Antennensimulationen mit 4NEC2 einschließlich Anwendungsbeispiel für 2300 MHz

von Gunthard Kraus, DG8GB

1. Simulation von Drahtantennen mit NEC

NEC (= Numerical Electric Code) ist eine vom Lawrence Livermore Labor im Jahre 1981 für die Navy entwickelte Simulationsmethode für Drahtantennen. Dabei wird die Antenne in sehr kurze Stücke = „segments“ zerlegt, auf denen sich der Strom und die Spannung (fast) nur linear ändern. Dann wird das Verhalten jedes Segmentes berechnet und am Ende alles aufsummiert (= integriert). Damit lassen sich erstaunlich genaue Simulationen durchführen. Der Standard -- für den es sehr viele, auch kostenlose, Anwendungssoftware gibt -- ist hierbei **NEC2**. Die Entwicklung ist natürlich weitergegangen und die Schwächen von NEC2 (z. B. falsche Berechnung von Strukturen mit sich sehr nahe kreuzenden oder in geringem Abstand parallel verlaufenden oder im Boden verbuddelten Drähten) wurden erst mit **NEC4** ausgebügelt, aber:

NEC4 war lange für den Export gesperrt und galt als geheim. Heute ist es auch außerhalb der USA erhältlich, aber recht teuer (500$).

NEC2 könnte man sich aus dem Internet herunterladen, aber damit lässt sich noch nichts anfangen. Es handelt sich dabei um die reine „Rechenmaschine“, die ursprünglich in Fortran geschrieben wurde und auch als compiliertes Maschinenprogramm zur Verfügung steht. Folglich haben viele Leute dazu umfangreiche Bedieneroberflächen entworfen und diese als Windows-Programme vermarktet -- teils kostenlos, teils gegen erträgliche Preise. Hier sticht nun **„4NEC2“** mit großem Abstand heraus, denn es wurde mit soviel Zusatzprogrammen und Möglichkeiten ausgestattet, dass man nur staunen kann. Und es ist **kostenlos**. Deshalb Lob und Dank an den Autor Arie Voors!

2. Richtiger Umgang mit 4NEC2

Es ist wichtig, die Spielregeln für eine erfolgreiche Simulation einzuhalten. NEC arbeitet nämlich nach der „Momentenmethode“, bei der die Antennenstruktur aus Drähten („Wires“) besteht und bei der jeder Draht selbst wieder in Segmente unterteilt wird. Dabei darf man folgende Vorschriften nicht verletzen:

Die Länge jedes Segmentes muss kleiner als 5% bis maximal 10% der Wellenlänge sein.

Bei der Drahtstärke muss man genau hinsehen, sonst handeln wir uns Probleme ein. Maßgebend ist dabei das **Verhältnis von Segmentlänge zu Drahtradius**:

Ist dieses **Verhältnis größer als 8**, dann reicht das normale Rechenprogramm mit dem Namen „**Thin Wire Kernel“** aus. Es simuliert den Stromfluss im Draht als einen hauchdünnen Stromfaden.

Bei einem **Verhältnis zwischen 2 und 8** muss der „**Extended Thin Wire Kernel“** über die Karte **„EK**“ im NEC-File aufgerufen werden. Damit wird der Strom rechnerisch gleichmäßig auf den Umfang des Drahtes verteilt und das erhöht die Genauigkeit beim Ergebnis beträchtlich.

Für ein **Verhältnis unter 2** sinkt die Rechengenauigkeit sehr rasch, bleibt aber beim EK noch erträglich bis herunter zu einem Wert von 0,5.

Also wird 4NEC2 nun aus dem Internet geholt, entpackt und installiert. **Unbedingt gehört dazu der Download von NEC-2/MP.** Der Zusatz „MP“ bedeutet nämlich **„Multiprocessing“** und mit den darin enthaltenen überarbeiteten „exe“-Dateien erhält man eine deutlich höhere Rechengeschwindigkeit und so eine verminderte Wartezeit auf das Ergebnis. Der Download erfolgt von
http://www.ok1rr.com/files/NEC2MP.ZIP

In der Zwischenzeit scheinen diese „MP“-Dateien aber bereits im 4NEC2-Paket enthalten zu sein und werden deshalb bei der Installation automatisch einbezogen.

2.1. Erstellung des NEC-Files für einen 2375 MHz-Dipol

Da gibt es mehrere Möglichkeiten und sie verstecken sich hinter „Settings“ im 4NEC2-Menü.Da findet man nacheinander:

Notepad Editor / NEC Editor / Geometry Edit / NEC Editor (new)

Aber dabei geht nichts über den **Notepad-Editor unter Windows**, denn mit ihm lassen sich bei zunehmender Erfahrung Änderungen und Erweiterungen am schnellsten einbringen und austesten. Allerdings darf man nie vergessen, beim fertigen File für den Dateityp „**Alle Dateien“** einzustellen und dann beim Dateinamen immer die Endung „***.NEC**“ anzuhängen.....
So müssen Standard-NEC-Files im Prinzip aufgebaut werden: es handelt sich um eine Folge von Anweisungen, die als „**Cards**“ = „Karten“ bezeichnet werden. Denn die zugehörigen Details jeder Karte wurden früher auf IBM-Lochkarten gespeichert und deren Speichersystem (rows and columns = Reihen und Spalten) gilt noch heute.

CM (= Comment = Kommentar)

' = Mit dem Hochkomma leitet man bei Bedarf eine Kommentarzeile ein

CE (= Comment End)

SY (Symbols = nun folgt eine Sammlung an Deklaration von Variablen für die verschiedenen verwendeten Faktoren oder physikalischen Größen oder Berechnungsgleichungen)

GW (Geometry of wire = Definition von Drähten mit allen nötigen Details)

GM (Geometry Move = Drehen, Verschieben oder Kopieren der erzeugten Struktur bzw. Teilen davon)

GE (Geometry End = Ende der Struktureingabe). Zusätzliche Eingabe des Groundtyps möglich.

