SIX MTR 4 EL 6 BAND QUAD.doc

Bob Hume   KG6B   07-04-2005

1.) Comparison of Actual 6 MTR Quad Antenna Array Tuning to EZNEC 4.0 Predictions

A mono band 4 element 6 MTR quad was constructed with #12 bare copper wire with the following (+/-0.2 inch tolerance) dimensions:

Diamond driven element loop length=20.16427 FT
Reflector length (i.e. compared to driven element) = +3.00%
First director length = -1.9%
Second director length = -1.7%
Reflector to driven element spacing = 4.0 FT
Driven element to first director spacing = 5.0 FT
First director to second director spacing = 5.0 FT
14 foot boom made of 3 inch diameter aluminum tubing with no guying was used.
Quad wire support arms made of 0.5 inch diameter solid fiberglass rod.
Arm wire holes backed off 1/8 inch from taught square position for some wire slack.
Antenna placed on a self supporting tower at a 38 foot height over ground

An MFJ-259B impedance meter was used to measure the quad array minimum SWR frequency looking into a 110 foot feed of RG 213 A/U 50 ohm coax. Figure 1A shows the measured and EZNEC modeled SWR versus frequency plots. The measured minimum SWR was 1.15 at a frequency of 49.915 Mhz. This antenna (with 38 foot height over real ground) was modeled in EZNEC and resulted in a predicted minimum SWR of 1.10245 at a frequency of 50.222 Mhz. (The EZNEC predicted array resonant frequency was 50.253 Mhz) The real antenna tuning is thus a factor of (49.915/50.222)=0.99388714, or 0.6113 percent, or 306 khz lower than that predicted by EZNEC. The frequency measurement made at the minimum SWR point of 1.15 should not be sensitive to the feed line length or the precision limitations of the MFJ-259B meter. These become significant factors at higher SWR values where the antenna load reactance is high and or its resistance is substantially different than the 50 ohm line characteristic impedance.

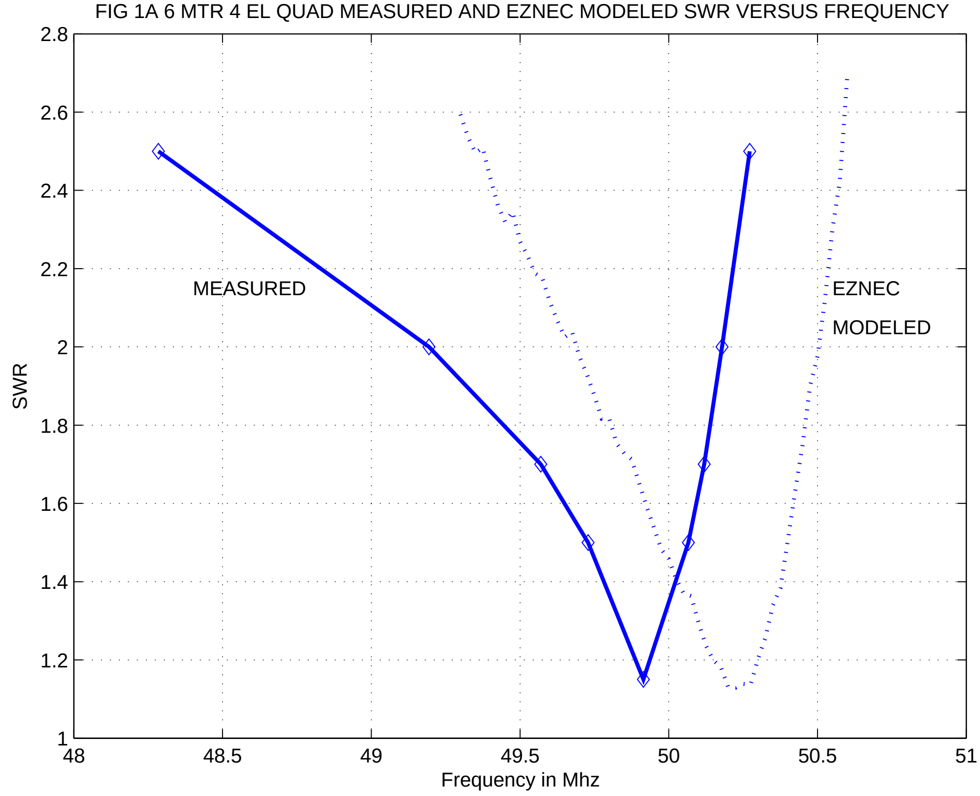
One would thus reduce or shorten the actual quad loop wire lengths by 0.6113 percent from those in EZNEC to get the tuning results predicted by EZNEC for the real world antenna. For the subject 6 MTR beam this requires a 1.48 inch reduction in all the element loop lengths. This probably also applies to tuning the six band version of this 6 MTR quad array. Some future six band quad array measurements are planned to check this out.

If the +0.6113 percent 6 MTR quad tuning difference of EZNEC scales to quad antennas made for other HF bands then the following applies.

MTR BAND	F Mhz	EZNEC QUAD TUNING DIFFERENCE* IN KHZ	
6	50.22	+306	MEASURED
10	28.45	+173	SCALED
12	24.93	+152	"
15	21.20	+129	"
17	18.11	+110	"
20	14.15	+86	"

* EZNEC QUAD TUNING DIFFERENCE= $(F_{EZNEC}-F_{REAL\ WORLD})$

FIG 1A 6 MTR 4 EL QUAD MEASURED AND EZNEC MODELED SWR VERSUS FREQUENCY



The EZNEC quad tuning difference may be related to the real versus modeled current distribution at the corners of the square loops. The constructed loops have slightly rounded corners compared to the EZNEC perfectly square corners. The boom, tower, and feed line were not modeled in EZNEC. For these reasons, I call it a tuning "difference" of EZNEC rather than a tuning "error" of EZNEC. (Although it could in fact turn out to be an EZNEC modeling error.) The tuning difference may be peculiar to the geometric shape of the quad loops (i.e. tuning error could be different for circular or triangular shaped loops). The SWR curve shape below the minimum SWR frequency is more generous for the Figure 1A measured curve than for the EZNEC modeled curve. The measurements at the higher SWR level may have error however for reasons previously mentioned.

The subject test quad has only a +0.71% and -2.38% bandwidth between the lower and upper SWR=2.5 points and the minimum SWR frequency. The EZNEC 0.6113% tuning difference becomes significant when it is the goal to put the minimum SWR point near the 6 MTR lower band edge. The six band quad has 20 band loops. It is very difficult to get at all these loops for retuning once the antenna is on the mast. The goal in using EZNEC modeling is to correctly size the loops on the ground so that retuning is not required after the antenna is on the mast.

