	0
dHighSpecLi	0	0	0	0	0
Capability Cp					
Average	0,4999	2,50	0,1468	0,7990	0,9998
SteDev	6,03E-04	0,0322	1,86E-03	0,0102	1,21E-03
lo Limit 4.5S	0,4972	2,36	0,1385	0,7529	0,9943
up Limit 4.5S	0,5028	2,65	0,1552	0,8451	1,0052
AD Test	0,60	0,43	0,73	0,56	0,61

Bei einem Messbereich von 0Ω bis 2Ω beträgt der Messfehler für $100m\Omega$ und für 1.000Ω jeweils $\pm 0.5\%$.

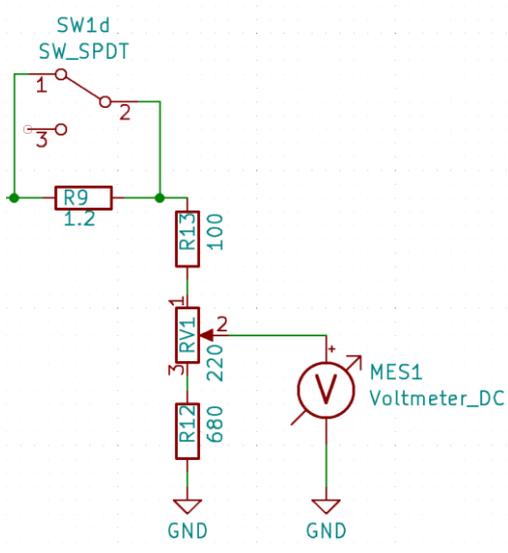


Einflussfaktoren sind dann einmal der Widerstand R1 der den Strom aufgrund der Spannung vom LM314 definiert sowie die beiden Widerstände R2 und R2b (sind schon 0.1%), R3 und R4 (je 1%), wegen der Umschaltung der Verstärkung werden diese Fehler nicht beim Kalibrieren eliminiert. Im Gegensatz zu den Widerständen R5 bis R8, deren Toleranzen zum großen Teil beim Kalibrieren eliminiert werden. Mit Toleranzen für R1, R3 und R4 von 0.1% wird der Messfehler nochmals halbiert auf $\pm 0.25\%$ für 1.000Ω bzw. $\pm 0.3\%$ für $100m\Omega$. Dann zeigt sich auch nochmals der Einfluss von R5-bis R8.



R1 bis R5 sind mit 0.1% und R5 bis R8 mit 1% Toleranz ausgewählt.

Die Einstellung mit RV1 wird den Anforderungen an den Einstellbereich angepasst.



Mit dieser Methode die Kalibrierung von Messgeräte zu simulieren endet diese Tutorial.

Zusammenfassung

Diese Tutorial hat gezeigt, dass statistische Methoden Toleranzen und deren Ursachen in einer Schaltung aufzeigen können ohne dass die Methode Schaltungs know how besitzt. Einfach durch die Anwendung statistischer Methoden. Mit dem Werkzeug kann jeder seine Schaltung untersuchen und ermitteln welche Bauteile eine hohe Toleranzanforderung haben und welche weniger kritisch sind.

Für die Optimierung wurde aber auch in diesen Fällen Schaltungs Know How benötigt, z.B. das man Widerstände niederohmiger wählen muss wenn ein Eingangsoffsetstrom groß ist. Auch hier gibt es statistische Methoden die den Zusammenhang aufzeigen, wesentlich hierbei ist die DOE, Design of

Experience Methode. Damit ist man in der Lage Schaltungsoptimierungen zu ermitteln ohne das man diese Zusammenhänge kennt.

An einer Excel Vorlage die dies unterstützt und Bauteilwerte optimiert ist noch in der Entwicklung und ich werde dies in einer weiteren Anleitung vorstellen.

Vita

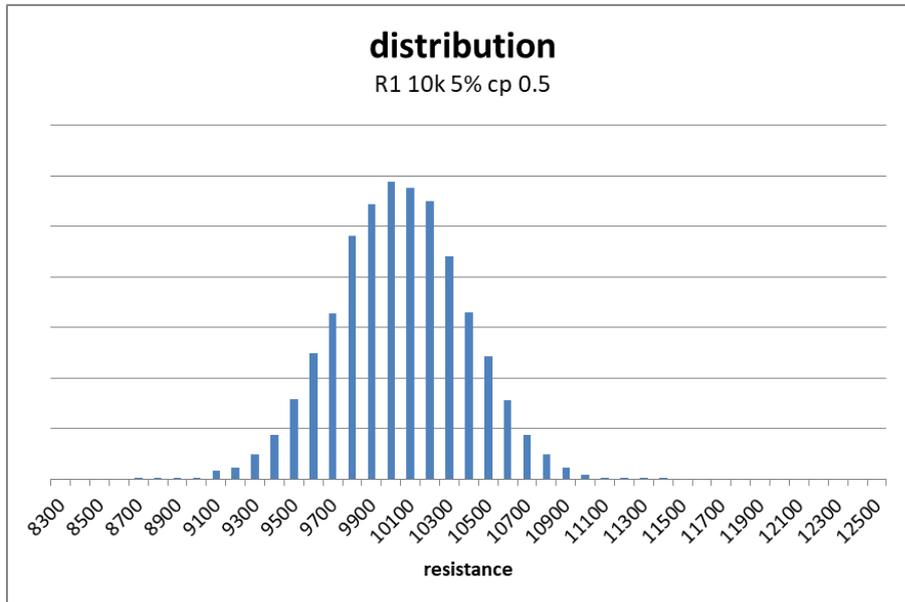
Seit mehr als 30 Jahren als Entwicklungsingenieur in der Automobilindustrie tätig. Neben der Entwicklung elektronischer Hardware wie Leistungselektronik und Messtechnik habe ich mich auch wesentlich mit der Industrialisierung von Sensoren beschäftigt. Hierbei liegt das Hauptaugenmerk weniger auf den Sensor vielmehr auf die Genauigkeit der Messtechnik welche diese Sensoren überpüft.