LD (= Load = damit können z. B. zusätzliche Bauteile wie R, L, C bei bestimmten Segmenten eingebaut werden)

FR (= Frequency = Angabe der Frequenz oder des gesweepten Bereichs)

EK (= Extended Kernel = Programmzusatz zur korrekten Simulation von dünnen Drähten)

EX (= Excitation = Anregung der Struktur durch Zufuhr von Energie)

EN (= End = File Ende)

Die Details wollen wir nun am NEC-File des Lambda-Halbe-Dipol genauer ansehen.

Bitte beachten:
Alle Längen- oder Durchmesserangaben erfolgen IMMER in der Maßeinheit „Meter“!
Nun folge das echte NEC-File. Rot geschrieben sind die Erklärungen – sie sind nur hier im Vortrag erlaubt!
Zwischen den einzelnen Karten sollte man vorsorglich leere Kommentarzeilen (durch das Hochkomma) vorsehen.

CM Lambda - Halbe Dipol **= Kommentar**
CE **= Kommentarende**
'

SY freq=2375 **= Es soll bei der Frequenz f = 2375 MHz gearbeitet werden**
SY dipolarm=300/freq/4 **= Jeder Dipolarm hat die Länge „Lambda-Viertel“ bei 2375 MHz. So kann man leicht auf eine andere Frequenz gehen!**

SY dr=0.002/2 **= Der Radius des Antennendrahtes beträgt 1 mm**

GW 1 9 0 -dipolarm 0 0 dipolarm 0 dr = Nun müssen wir den Dipol als Wire definieren. Das geht so:

1 = Der Draht stellt „tag 1“ (= Objekt 1) dar
9 = Er wird in 9 Segmente aufgeteilt
0 -dipolarm 0 = xyz-Kordinaten des Drahtanfanges
0 dipolarm 0 = xyz-Kordinaten des Drahtendes
dr = Zum Schluß kommt noch der Drahtradius rein
'

GE = Geometry End = Ende der Stuktureingabe

FR 0 1 0 0 freq 0 = Programmierung des Sweep-Frequenzbereichs
Erklärung:
0 = Linearer Sweep
1 = Anzahl der Frequenzschritte (= hier nur eine Frequenz)
0 = beide folgenden Plätze auf der Karte müssen leer bleiben
0
freq = Startfrequenz
0 = Step size, also die Breite des Frequenzschrittes
'

EK = Extended Kernel = Zusatzprogramm für dünne Dräht aktiviert
'

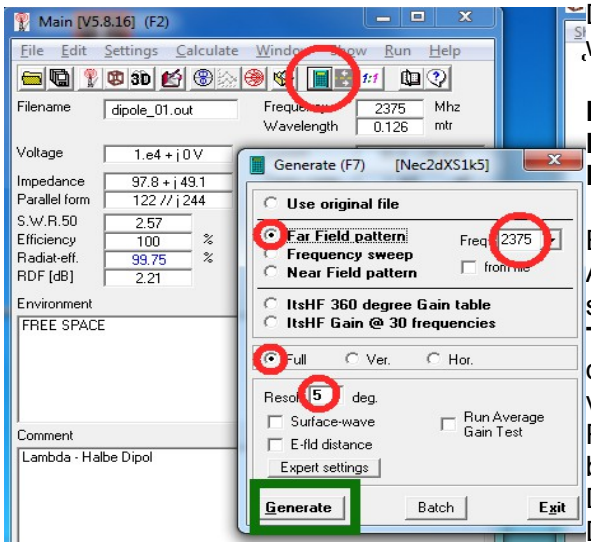
EX 0 1 5 0 1 0 = Excitation = Anregung bzw. Speisung mit Energie.

0 = Anregung durch Spannungsquelle (Stromquelle wäre „6“)
1 = Anregung in Tag 1.....
5 = ...und dort in Segment 5
0 = Leerer Platz auf der Karte
1 = Anregung durch eine reelle Spannung mit 1 Volt
0 = Kein imaginärer Spannungsanteil
'

EN = Ende des NEC-Files

Hat man dieses NEC-File fertiggestellt und gespeichert, braucht man nur noch die Software 4NEC2 zu starten und über „**File**“ und über „**Open 4NEC2 in/out file**“ für den Aufruf des erstellten Files zu sorgen.

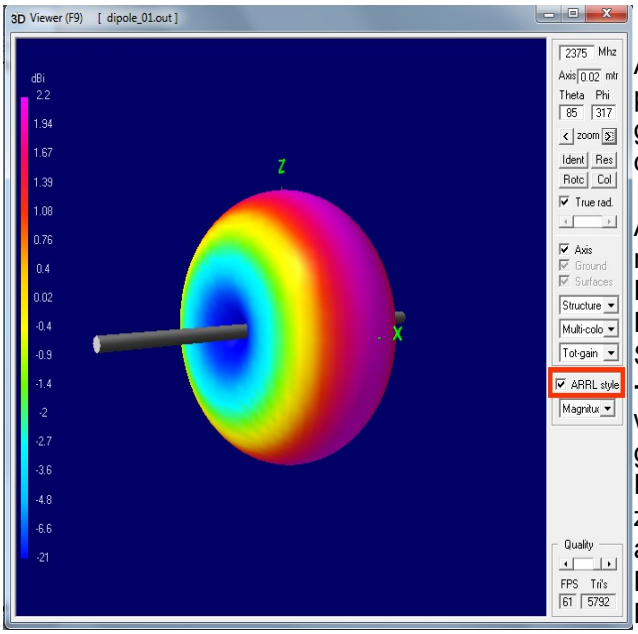
2.2. Simulation und Optimierung der Dipoleigenschaften