The six band version of this 6 MTR quad has a driven element length times minimum SWR frequency product of K=1011.6076 Mhz *Feet when modeled on EZNEC with an antenna height of 55 foot over ground. Based on the above real world tuning factor results, this product should be reduced by 0.6113 percent to K= 1005.4238 for actual driven element wire sizing using the formula DE=K/(Fmin swr). When the actual element lengths are used in EZNEC, the real world antenna tuning will be 306 khz lower than indicated by EZNEC.

The mono band version of this 6 MTR quad has a driven element length times minimum SWR frequency product of K=1012.5690 Mhz *Feet when modeled on EZNEC with an antenna height of 55 foot over ground. Based on the above real world tuning factor results, this product should be reduced by 0.6113 percent to K= 1006.3793 for actual driven element wire sizing using the formula DE=K/(Fmin swr). When the actual element lengths are used in EZNEC, the real world antenna tuning will be 306 khz lower than indicated by EZNEC.

To optimize performance on the low end of the 6 MTR band, I am choosing the minimum SWR design frequency (Fmin swr) to be 50.30 Mhz. This gives some pad for element stretching effects and cutting error since the SWR increases rapidly to 2.5 only 355 khz above (Fmin swr).

The EZNEC model for the 6 MTR mono band test antenna follows. Forty one segments per modeled wire were used.

EZNEC+ ver. 4.0

6 MTR 4 EL MONO BAND QUAD 6/25/2005 11:01:21 AM

--------------- ANTENNA DESCRIPTION ---------------

Frequency = 50.15 MHz
Wire Loss: Copper -- Resistivity = 1.74E-08 ohm-m, Rel. Perm. = 1

--------------- WIRES ---------------

No.		End 1 Coord. (ft)				End 2 Coord. (ft)			Dia (in)	Segs	Insulation	
	Conn.	X	Y	Z	Conn.	X	Y	Z			Diel C	Thk(in)
1	W4E2	16,	0,	51.3285	W2E1	16,	3.67151,	55	.080827	41	1	0
2	W1E2	16,	3.67151,	55	W3E1	16,	0,	58.6715	.080827	41	1	0
3	W2E2	16,	0,	58.6715	W4E1	16,	-3.6715,	55	.080827	41	1	0
4	W3E2	16,	-3.6715,	55	W1E1	16,	0,	51.3285	.080827	41	1	0
5	W8E2	20,	0,	51.4354	W6E1	20,	3.56457,	55	.080827	41	1	0
6	W5E2	20,	3.56457,	55	W7E1	20,	0,	58.5646	.080827	41	1	0
7	W6E2	20,	0,	58.5646	W8E1	20,	-3.5646,	55	.080827	41	1	0
8	W7E2	20,	-3.5646,	55	W5E1	20,	0,	51.4354	.080827	41	1	0
9	W12E2	25,	0,	51.5032	W10E1	25,	3.49685,	55	.080827	41	1	0
10	W9E2	25,	3.49685,	55	W11E1	25,	0,	58.4968	.080827	41	1	0
11	W10E2	25,	0,	58.4968	W12E1	25,	-3.4968,	55	.080827	41	1	0
12	W11E2	25,	-3.4968,	55	W9E1	25,	0,	51.5032	.080827	41	1	0
13	W16E2	30,	0,	51.496	W14E1	30,	3.50398,	55	.080827	41	1	0
14	W13E2	30,	3.50398,	55	W15E1	30,	0,	58.504	.080827	41	1	0
15	W14E2	30,	0,	58.504	W16E1	30,	-3.504,	55	.080827	41	1	0
16	W15E2	30,	-3.504,	55	W13E1	30,	0,	51.496	.080827	41	1	0

Total Segments: 656

-------------- SOURCES --------------

No.	Specified Pos.		Actual Pos.		Amplitude	Phase	Type
	Wire #	% From E1	% From E1	Seg	(V/A)	(deg.)	

1	5	0.00	1.22	1	1	0	SI

No loads specified

No transmission lines specified

Ground type is Real, High-Accuracy

--------------- MEDIA ---------------

No.	Cond. (S/m)	Diel. Const.	Height (ft)	R Coord. (ft)
1	0.005	13	0	0

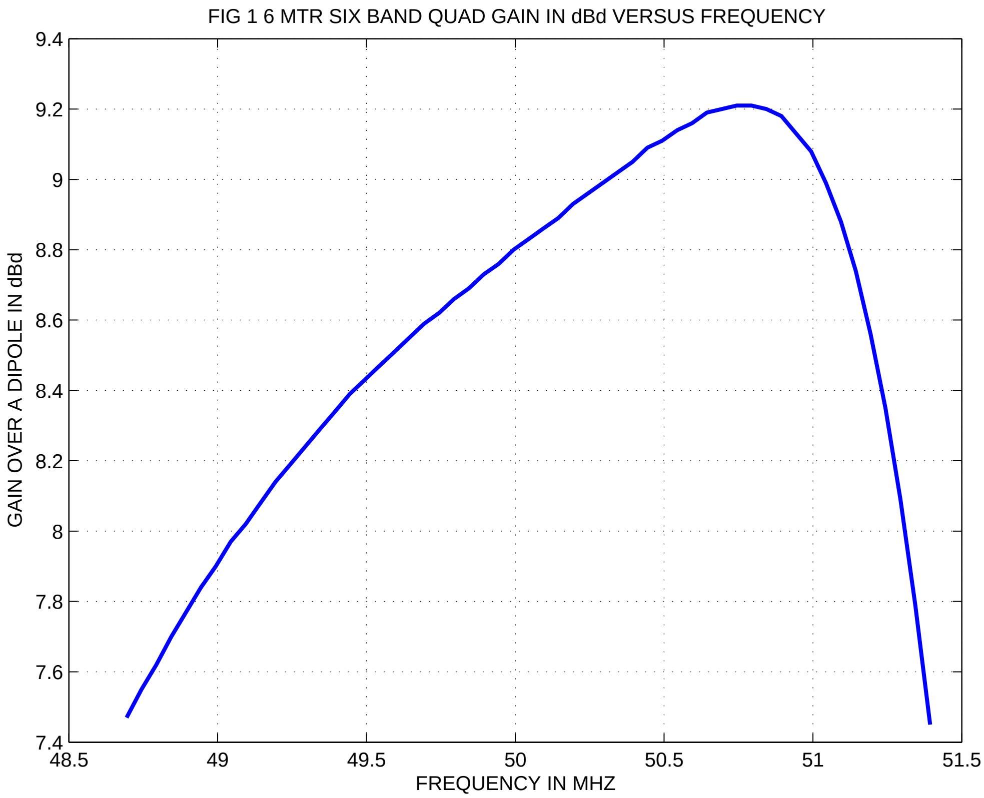
2.) 6 MTR 4 ELEMENT SIX BAND QUAD BASELINE DESIGN AND PERFORMANCE