Diese Aufgaben habe ich seit ca. 20 Jahre in der Funktion des Six Sigma Black Belt national und international wahrgenommen. In dieser Rolle war ich auch verantwortlich für die Entwicklung neuer Methoden zur Steigerung der Qualität.

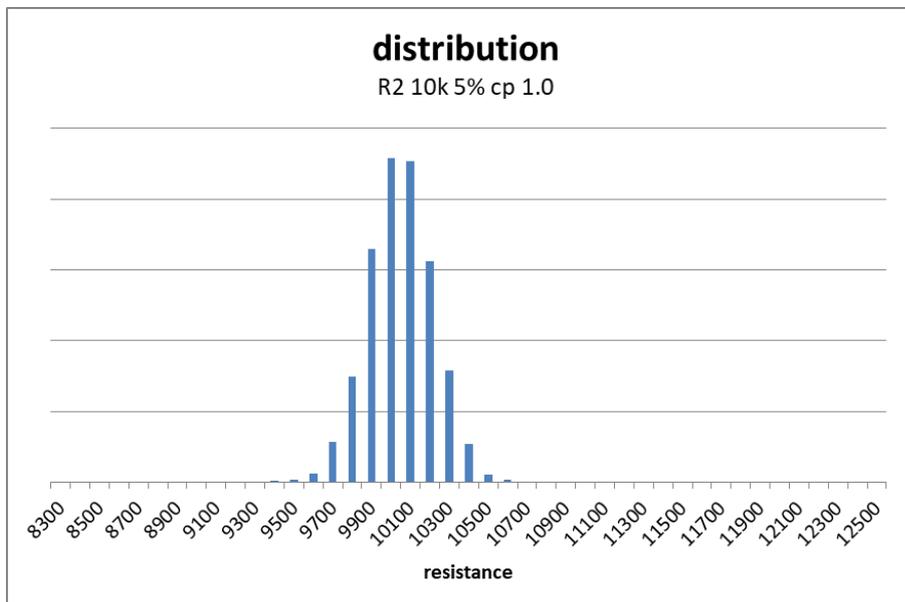
Anhang

Histogramme der Verteilungen die mit dem Excel Template erstellt werden können.