Dazu klickt man auf das Symbol des Taschenrechners, wählt „**Farfield**“ und prüft die Einstellungen:

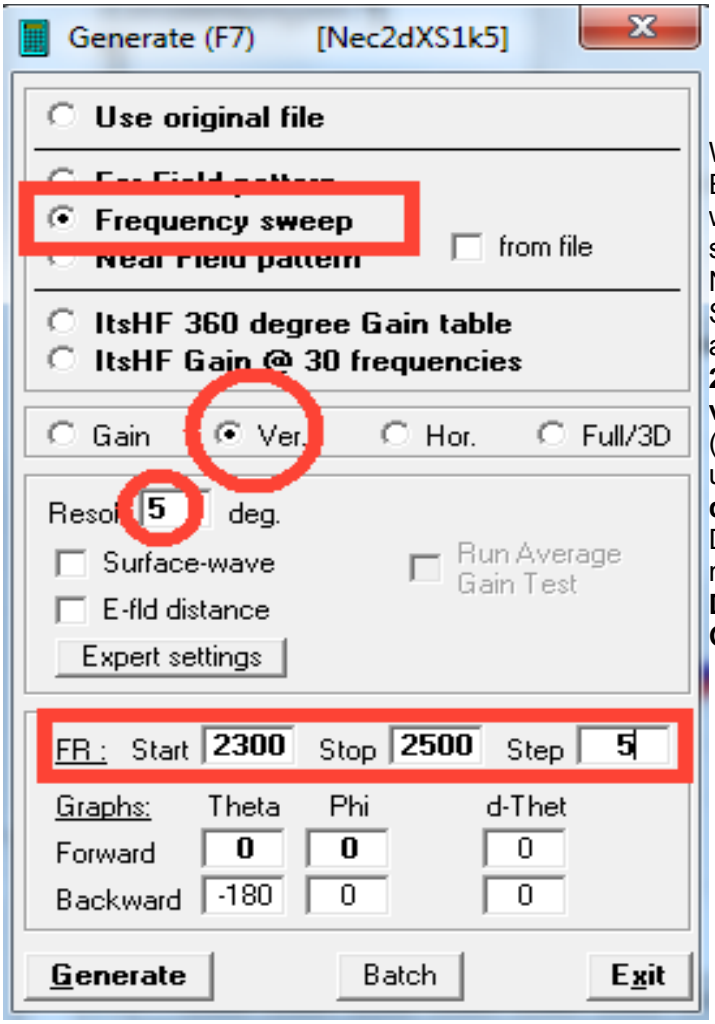
Frequenz = 2375 MHz
Full
Resolution = 5 degrees

Ein Klick auf „**Generate**“ startet die Berechnung und Ausgabe der gewünschte Richtdiagramme. Man sollte sich aber gleich selbst eine Freude machen und die **Taste F9** drücken. Damit erscheint eine 3D-Darstellung der Antennenstruktur (...vorausgesetzt ist die Installation von „DirectX“ auf dem Rechner sowie eine Rechnerhardware, die damit umgehen kann...eigene bittere Erfahrung mit einem älteren Notebook...).
Das Ergebnis ist eine hübsche 3D-Darstellung des Dipols. Er kann durch „Ziehen mit der Maus“ gedreht und herumgeschwenkt werden, seine Größe lässt sich über die Zoom-Tasten verändern. Im Untermenü mit der Angabe „**hide pattern**“ finden wir aber „**Multi-color**“ und wer das anklickt, bekommt folgendes Bild auf den Schirm.

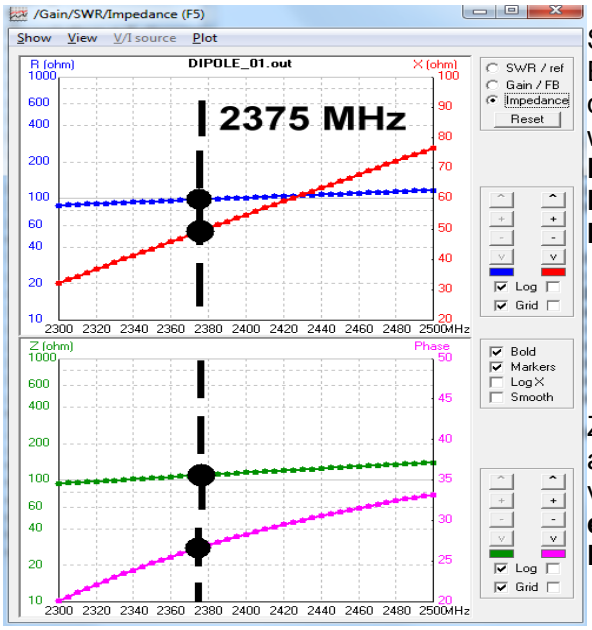


Auch da stellt man sich mit den Zoom-Tasten eine passende Größe des Strahlungsdiagramms ein und genießt die Codierung der Gewinnwerte in **„dBi“** durch die links angeordnete Farbskala.