Based on the experimental and analytical results of section 1.0, the driven element loop length of the six band quad is set to DE = (K/Fmin swr) where K=1005.4238 Mhz* Ft and Fmin swr=50.30 Mhz. This results in a driven element loop length DE= 19.9885 feet. The baseline 6 MTR 4 element six band quad design is then:

Diamond driven element loop length = 19.9885 FT
Reflector length (i.e. compared to driven element) = +3.00%
First director length = -1.9%
Second director length = -1.7%
Reflector to driven element spacing = 4.0 FT
Driven element to first director spacing = 5.0 FT
First director to second director spacing = 5.0 FT

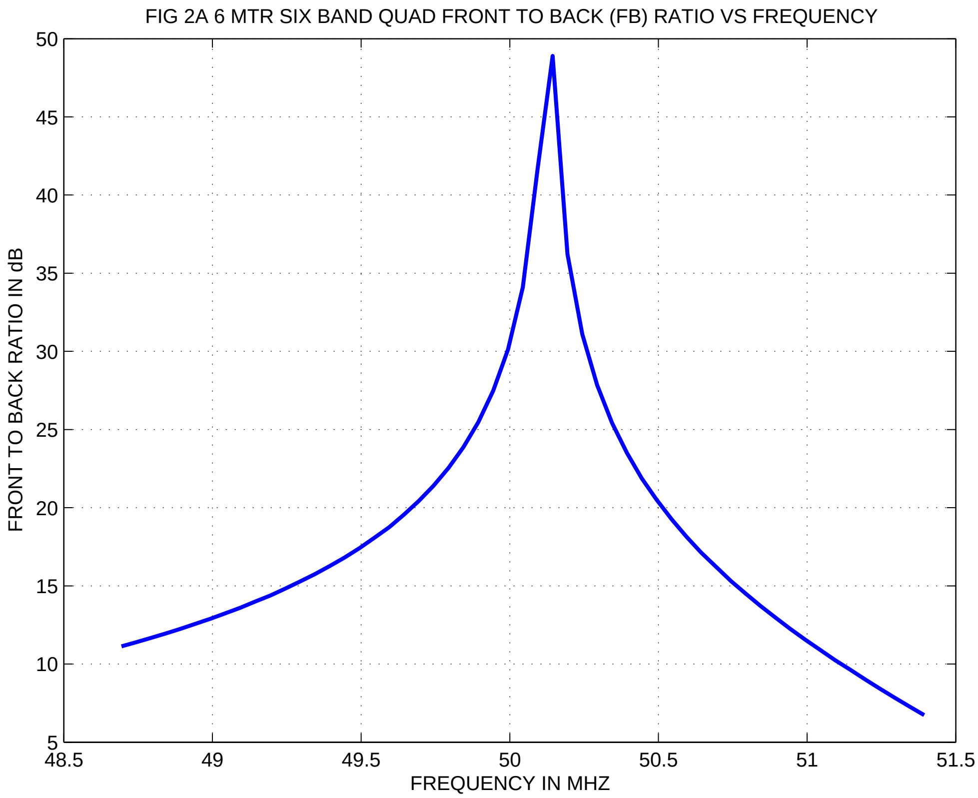
Based on the results of section 1.0, when these dimensions are modeled in EZNEC (along with the other 5 bands of quad wires), one must subtract 0.306 Mhz from the 6 MTR frequency used in EZNEC to get the adjusted "real world" response of the 6 MTR antenna. All of the performance plots for the antenna that follow have made this EZNEC frequency difference adjustment.

The 6 MTR six band quad gain in dBd over a dipole as a function of frequency is shown in Figure 1. A reference dipole at the same 55 foot antenna height and 5 degree vertical wave angle has a gain of 7.8 dBi. Thus, the Figure 1 dBd gain can be converted to dBi gain by adding 7.8 dB



MAX GAIN FREQ IN MHZ=50.768
MAX GAIN IN dBi=17.0117
MAX GAIN IN dBd=9.2117

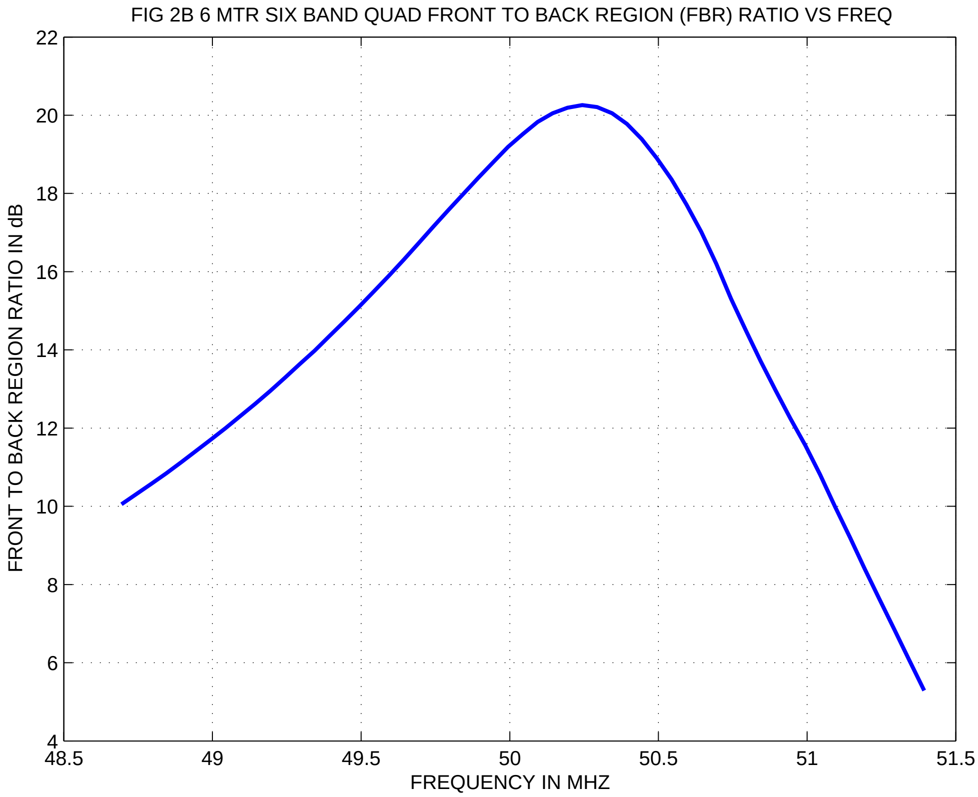
The 6 MTR six band quad front to back (FB) ratio as a function of frequency is shown in Figure 2A.