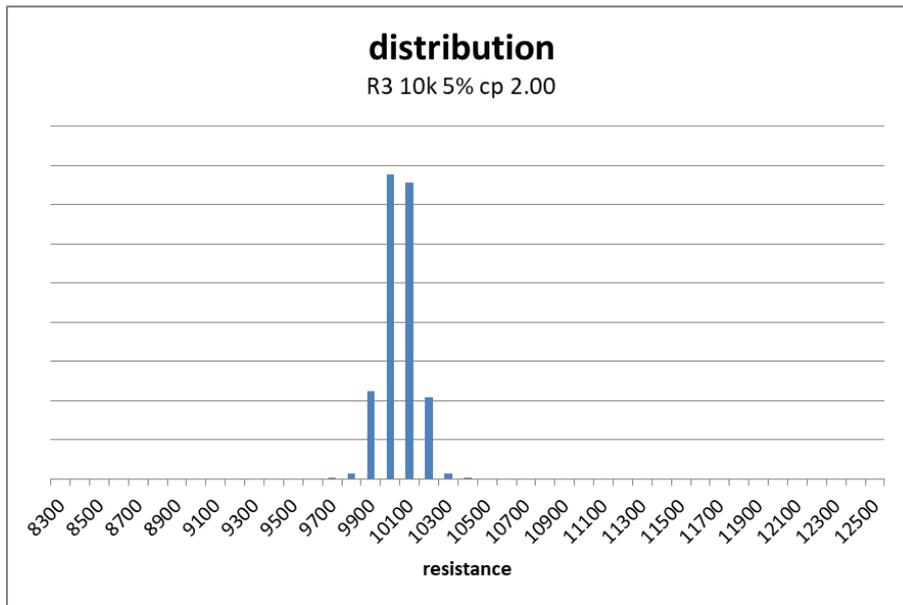
Gauss



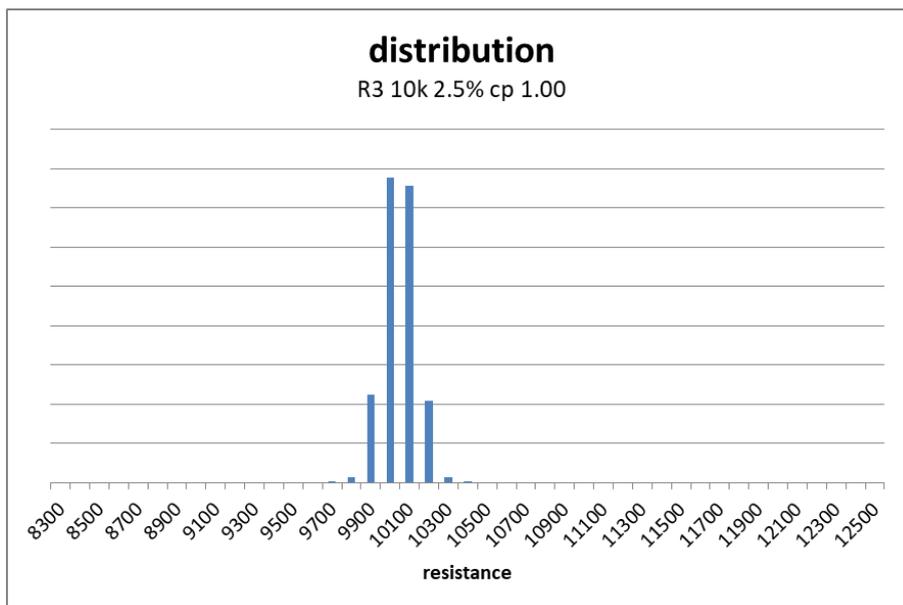
Gauss Verteilung für 10 kΩ mit 5% und $cp = 0.5$, das bedeutet im einseitigen Toleranzfenster von 5% entsprechend 500Ω passen $3 \times cp = 3 \times 0.5 = 1.5$ Standardabweichungen -> Standardabweichung ist $500\Omega / 1.5 = 333\Omega$



Gauss Verteilung für 10 kΩ mit 5% und $cp = 1$, das bedeutet im einseitigen Toleranzfenster von 5% entsprechend 500Ω passen $3 \times cp = 3 \times 1.0 = 3$ Standardabweichungen -> Standardabweichung ist $500\Omega / 3.0 = 166\Omega$, als Folge ist die Gausskurve halb so breit



Gauss Verteilung für 10 kΩ mit 5% und $cp = 2.0$, das bedeutet im einseitigen Toleranzfenster von 5% entsprechend 500Ω passen $3 \times cp = 3 \times 2.0 = 6.0$ Standardabweichungen -> Standardabweichung ist $500\Omega / 6.0 = 83.33\Omega$



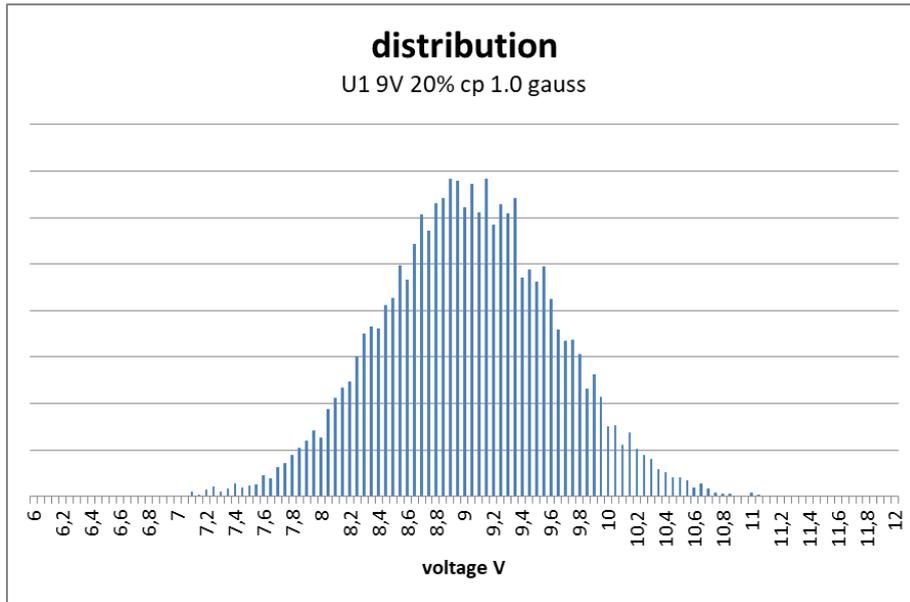
Gauss Verteilung für 10 kΩ mit 2.5% und $cp = 1.0$, das bedeutet im einseitigen Toleranzfenster von 2.5% entsprechend 250Ω passen $3 \times cp = 3 \times 1.0 = 3.0$ Standardabweichungen -> Standardabweichung ist $250\Omega / 3.0 = 83.33\Omega$

Es spielt demnach keine Rolle ob man die Toleranz halbiert oder den Fähigkeitsindex verdoppelt, die Streuung ist in beiden Fällen gleich.