Allerdings ist da eine ganz wichtige Bemerkung nötig:
Rechts unten ist ein Häkchen bei „**ARRL**“ gesetzt. Da wird mit einer komplett anderen logarithmischen Skala gearbeitet und dadurch der **Bereich von +2,15 dBi bis -4,7 dBi stark gedehnt dargestellt**, während es ab -4,7 dBi mit einem winzigen Ruck gleich zu -21 dBi hinuntergeht.
Hat seine Vorteile, wenn man das absolute Minimum zwar noch erkennen will, aber den größeren Wert auf eine gute Auflösung legt, wenn es in Richtung Maximum geht.
Man braucht ja nur kurz das Häkchen bei „ARRL“ entfernen um den Unterschied zu sehen: dann haben wir eine korrekte (= dB-lineare) farbige dBi-Teilung vor uns und die Dehnung in der Nähe des Maximums ist verschwunden. Muss man ausprobieren und dann selbst entscheiden, was praktischer ist....

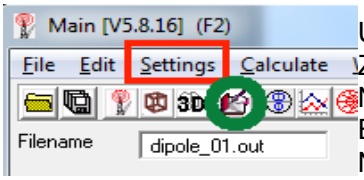


Wer sich für den Verlauf der Eingangsimpedanz interessiert, der wählt wieder die „Taschenrechner-Taste“ und schaltet dann auf „**Frequency sweep**“ um. Natürlich darf man nun nicht vergessen, den Sweep-Bereich und die Schrittweite anzugeben. Gewählt wurde ein Bereich von **2300 bis 2500 MHz** mit einer **Schrittweite von 5 MHz**, die übrigen Grundeinstellungen (z. B. Winkelauflösung = 5 Grad) bleiben unverändert, **aber das Programm schaltet dabei selbständig auf „Vertikal“ um.**
Die Berechnung dauert nun wenig länger und man kann beim Ergebnis unter der **Darstellung von SWR, Reflektionsfaktor, Gain oder Impedance** auswählen.



Sehen wir uns jedoch den ermittelten Eingangsimpedanzverlauf genauer an. Da erkennen wir, dass die Antenne eine gehörige „Open End Extension“ wie jede am Ende offene Microstrip-Leitung aufweist. **Eine Wiederholung des Sweeps für 2100 bis 2200 MHz zeigt uns, dass dadurch die Resonanzfrequenz bis auf 2160 MHz abgesunken ist !**

Zur Korrektur sieht man einen **Verkürzungsfaktor „x“** als neues Symbol für die Dipol-Armlängen im NEC-File vor und spielt etwas mit seinem Wert. **Mit x = 0,907 erhält man dann genau die gewünschte Resonanzfrequenz von 2375 MHz.**



Und das geht so:
Zuerst rufen wir die Settings (= roter Rahmen) auf und wählen den Notepad-Editor (= erste Option).
Ein Klick auf den Editor-Button (= grüner Kreis) öffnet dann das benützte NEC-Input File. Darin wird der Korrekturfaktor als zusätzliches Symbol „x“ eingefügt. Die y-Koordinaten der beiden verwendeten Wires werden dann zusätzlich mit diesem Faktor multipliziert.

```
CM Lambda - Halbe Dipol
CE
'
SY freq=2375
SY dipolarm=300/freq/4
SY dr=0.002/2
SY x=0.907
'
GW 1 9    0  -dipolarm*x  0   0  dipolarm*x  0  dr
'
GE
'
FR 0 1 0 0 freq
'
EK
'
EX 0 1 5 0 1 0
'
EN
```

GANZ WICHTIG:
Nach dem Öffnen des Datei-Menüs im linken oberen Eck des Editors wählt man „Speichern“. Dadurch schließt sich das Menü wieder. Aber wir öffnen es nochmals und klicken darin auf „Beenden“.
Wer das nicht so macht, handelt sich unwahrscheinlichen Ärger ein, denn ohne „Beenden“ werden die Ergebnisse der vorherigen Aktion nicht gelöscht, sondern die Simulationsergebnisse unserer Änderung einfach dazu geschrieben. Das gibt unglaublich wirre Ergebnisse (...= eigene Erlebnisse mit etlichen Stunden Fehlersuche...)**!**

Hinweis:
Es gibt unter „**Settings**“ noch drei weitere Editor-Optionen. Die sollte man sich gelegentlich mal ansehen. In manchen Fällen ist es nämlich von Vorteil, unter „**Settings“** die **zweite Möglichkeit,** nämlich „**NEC Editor**“ zu wählen. Sie präsentiert das NEC-File in derselben Form wie Notepad, liefert aber **sämtliche Detais einer angeklickten Zeile (Card / Tag / Segment / Koordinaten usw. usw.) in Form einer zusätzlichen Übersicht**. Nun muss man nicht mehr im NEC-Handbuch nachsehen, was ein bestimmter Wert gerade bedeutet..... und das betreffende Teil wird in der Geometrie-Darstellung extra grün markiert.

Auch die **vierte Editorversion „NEC Editor (new)“** ist oft sehr hilfreich. Hier wird **alles in Tabellenform** dargestellt, die eingetragenen Werte können leicht geändert und neue Tags hinzugefügt werden.