MAX FB FREQ IN MHZ = 50.138
MAX FB IN dB = 49.1093

The 6 MTR six band quad front to back region (FBR) ratio as a function of frequency is shown in Figure 2B. The back region picks up the maximum lobe in the 180+/-90 degree region from the antenna heading. The quad azimuth pattern can frequently have a pair of large "Mickey Mouse" ears in the back region with a good FB ratio.

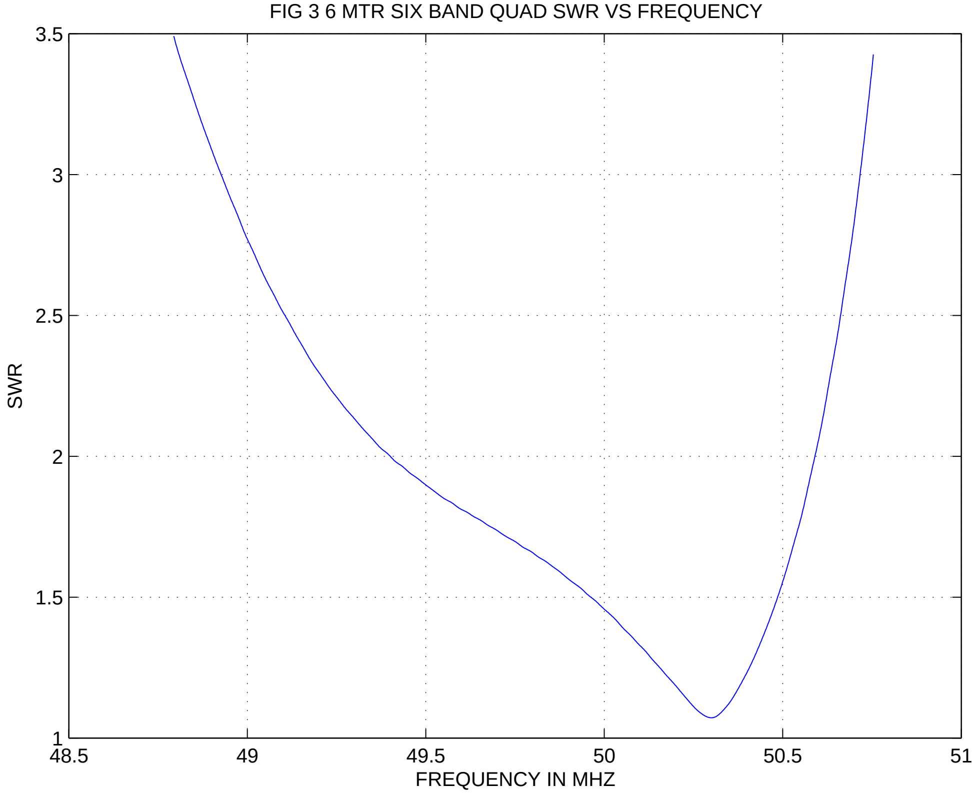
Section 3 has the listing of a MATLAB program named sixMTR6B.m used to generate Figures 1, 2A, and 2B.



MAX FBR FREQ IN MHZ = 50.249
MAX FBR IN dB = 20.2607

The 6 MTR six band quad SWR versus frequency plot is shown in Figure 3. The plot is based on a 50 ohm coaxial cable antenna feed line. Section 3 has the listings of MATLAB programs named zcon.m and swrQ.m used to generate Figure 3 by cut and paste of the EZNEC LastZ.txt file data to a matrix z in the MATLAB working space. Figure 3 verifies the minimum SWR design frequency of 50.30 Mhz.

FIG 3 6 MTR SIX BAND QUAD SWR VS FREQUENCY



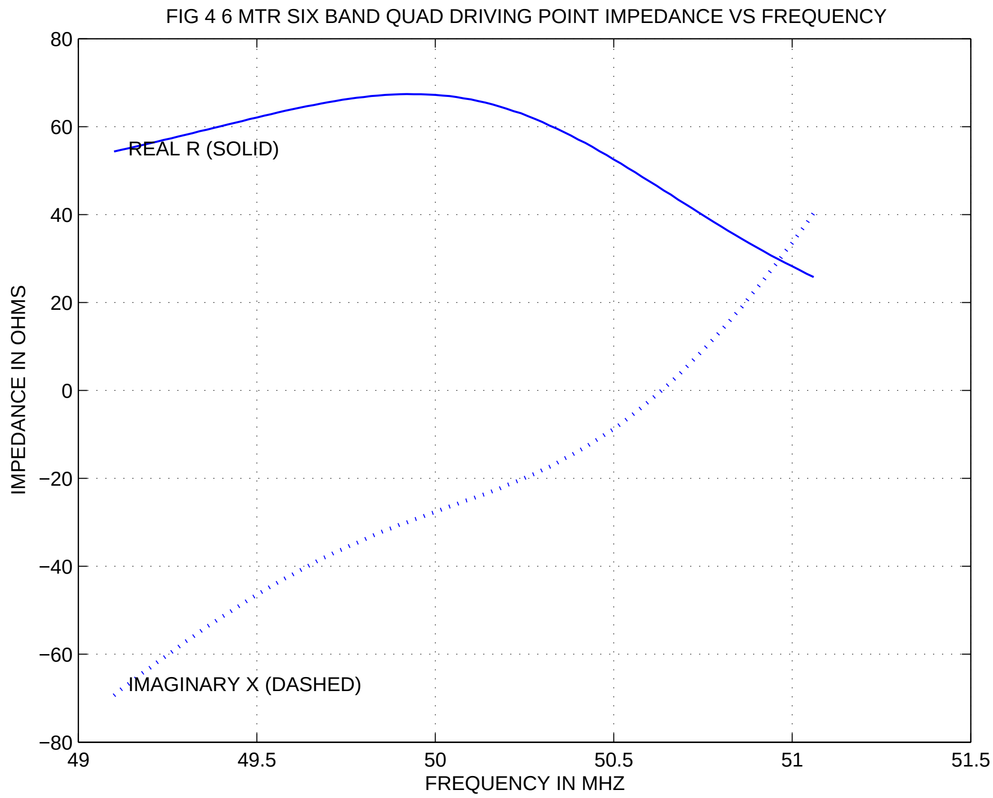
Specific Figure 3 frequency points of interest are:

@ ADJUSTMENT TO EZNEC TUNING IN MHZ = -0.306
LOWER SWR = 3.0   POINT FREQUENCY = 48.927
LOWER SWR = 2.5   POINT FREQUENCY = 49.106
MINIMUM SWR FREQUENCY = 50.300   MIN SWR = 1.0719
UPPER SWR = 2.5   POINT FREQUENCY = 50.662
UPPER SWR = 3.0   POINT FREQUENCY = 50.717

48.9270	49.1060	50.3000	50.6620	50.7170	FREQUENCIES MHZ
3.0	2.5	1.0719 (MIN SWR)	2.5	3.0	SWRs
-2.7296	-2.3738	0	0.7197	0.8290	% FREQ CHANGES

SWR<2.5 PERCENT BANDWIDTH = 3.0934
SWR<3.0 PERCENT BANDWIDTH = 3.5586

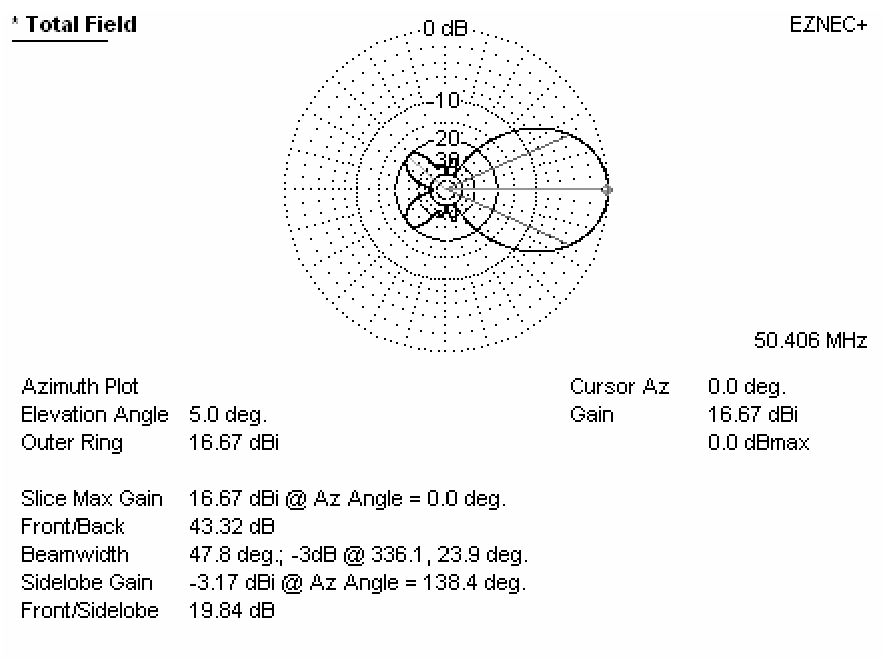
The 6 MTR six band quad driving point impedance real and imaginary parts versus frequency plots are shown in Figure 4. MATLAB program zcon.m generated this figure.