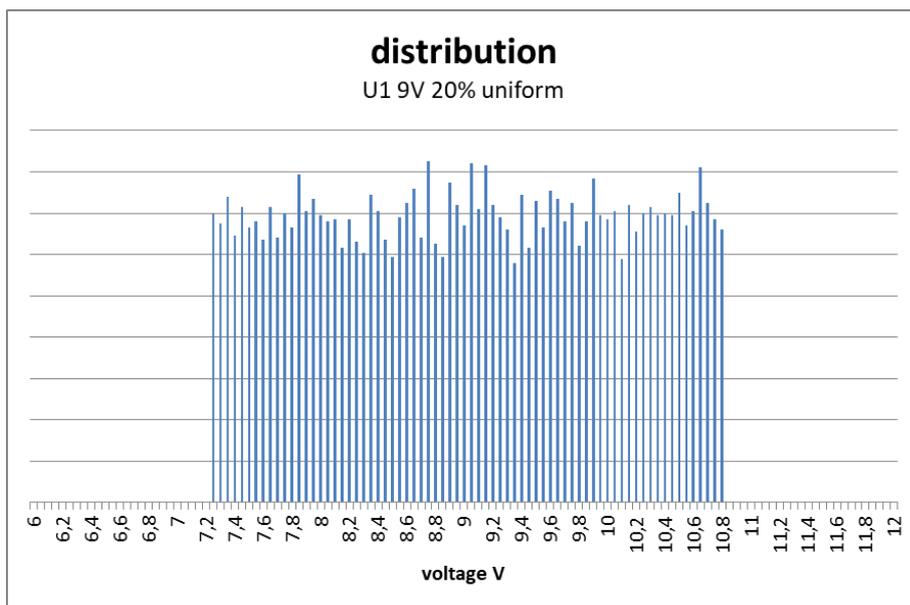
In der Praxis ist es aber so, das zum einen Toleranzen definiert werden und in der Fertigung die Fähigkeits-Indizes definiert sind und überwacht werden. Das ist der Industriestandard.

Für 1% Widerstände ist ein $cp=1.5$ (4.5 Standardabweichungen innerhalb der Toleranz) üblich, das gilt im Neuzustand bei Raumtemperatur.

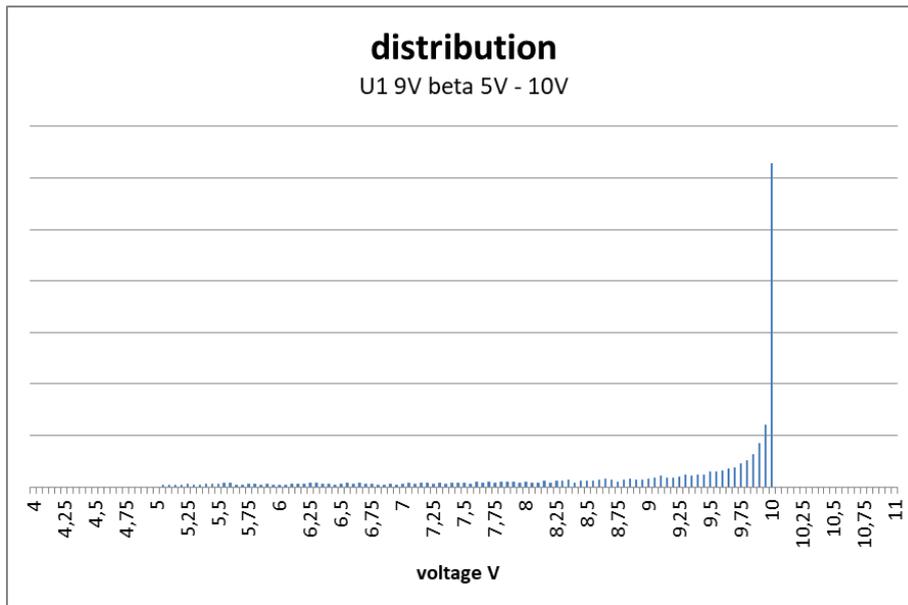
Gauss Uniform und beta verteilung im Vergleich



Gauss Verteilung (Median) 9V 20% mit $cp=1.0$, das bedeutet im einseitigen Toleranzfenster von 20% entsprechend 1.8V passen $3 \times cp = 3 \times 1.0 = 3$ Standardabweichungen, \rightarrow Standardabweichung ist $1.8V/3=0.6V$