Die **dritte Möglichkeit** („**Geometry Editor“, Tasten <Ctrl> +< F3>**) öffnet eine räumliche Darstellung des untersuchten Objektes und beim Anklicken eines Wires der Antenne eine unglaubliche Menge an Details. Ist fast zuviel und es ist viel besser, nur **<F3> zu drücken**. Dadurch öffnet sich die Darstellung „**Geometry**“. Und bei gleicher Darstellung kann man dann unter „Show“ die **Wire – oder Tag- Nummern oder die einzelnen Segmente** einblenden lassen.
Das reicht in den meisten Fällen und ist wesentlich übersichtlicher

3. Simulation einer Übertragungsstrecke

3.1. Eine zweite Antenne muss her

Mit NEC lassen sich auch Simulationen im Fernfeld durchführen und das kann man ausnützen, um z. B. später die genauen Eigenschaften anderer simulierter Antennen zu untersuchen. **Man ordnet dabei einfach einen „Messdipol“ in ausreichendem Abstand an, dreht und wendet diesen nach Geschmack und ermittelt jedes Mal den fließenden Strom in seinem 50 Ω – Abschlusswiderstand.**
Addieren wir also in unserem NEC-File „dipole_01.nec“ einen identischen Dipol (durch Kopieren der Originalantenne) in **x = 2 Meter** Abstand. Das entspricht 16 Wellenlängen und somit einen sicheren Betrieb im Fernfeld . Außerdem bringen wir in der Mitte dieses Empfangsdipols einen Widerstand mit 50 Ω an. Dann erhält das neue NEC-File folgende Form.

```
CM Lambda - Halbe Dipol
CE
'
SY freq=2375
SY dipolarm=300/freq/4
SY dr=0.002/2
SY x=0.907

GW 1  9    0  -dipolarm*x     0   0dipolarm*x  0     dr
GM 10  1         0  0  0            2  0  0            1
'
GE
'
LD 0  11  5  5  50
FR 0 1 0 0 freq
'
EK
'
EX 0 1 5 0 1 0
'
EN
```

Schauen wir uns die erste rote Zeile genauer an, denn das ergibt die zweite Antenne:

GM	= **Geometry move** = damit wird eine Kopie erzeugt
10	= Tag incement. Damit wird diese Antenne als „**tag 10+1 =tag 11**“ verwaltet
1	= wir wollen nur **eine Kopie** erzeugen
0 0 0	= keine Rotation um irgend eine Achse vorsehen
2 0 0	= verschiebe die Kopie um **2 Meter in X-Richtung**
1	= beginne den Kopiervorgang ab **tag 1** (= ursprüngliche Antenne)

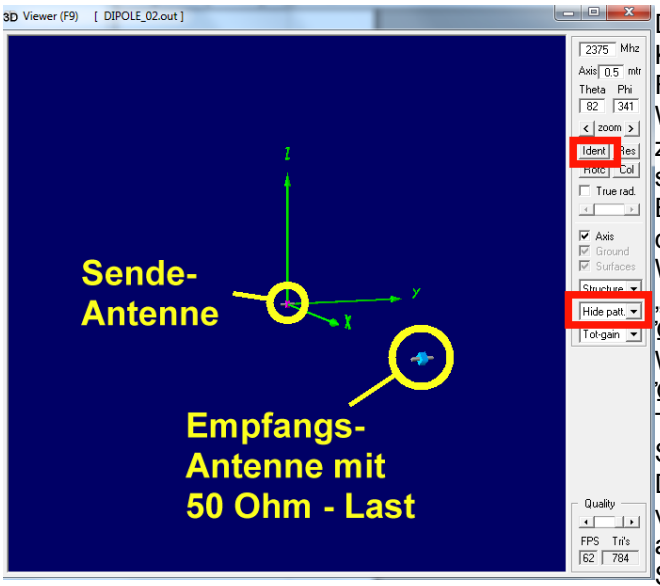
Und mit der zweiten roten Zeile fügen wir in der Mitte der neuen Empfangsantenne einen Lastwiderstand mit 50 Ω ein:

LD	= **Load card.** Damit können wir z. B. Bauteile hinzu fügen
0	= wir wollen eine RLC-Anordnung einfügen
11	= eingefügt wird in den Wire der neue Antenne, der **tag 11** (= 10 +1) heißt
5 5	= eingefügt wird zwischen Anfang und Ende von **Segment 5** (= Drahtmitte)
50	= eingefügt wird ein Ohmscher Widerstand mit 50Ω. L und C würden folgen, haben aber beide den Wert Null

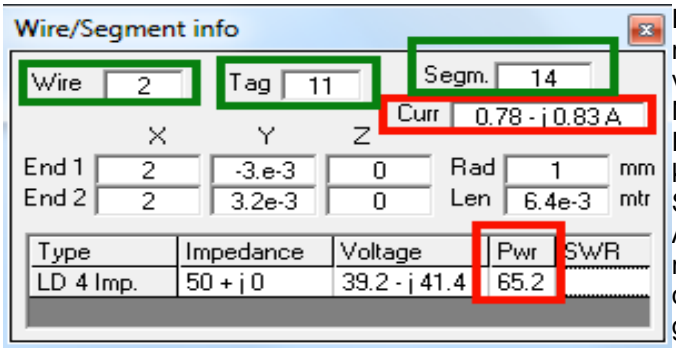
Zum Abschluss gehen wir in „Settings“ im Hauptmenü und öffnen „**Input Power**“. Dort geben wir eine **Sendeleistung von 1 Megawatt** (= 1000000 = 1e6) ein, damit sich auch in größeren Entfernungen noch etwas rührt und sind somit bereit für weitere Taten.

Bitte die Änderungen im NEC-File eintragen und wieder mit der „Zweistufen-Prozedur“ abschließen.
(Reihenfolge: Ändern / Speichern / Schließen / wieder öffnen / „Beenden“ anklicken)

3.2. Das Ergebnis der Fernfeld-Simulation



Die Simulation erfolgt exakt nach der Prozedur in Kapitel 2.2 – aber die 3D-Darstellung über die Taste F9 ist hier sehr hilfreich.
Wenn darin „**hide pattern**“ aktiv ist und wir gefühlvoll zoomen, dann haben wir dieses Bild vor uns. Darin sehen wir die Sendeantenne mit einer violetten Bauchbinde, die kennzeichnet die Einspeisung. Bei der Empfangsantenne sitzt dagegen ein hellblauer Würfel auf dem mittleren Segment – er stellt die „Load“, also den eingefügten Lastwiderstand mit 50 Ω, dar.
Wir interessieren uns für die Empfangsleistung im 50 Ω-Widerstand und klicken deshalb doppelt auf die Taste **„Ident**“. Da können wir das gewünschte Segment eingeben und es trägt die **Nummer „14**“. Das ist einfach zu verstehen, denn alle vorkommenden Segmente werden der Reihe nach aufgelistet und jede Antenne besteht aus 9 Segmenten. Also finden wir die Mitte der Empfangsantenne bei „9 + 5 = 14“.