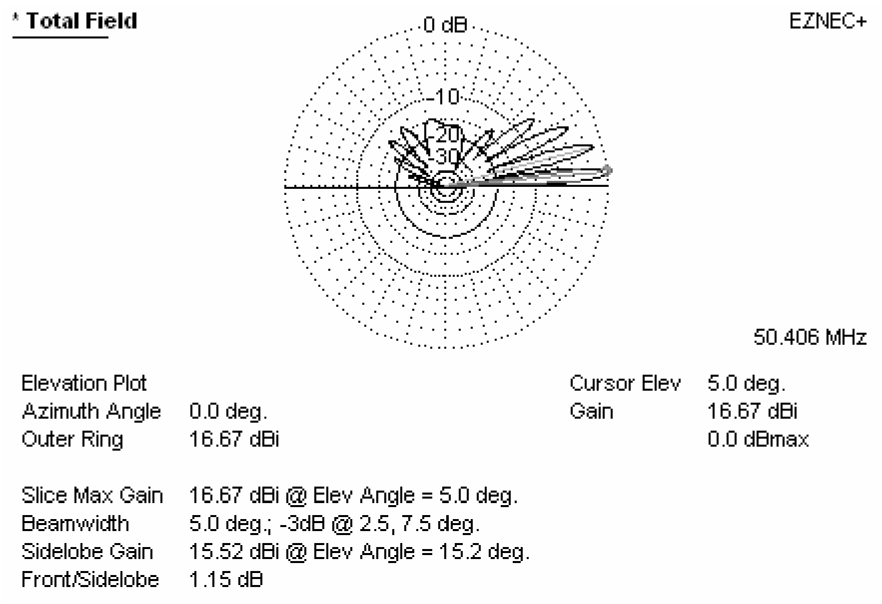
Resonant Frequency in MHZ = 50.634

Resonant Resistance in Ohms = 45.824

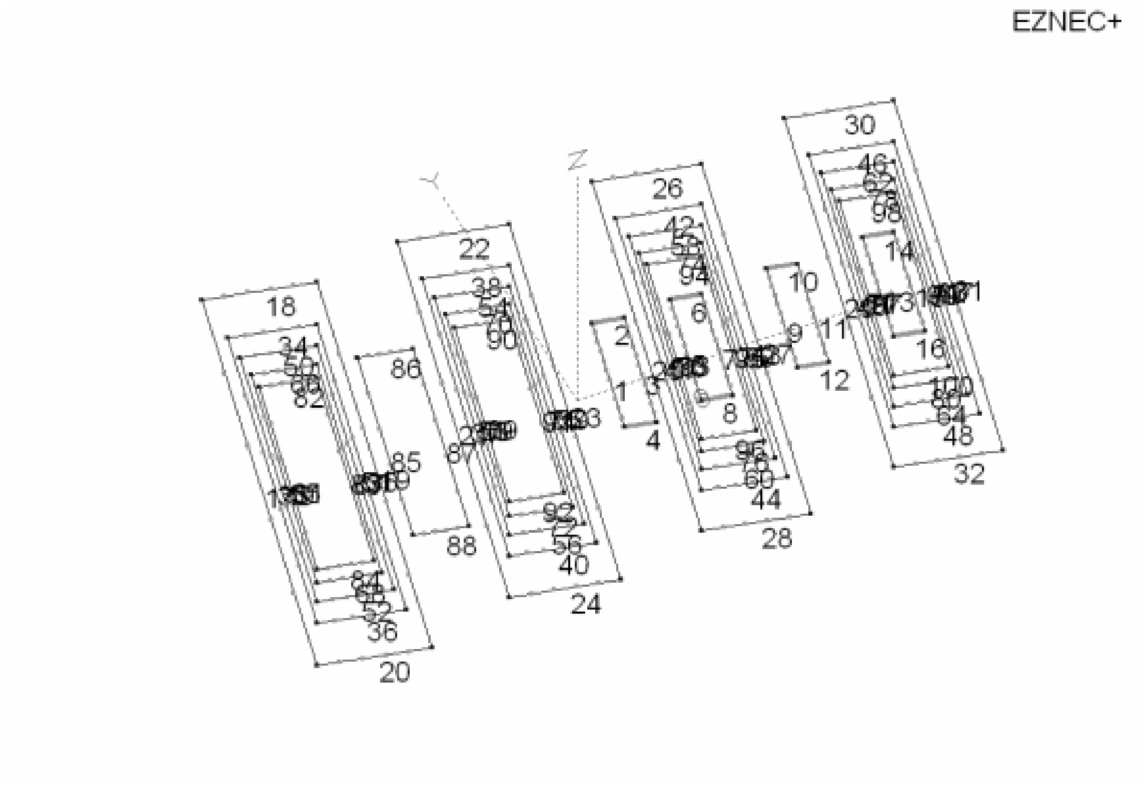
The azimuth gain plot of the 6 MTR six band quad at 50.1 Mhz follows. Note that the EZNEC frequency must be set 0.306 Mhz higher based on the tuning difference results of Section 1.



The elevation pattern of the 6 MTR six band quad at 50.1 MHZ follows. Note that the EZNEC frequency must be set 0.306 Mhz higher



A diagram of the six band quad antenna follows. The 4 element 6 MTR quad loops are located on the right half of the X axis boom. Two of the four loops require special 6 MTR quad support arms.



Construction details for the 6 MTR six band quad were derived by running a MATLAB program named quad6mtr.m (See Section 3 for quad6mtr.m program listing) that follows. The wire mark points include the corner locations and reference marks on each side of the soldering point that allow exact sizing of the loop length before and after soldering. See the two figures that define the wire marks for driven and non driven elements that follow the computer printout results. The wire is held taught and placed next to a tape and marked at the indicated tape points.

```
>> quad6mtr

6 MTR QUAD OPPOSING ARM PAIR BUTT TO BUTT SEPARATION IN
INCHES=3.875

6 MTR QUAD ARM WIRE HOLES FROM BUTT ENDS

EL      FT     INCH    1/16 INCH INCREMENTS (NUMBER OF)

REF     3      5       7
DE      3      4       1
DIR1    3      3       6
DIR2    3      3       6

6 MTR QUAD ELEMENT WIRE LENGTHS

 EL     DECIMAL FT

REF     20.58820
DE      19.98854
DIR1    19.60876
DIR2    19.64874
```

NON DRIVEN ELEMENT WIRE MARK POINTS FOLLOW

QUAD SPIDER NUMBER=1   6 MTR ELEMENT= REF

(p9-p2) in FT=20.5882   ELEMENT LENGTH in FT=20.5882

PT#	FT	INCH	1/16th INCH UNITS
1	1	0	0
2	2	0	0
3	2	6	0
4	3	6	0
5	8	7	12
6	13	9	8
7	18	11	5
8	22	1	1
9	22	7	1
10	23	7	1

QUAD SPIDER NUMBER=3   6 MTR ELEMENT=DIR1

(p9-p2) in FT=19.6088   ELEMENT LENGTH in FT=19.6088

PT#	FT	INCH	1/16th INCH UNITS
1	1	0	0
2	2	0	0
3	2	6	0
4	3	6	0
5	8	4	13
6	13	3	10
7	18	2	8
8	21	1	5
9	21	7	5
10	22	7	5

QUAD SPIDER NUMBER=4   6 MTR ELEMENT=DIR2

(p9-p2) in FT=19.6487   ELEMENT LENGTH in FT=19.6487

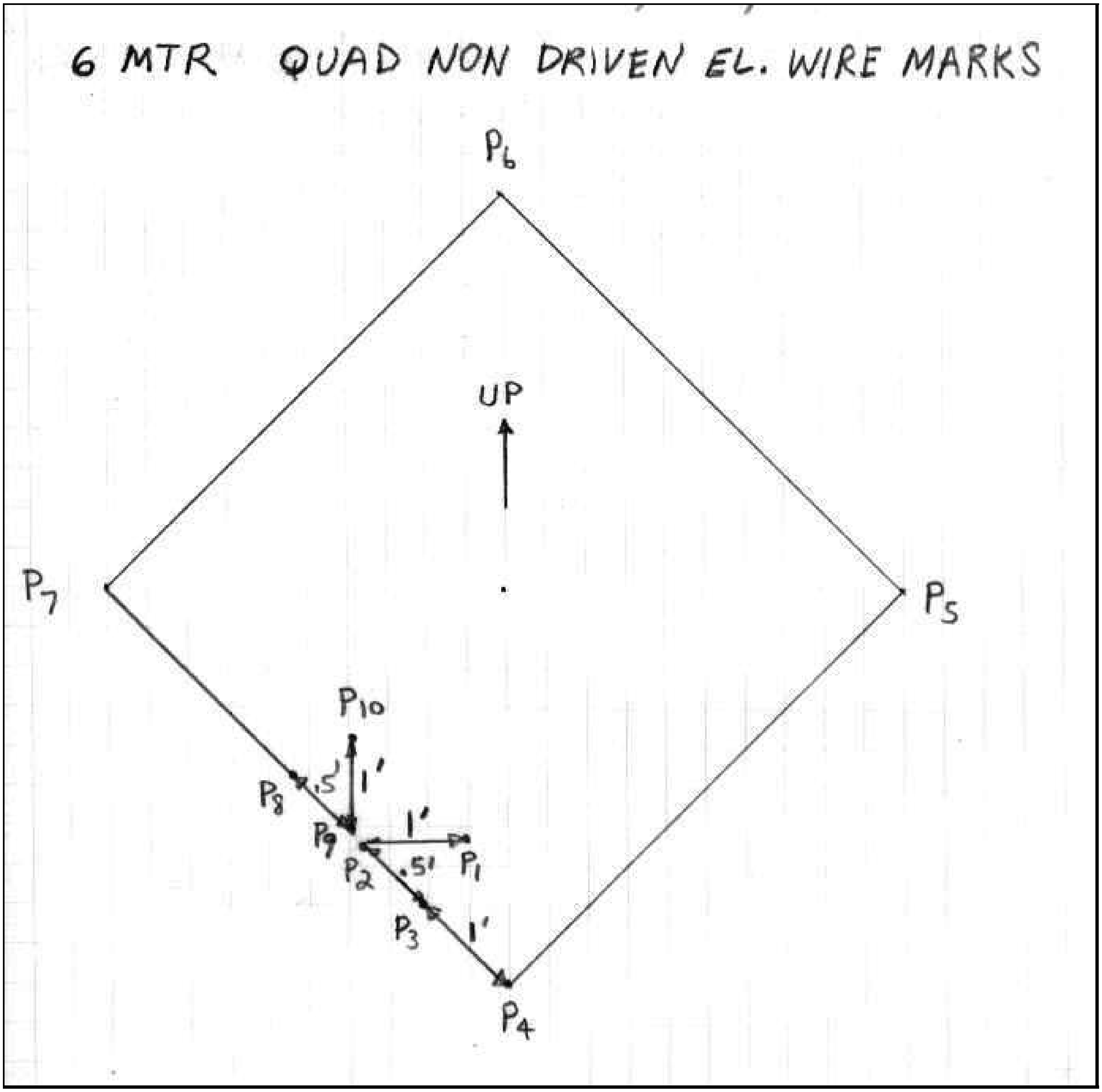
PT#	FT	INCH	1/16th INCH UNITS
1	1	0	0
2	2	0	0
3	2	6	0
4	3	6	0
5	8	4	15
6	13	3	14
7	18	2	13
8	21	1	13
9	21	7	13
10	22	7	13