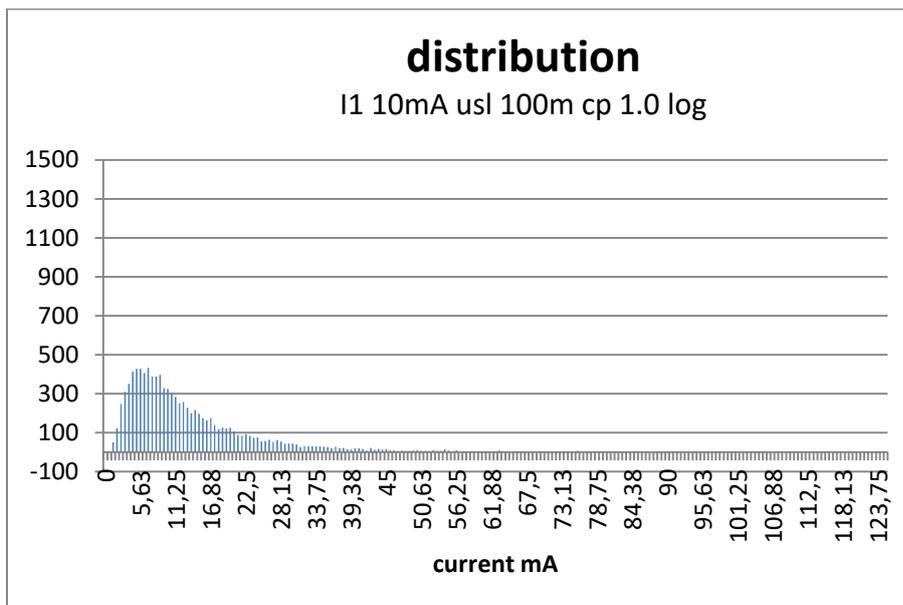
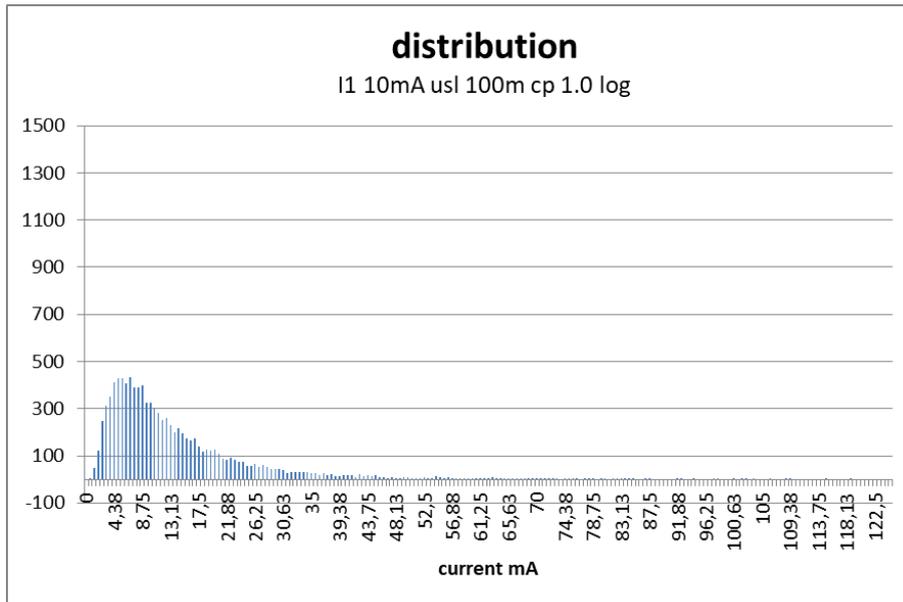


Gleichmäßige Verteilung (Median) 9V 20%, das bedeutet die Werte sind gleichmäßig verteilt von $9V-20\% = 7.2V$ bis $9V+20\%=10.8V$



Beta Verteilung mit (Median) 9V und der unteren Grenze von 5V und der oberen Grenze von 10V, die Verteilung ist links schief (rechts steil). Eine solche Verteilung kann gewählt werden wenn im Datenblatt Angaben der typische Wert deutlich oberhalb dem min Wert liegt und nahe am max Wert.