Hier sehen wir das Ergebnis und wir finden darin neben allen übrigen Daten eine Empfangsleistung von 65 Watt sowie einen Strom von 0,78 - j0,83 A. Nun können wir prüfen, ob wir uns wirklich im Fernfeld befinden und ob 4NEC2 die Verhältnisse korrekt berechnet. Dazu wiederholen wir die Simulation mehrmals und verdoppeln jeweils den Abstand der beiden Antennen (also 4 m / 8 m / 16 m usw. bei X eintragen). Das geht sehr einfach, denn wir brauchen nur beim Editor mit dem geöffneten NEC-File in dieser Zeile die rot gekennzeichnete Eingabe der X – Verschiebung zu ändern, zu speichern, nochmals zu starten, „Beenden“ anzuklicken und dann zu simulieren. Jetzt kann man den Strom in Segment 14 ansehen:

GM 10 1 0 0 0 2 0 0 1

Die Ergebnisse lauten

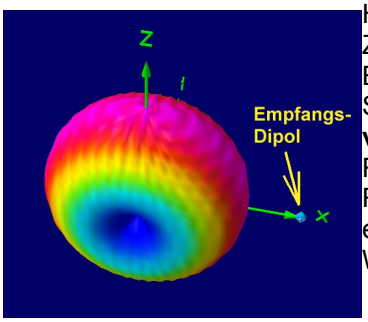
für 2m:	**I = 0,78 – j0,83**	**(P =65,2 W)**
für 4m: =	**I = 0,4 +j0,1**	**(P = 16,3 W)**
für 8m:	**I = -0,2 +j0,24**	**(P = 4,03 W)**
für 16m:	**I = 0,14 -j0,03**	**(P = 1,02W)**

Man sieht, dass die Feldstärken und die reellen Stromanteile etwa linear mit steigendem Abstand abnehmen und damit die Leistungen quadratisch sinken. Das entspricht genau der Theorie und stärkt das Vertrauen zum Programm doch sehr.....

Noch ein Test ist interessant: wenn man beim Empfangsdipol eine **Rotation von 90 Grad um die X – Achse** vornimmt (= seine Polarisation von Horizontal auf Vertikal ändert), sollte die **Empfangsleistung auf Null** sinken. Also geht man wieder ins NEC-File und sieht eine entsprechende **Rotation um die X – Achse** bei der GM-Karte (mit der vorgeschriebenen folgenden Speicherprozedur) vor:

GM 10 1 90 0 0 2 0 0 1

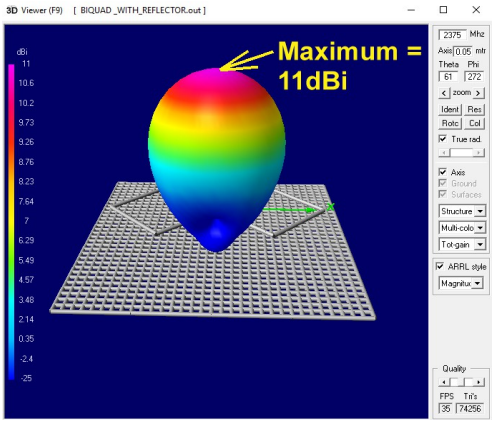
Die Simulation bestätigt das anschließend sehr schön.