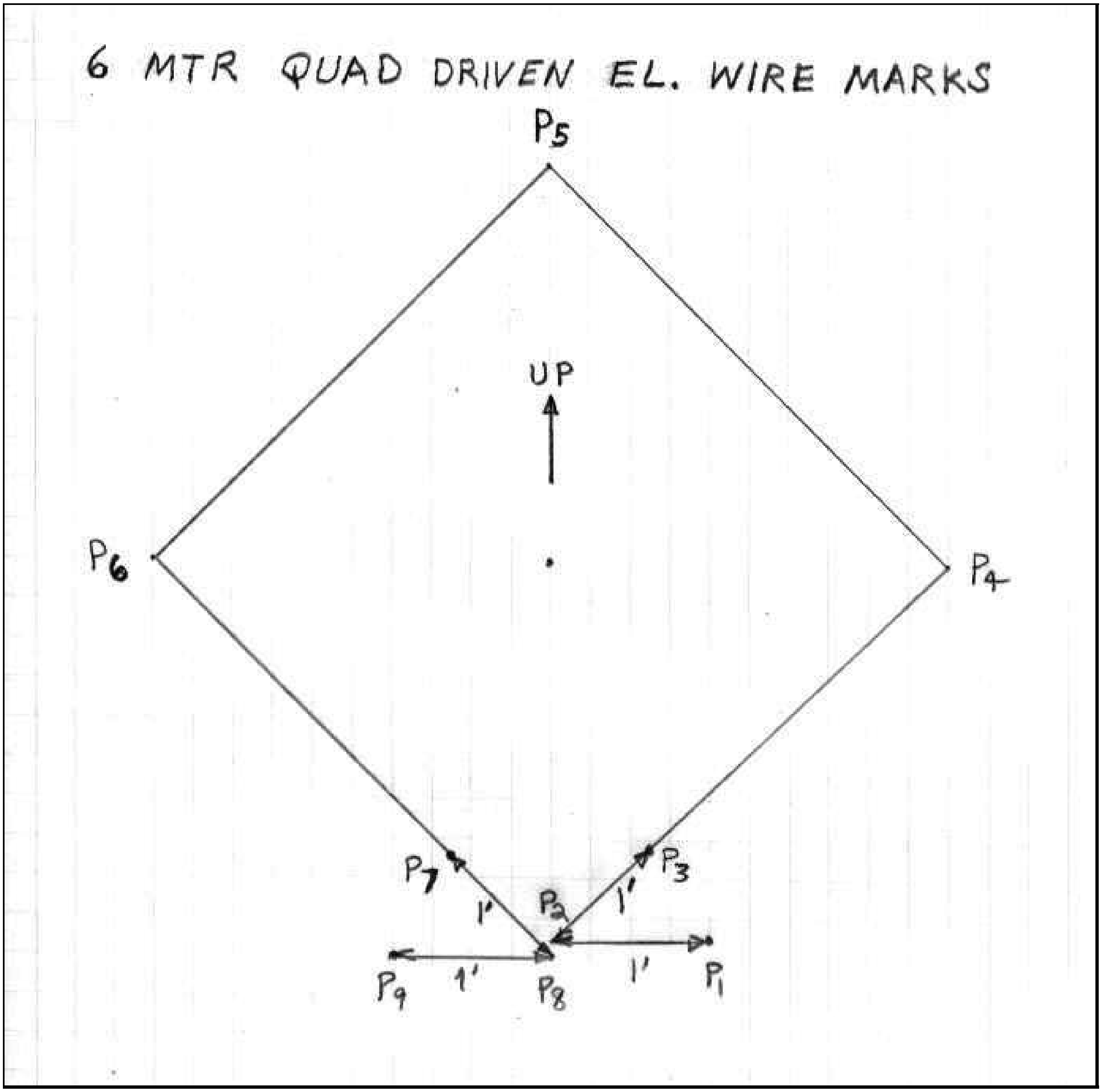
6 MTR DRIVEN ELEMENT (SPIDER 2) WIRE MARK POINTS FOLLOW