Log normal Verteilung

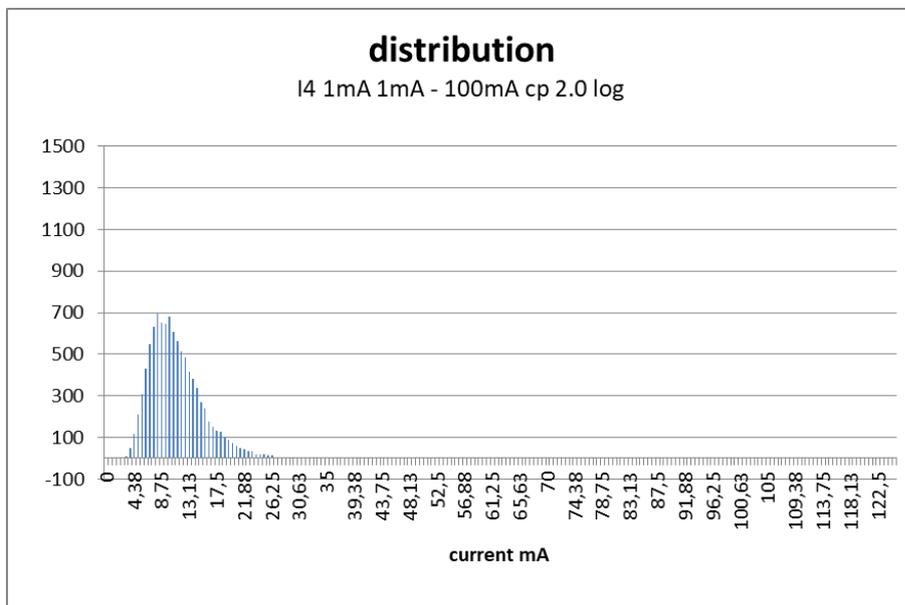
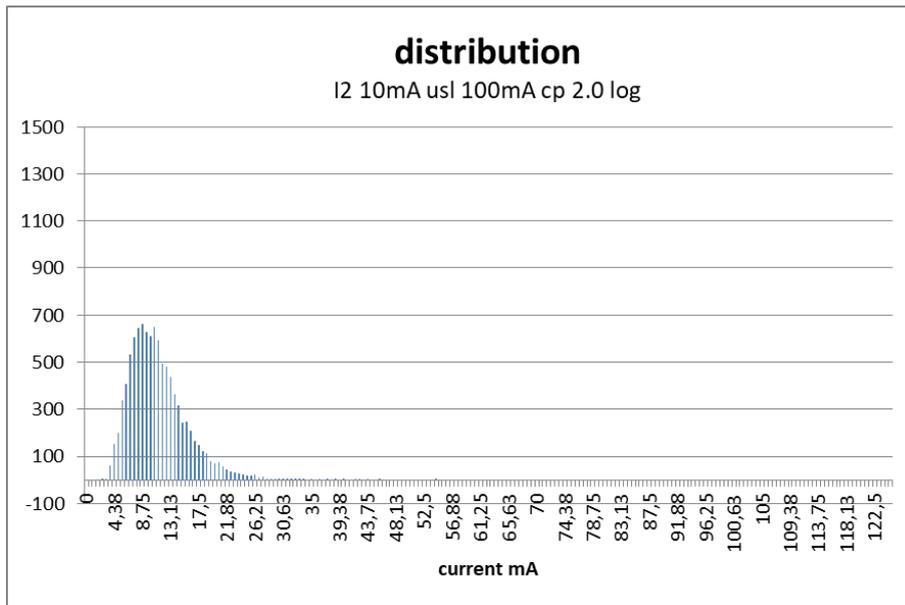


Log Nomal Verteilung der oberen Grenze von 100mA und oben mit dem Median Wert 10mA und unten mit dem Min Wert 1mA und cp=1.0. Beiden Kurven sind identisch. Der Zusammenhang zwischen LSL USL und Median ist:

$$median = \sqrt{lsl \cdot usl}$$

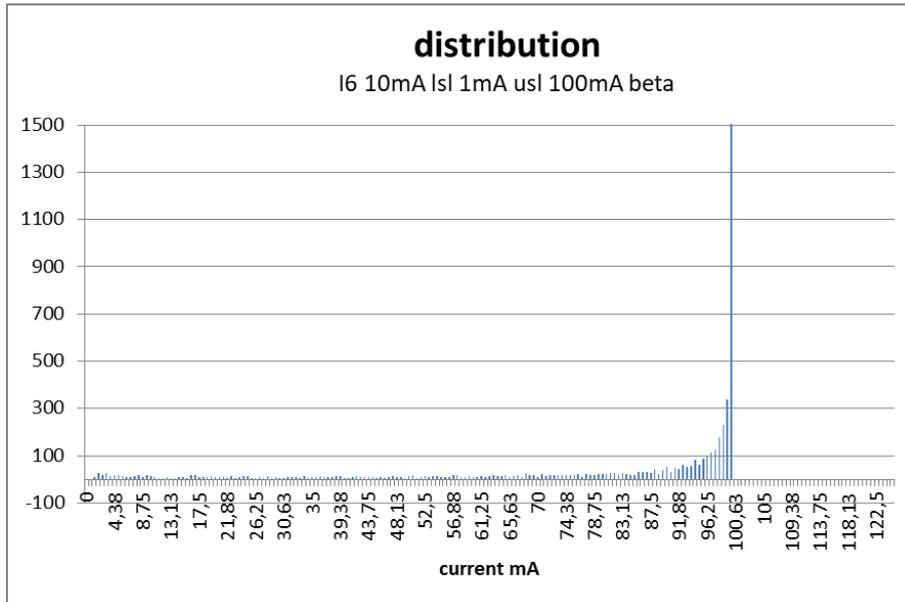
lsl = lower specification limit *usl* upper specification limit

Die Verteilung ist rechts schief (links steil). Eine solche Verteilung kann gewählt werden wenn im Datenblatt Angaben der typische Wert deutlich unterhalb dem max Wert liegt und nahe am min Wert