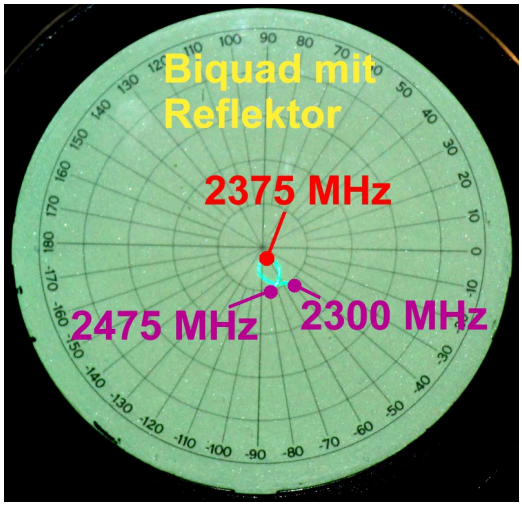


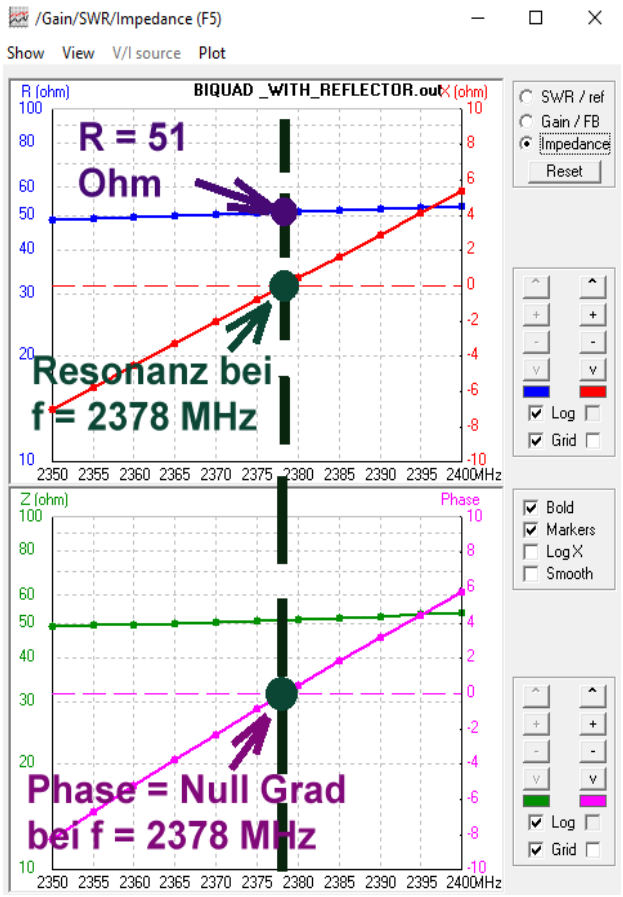
Hinweis:
Zur Vorschrift „**Messung nur im Fernfeld“** sollte man sich zusätzlich dieses Bild ansehen. Dazu wurde der Abstand der Empfangsantenne zum Sendedipol von **2 m (= 16 Wellenlängen) auf 1 m (= 8 Wellenlängen) vermindert** und da sieht man, dass die zweite Antenne doch deutliche Rückwirkungen verursacht. Dadurch entsteht eine deutliche „Kräuselung“ des Richtdiagramms beim Sendedipol. Bei 2 m Abstand ist das noch leidlich erträglich, aber so richtig gut wird das erst bei deutlich größeren Abständen. Wer hätte das gedacht....?

4. Ein Bespiel für Fortgeschrittene: eine Biquad-Antenne für 2375 MHz

So sieht das Strahlungsdiagramm für 2375 MHz aus... ….und so die Eingangsimpedanz



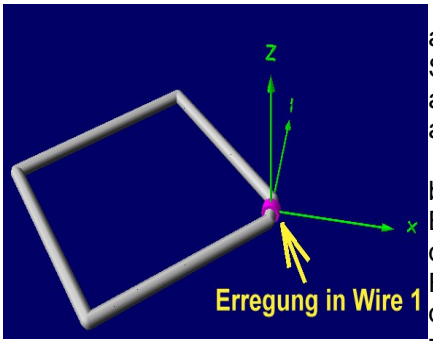




Die Messung mit dem Vektoriellen Network-Analyser bestätigt die rechten Verläufe.

4.1. Der Antennenentwurf

Da fangen wir mit einer einfachen Quad-Antenne an und setzen sie aus 5 Drähten zusammen:



a) einem nur 4 mm langen Drahtstück mit 2 mm Durchmesser zur Speisung der Antenne (= Feed).. Es ist in Richtung der y-Achse angeordnet, beginnt bei „-2 mm“ und endet bei „+2 mm“. Er besteht nur aus einem einzigen Segment.

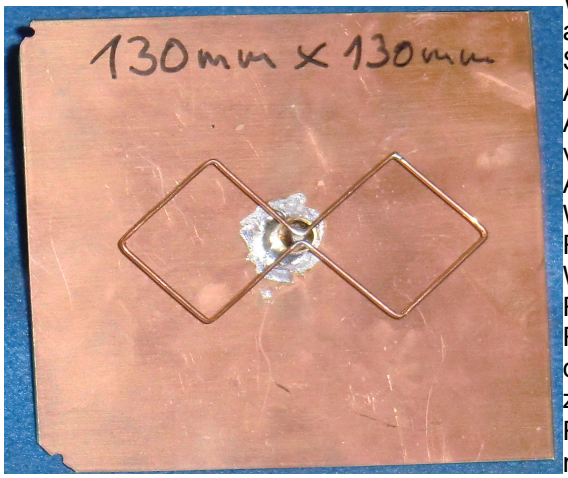
b) Dazu kommen die 4 Drähte (Länge jeweils = λ/4 und 8 Segmente) zur Bildung des Quadrats. Vorschriftsmäßig verbinden wir ein Drahtende mit dem nächsten Drahtanfang und starten bei „+2 mm“, also am Ende des Feeds. Einmal im Ring herum landen wir so schließlich bei „-2 mm“ und damit am Anfang des Feeds. Das Bild zeigt die Anordnung und das zugehörige NEC-File ist nun leicht zu verstehen:

```
CM Quad 2375 MHz
CE
SY FREQ = 2375
SY dr = 0.002/2          'Radius des Kupferdrahtes =  1 mm
SY x = 300/freq/4*sin(45)

' Strahler
GW   1   1   0    -0.002   0    0     0.002   0    dr          Feed.
GW   2   8   0    0.002    0    -x    x       0    dr          Nun folgen vier
GW   3   8   -x   x        0    -2*x  0       0    dr          Drahtstücke
GW   4   8   -2*x 0        0    -x    -x      0    dr          zur Bildung des
GW   5   8   -x   -x       0    0     -0.002  0    dr          Quadrates
GE   0                                                        „Null“ bedeutet „Free Space“

EK
EX   0   1   1    0        1    0
FR   0   1   0    0        FREQ
EN
```

Dann kopieren wir einfach diese Quad-Struktur und drehen sie dabei um 180 Grad um die Y-Achse. Das geschieht durch die rote zusätzlich eingefügt Zeile nach den „GW“-Cards:

```
' Strahler
GW   1   1   0    -0.002   0    0     0.002   0    dr
GW   2   8   0    0.002    0    -x    x       0    dr
GW   3   8   -x   x        0    -2*x  0       0    dr
GW   4   8   -2*x 0        0    -x    -x      0    dr
GW   5   8   -x   -x       0    0     -0.002  0    dr
GM   4   1   0    180      0    0     0       0    2
GE   0
```

Da haben wir schon mal unsere Biquad fertig und es fehlt nur noch der Reflektor.