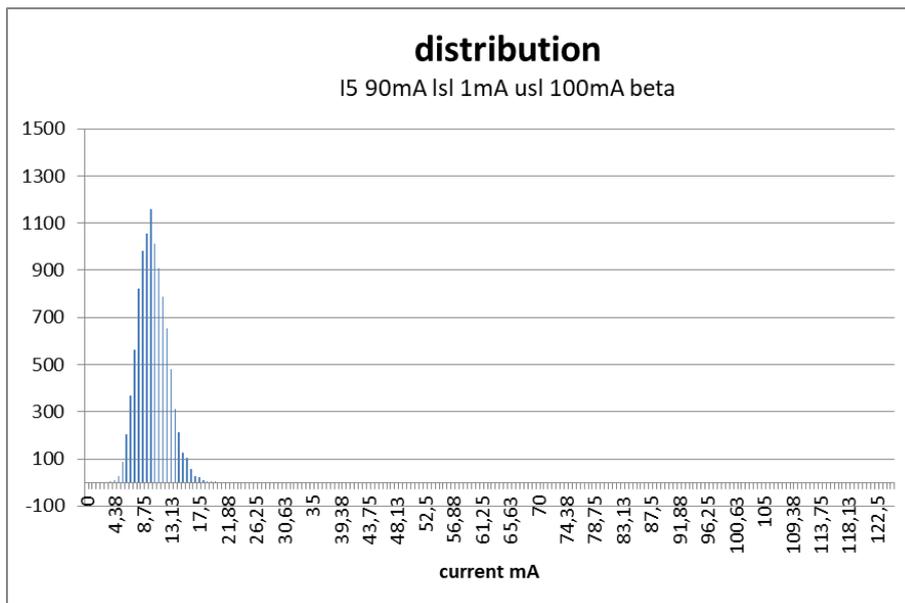


Log Normal Verteilung der oberen Grenze von 100mA und oben mit dem Median Wert 10mA und unten mit dem Min Wert 1mA und $cp=2.0$. Die Anzahl an Werten im Bereich rund um 10mA ist mit dem besseren cp deutlich höher, darum sind Werte weit oberhalb 10mA deutlich weniger vorhanden.

Beta Verteilung

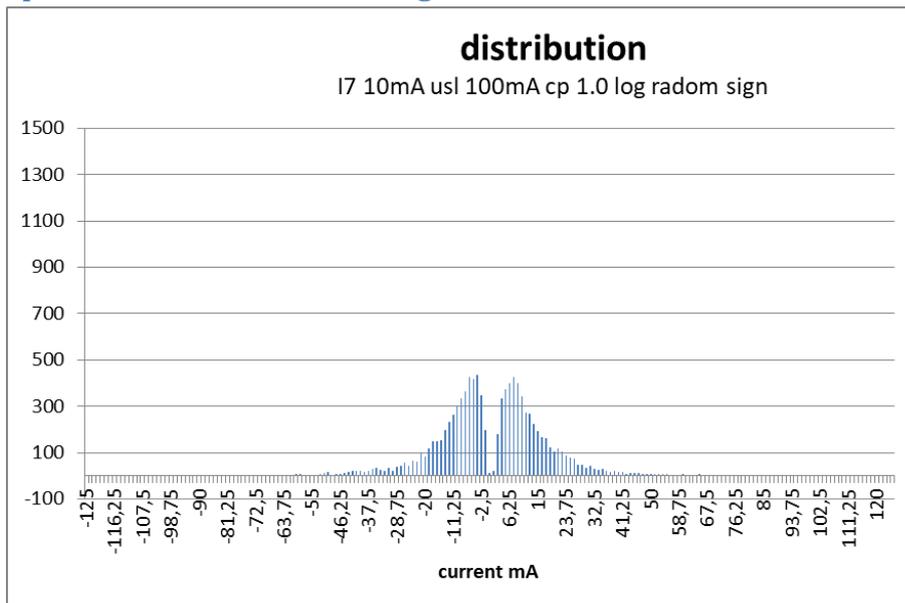


Beta Verteilung mit Median 90mA min Wert 1mA und max Wert 100mA. Die beta verteilung kann man quasi als gegenstück zur Log Normalverteilung sehen (Mathematiker mögen mir die flapsige Beschreibung verzeihen) weil diese von der Schiefe genau andersherum ist.



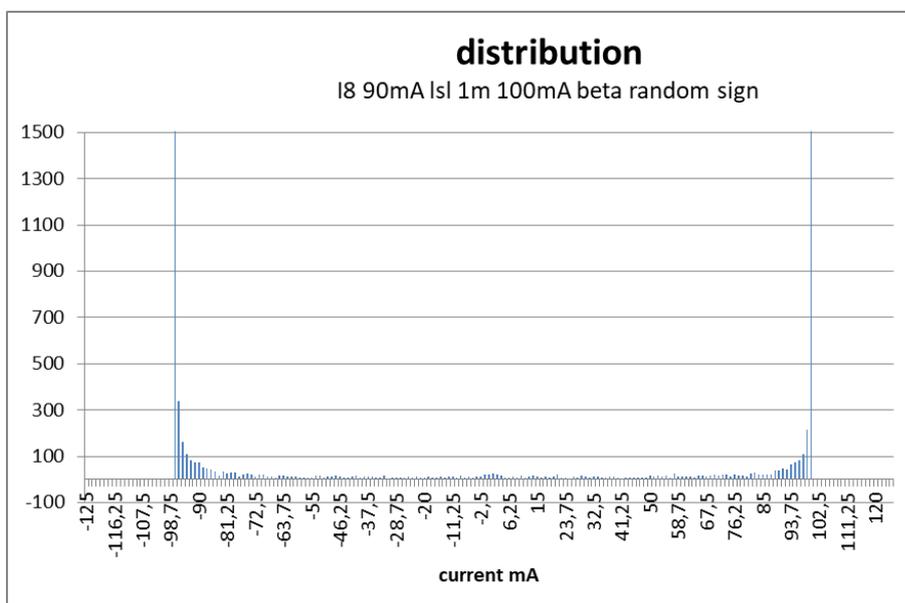
Das ist das Ergebnis einer Beta Verteilung Median 90mA min Wert 1mA und max Wert 100mA. Die Beta Verteilung ist nicht geeignet für rechts schiefe (links Steil) Verteilungen wenn wir eine signifikante Schiefe sehen wollen.

Spezielle namenlose Verteilungen



Das ist auf Basis einer Log Normalverteilung mit median 10mA und max. Wert 100mA und $cp=1.0$, die Log Normalverteilung hat nur positive Werte, es gibt aber die Anforderung Toleranz in dieser Form darzustellen typ $\pm 10\mu A$, max $\pm 33\mu A$ oder min= $\pm 3\mu A$, max $\pm 33\mu A$

Sollte min= $\pm 3\mu A$ typ $\pm 10\mu A$, max $\pm 33\mu A$ vorgegeben werden dann ist erst zu prüfen ob der Wert für typ den Zusammenhang erfüllt $abs(\text{median}) = \sqrt{abs(\text{lsl}) \cdot abs(\text{usl})}$ also die „±“ ignoriert Wenn nicht muss man gegebenenfalls einen Kompromiss eingehen und entweder den typ Wert ignorieren oder die untere Grenze lsl, mehr Sicherheit hat man meistens wenn man lsl ignoriert.



Das gleich auch mit der Beta Verteilung, auch hier mit zufälligen Vorzeichen wenn es entsprechenden Angaben analog zur log normal Verteilung umzusetzen gibt.

Übersicht Verteilungen für Bauteile

Das ist jetzt nicht die ultimative richtige Tabelle welche Verteilungen einzustellen sind. Wenn man das wirklich mal ganz genau wissen muss bedeutet das viele Stunden Diskussion mit dem Hersteller.

Bezeichner	Beschreibung	Toleranz neu	Tol. Incl. Alterung & Temp.	Fähigkeit	Verteilung	Anmerkung
R	Kohleschicht 5%	5%	8%	1.5	Gauss	
R	Metallfilme 1%	1%	2.5%	1.5	Gauss	
R	Metallfilm 0.1%	0.1%	0.5%-1%	1.5	Gauss	
C	Kondensator 10%	10%	15%	1.5	Gauss	
C	Kondenstaor 5%	5%	7%	1.5	Gauss	
C	Elektroly Kondensator 10%	20%	30% -	1.5	Gauss neu	Beta Alterung
L	Induktivitäten			1.5	Gauss	
IC	Spannungsreferenz	Siehe Datenblatt	Gauss	1.5-2.0		
IC	Offsetspannung		Log	1.5-2.0		
IC	Eingangsoffsetstrom		Normal	1.5-2.0		
IC	Gain Bandwith		Gauss	1.5-2.0		
IC	OPamp I bias		Log	1.5-2.0		
IC	OPamp I offset		Log	1.5-2.0		

Ich hatte aus Gründen der Vereinfachung für I_{inp} und I_{inm} immer I offset Werte aus dem Datenblatt übernommen. Die korrekte Vorgehensweise ist aber ein Verteilung für I_{Bias} und eine für I_{Offset} zu erstellen.

Beispiel OP 07

		min	typ	max
I_B	Input Bias Current		$\pm 1.8nA$	$\pm 12nA$
I_{os}	Input Offset Current		$\pm 0.8nA$	$\pm 6nA$

Daraus folgt:

Itspice input and output variables [the number of variables must correspond to the simulation setup]										
Itspice input parameter										
pos	parameter	value	tolerance	cp	cpk	meanofs	low Limit	high Limit	distribution	tolerance typ asym low asym high
1	lbias	$\pm 1.8n$	$\pm 12n$		1,5				log	
2	loffset	$\pm 0.8n$	$\pm 6n$		1,5				log	

Und in Itspice: $I_{inp} = I_B + \frac{I_{os}}{2}$ und $I_{inm} = I_B - \frac{I_{os}}{2}$