Wir gehen von einem symmetrischen Betrieb der Antenne aus und in Praxis erfolgt die Symmetrierung durch einen λ/4-Sperrtopf. Er besteht aus einem Messingröhrchen mit 10 mm Außen- und 8,3 mm Innendurchmesser aus dem Baumarkt. Als Innenleiter dient der Kupfermantel des als Zuleitung verwendeten Semi-Rigid-Kabels mit 3,55 mm Außendurchmesser. So erhält man einen Topf mit einen Wellenwiderstand von fast genau 50 Ω und eine Röhrchenlänge von 31,7 mm entspricht einem Viertel der Wellenlänge. Ein Kupferscheibe wird dann über das Semi-Rigid- Kabel geschoben und sowohl mit dem Ende des Röhrchens wie auch mit dem Kabelmantel verlötet. Fertig ist der Kurzschluss am Ende des Topfes! Und die Theorie sagt zusätzlich, dass der Abstand zwischen Antenne und Reflektor ca. λ/8 = 16 mm betragen muss. So genau muss man schließlich das Röhrchen in die als Reflektor dienende, doppelseitig beschichtete Leiterplatte einlöten.

4.2. Der Reflektor-Entwurf

Da können wir zwischen zwei Möglichkeiten wählen:

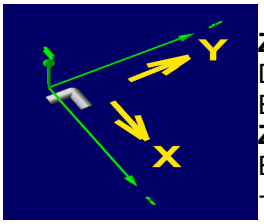
a) Wir arbeiten mit dem **mitgelieferten 4NEC2- Zusatzprogramm „build**“ und erstellen einen „**Patch**“ mit **128 mm x 128 mm**. „Patch“ ist in „build“ fest eingeplant und der Entwurf ein Kinderspiel. Allerdings wird anschließend im damit erzeugten NEC-File jeder einzelne Wire mit seinen Daten aufgelistet und bei über 2000 Drähten ergibt das viele Seiten an erzeugtem Code.

b) Man fängt mit einer sehr simplen Struktur aus **nur zwei Drähten an und arbeitet mit mehrfachem geschickten Kopieren**, bis endlich der Reflektor (mit kleinen Kompromissen) fertig ist. **Dafür reichen nur 5 weitere Codezeilen im NEC-File** – aber die haben es in sich und erfordern Denkaufwand beim Entwurf oder der Fehlersuche. Aber kürzer und knapper geht es nicht mehr...

Nehmen wir die Möglichkeit b) , denn sie stellt die Minimierung des NEC-Files durch einen Experten (= Hardy Lau, DL1GLH) dar!

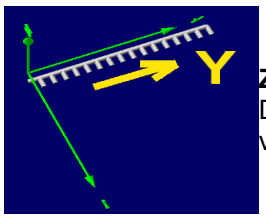
```
CM Biquad 2375MHz with reflector
CE

SY FREQ = 2375
SY dr = 0.002/2                          ' Radius des Kupferdrahtes = 1 mm
SY x = 300/freq/4*sin(45)*0.9928         ' λ/4 – Element mit Korrekturfaktor
SY rh = 0.016                            ' Abstand Reflektor-Dipol = 16mm

' Strahler
GW   1   1   0     -0.002   0     0      0.002   0   dr
GW   2   8   0     0.002    0     -x     x       0   dr
GW   3   8   -x    x        0     -2*x   0       0   dr
GW   4   8   -2*x  0        0     -x     -x      0   dr
GW   5   8   -x    -x       0     0      -0.002  0   dr
GM   4   1   0     0        180   0      0       0   2
'Reflektor
GW   10  1   0     0        -rh   0      0.004   -rh  dr    A
GW   11  1   0     0.004    -rh   0.004  0.004   -rh  dr    B
GM   2   15  0     0        0     0      0.004   0    10    C
GM   40  15  0     0        0     0.004  0       0    10    D
GM   700 3   0     0        90    0      0       0    10    E
GE

EK
EX   0   1   1     0        1     0
FR   0   1   0     0        FREQ
EN
```

Sehen wir uns den Reflektorentwurf genau an:

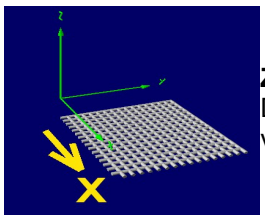


Zeile A:
Damit wird ein Wire mit 2 mm Durchmesser von y = Null mm nach y = +4 mm definiert. Er verläuft in den Ebenen x = Null und z = -16 mm

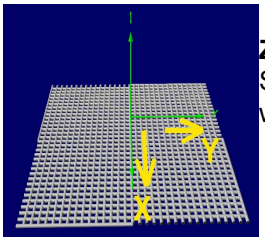
Zeile B:
Ein zweiter Wire startet (in derselben z-Höhe) bei x = 0 / y = +4 mm und führt nach x = +4 mm / y = +4 mm. Er steht damit senkrecht auf dem Wire von Zeile „A“ .



Zeile C:
Dieses Gebilde wird 15x kopiert und bei jedem Kopierschritt um 4 mm in Y-Richtung verschoben



Zeile D:
Das Ergebnis von „C“ wird erneut 15x kopiert, aber jede Kopie um 4 mm in X-Richtung verschoben



Zeile E:
Schließlich folgen nochmals 3 Kopien, wobei jede um 90 Grad um die Z-Achse gedreht wird. Das wärs!