(p8-p2) in FT=19.9885   DRIVEN ELEMENT LENGTH in FT=19.9885

PT#	FT	INCH	1/16th INCH UNITS
1	1	0	0
2	2	0	0
3	3	0	0
4	6	11	15
5	11	11	15
6	16	11	14
7	20	11	14
8	21	11	14
9	22	11	14

>>


6 MTR QUAD NON DRIVEN EL. WIRE MARKS
P6
UP
P7
P5
P10
.5'
1'
P8
P9
1'
P2
.5'
P1
P3
1'
P4


6 MTR QUAD DRIVEN EL. WIRE MARKS
P5
UP
P6
P4
P7
P3
P2
1'
1'
P9
1'
P8
1'
P1

A MATLAB program named quadmodBL.m was developed to create a six band quad wire table for export to EZNEC. The run result from this program follows. See Section 3 for a listing of program quadmodBL.m.

```
>> quadmodBL

MONO OR MULTI BAND CUBICAL QUAD DESIGN CONSTANTS @ DIAMOND
ELEMENT SHAPES

FIRST BAND LISTED IS THE DRIVEN BAND. "DE" STANDS FOR DRIVEN
ELEMENT
DATA ELEMENT ORDER IS REF, DE, DIR1,  DIR2, ...DIRn

6 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS:   k=1005.4238   f=50.3   DE in FT=19.9885
ELEMENT LENGTHS AS A % FROM DE=3          0       -1.9       -1.7
ELEMENT BOOM LOCATIONS IN FT=16   20   25   30
SEGMENTS PER WIRE=11

20 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS:   k=997.6767   f=14.15   DE in FT=70.5072
ELEMENT LENGTHS AS A % FROM DE=2.976          0      -1.704      -1.725
ELEMENT BOOM LOCATIONS IN FT=0   10   20   30
SEGMENTS PER WIRE=5

17 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS:   k=994.848   f=18.11   DE in FT=54.9336
ELEMENT LENGTHS AS A % FROM DE=3          0      -1.75      -1.75
ELEMENT BOOM LOCATIONS IN FT=0   10   20   30
SEGMENTS PER WIRE=5

15 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS:   k=996.9452   f=21.2   DE in FT=47.0257
ELEMENT LENGTHS AS A % FROM DE=3.071          0      -1.848      -1.77
ELEMENT BOOM LOCATIONS IN FT=0   10   20   30
SEGMENTS PER WIRE=5

12 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS:   k=998.3181   f=24.93   DE in FT=40.0448
ELEMENT LENGTHS AS A % FROM DE=3          0      -1.75      -1.75
ELEMENT BOOM LOCATIONS IN FT=0   10   20   30
SEGMENTS PER WIRE=5
```

10 MTR QUAD DESIGN CONSTANTS
DE LENGTH CONSTANTS: k=1001.0343 f=28.45 DE in FT=35.1857
ELEMENT LENGTHS AS A % FROM DE=3.014 0 -2.066 -1.744 -1.723
ELEMENT BOOM LOCATIONS IN FT=0 5 10 20 30
SEGMENTS PER WIRE=5

MTR BAND	BAND WIRES	SEGS PER WIRE	TOTAL WIRES	TOTAL #WIRE SEGS	DRIVEN ELEMENT WIRE NRS 0% DEa#	100% DEb#
6	16	11	16	176	5	8
20	16	5	32	256	21	24
17	16	5	48	336	37	40
15	16	5	64	416	53	56
12	16	5	80	496	69	72
10	20	5	100	596	85	88

For the diamond quad loop configuration EZNEC must use a split source at the junction of wire numbers 5 (0% end) and 8 (100% end)

The above table also lists the driven element wire number(s) for the non driven bands in case impedance termination effects are to be modeled in EZNEC

EZNEC can work with up to 500 wire segments (SEGS) total
EZNEC-M Pro version can work with up to 10,000 wire segments total

EZNEC wire table output in Meter units with zero antenna height follows

4.876800 0.000000 -1.109324 4.876800 1.109324 0.000000 0.002053 11.000000
4.876800 1.109324 0.000000 4.876800 0.000000 1.109324 0.002053 11.000000
4.876800 0.000000 1.109324 4.876800 -1.109324 0.000000 0.002053 11.000000
4.876800 -1.109324 0.000000 4.876800 0.000000 -1.109324 0.002053 11.000000
6.096000 0.000000 -1.077014 6.096000 1.077014 0.000000 0.002053 11.000000
6.096000 1.077014 0.000000 6.096000 0.000000 1.077014 0.002053 11.000000
6.096000 0.000000 1.077014 6.096000 -1.077014 0.000000 0.002053 11.000000
6.096000 -1.077014 0.000000 6.096000 0.000000 -1.077014 0.002053 11.000000
7.620000 0.000000 -1.056550 7.620000 1.056550 0.000000 0.002053 11.000000
7.620000 1.056550 0.000000 7.620000 0.000000 1.056550 0.002053 11.000000
7.620000 0.000000 1.056550 7.620000 -1.056550 0.000000 0.002053 11.000000
7.620000 -1.056550 0.000000 7.620000 0.000000 -1.056550 0.002053 11.000000
9.144000 0.000000 -1.058704 9.144000 1.058704 0.000000 0.002053 11.000000
9.144000 1.058704 0.000000 9.144000 0.000000 1.058704 0.002053 11.000000
9.144000 0.000000 1.058704 9.144000 -1.058704 0.000000 0.002053 11.000000
9.144000 -1.058704 0.000000 9.144000 0.000000 -1.058704 0.002053 11.000000

0.000000 0.000000 -3.912095 0.000000 3.912095 0.000000 0.002053 5.000000
0.000000 3.912095 0.000000 0.000000 0.000000 3.912095 0.002053 5.000000
0.000000 0.000000 3.912095 0.000000 -3.912095 0.000000 0.002053 5.000000
0.000000 -3.912095 0.000000 0.000000 0.000000 -3.912095 0.002053 5.000000
3.048000 0.000000 -3.799036 3.048000 3.799036 0.000000 0.002053 5.000000
3.048000 3.799036 0.000000 3.048000 0.000000 3.799036 0.002053 5.000000
3.048000 0.000000 3.799036 3.048000 -3.799036 0.000000 0.002053 5.000000
3.048000 -3.799036 0.000000 3.048000 0.000000 -3.799036 0.002053 5.000000
6.096000 0.000000 -3.734300 6.096000 3.734300 0.000000 0.002053 5.000000
6.096000 3.734300 0.000000 6.096000 0.000000 3.734300 0.002053 5.000000
6.096000 0.000000 3.734300 6.096000 -3.734300 0.000000 0.002053 5.000000
6.096000 -3.734300 0.000000 6.096000 0.000000 -3.734300 0.002053 5.000000
9.144000 0.000000 -3.733502 9.144000 3.733502 0.000000 0.002053 5.000000
9.144000 3.733502 0.000000 9.144000 0.000000 3.733502 0.002053 5.000000
9.144000 0.000000 3.733502 9.144000 -3.733502 0.000000 0.002053 5.000000
9.144000 -3.733502 0.000000 9.144000 0.000000 -3.733502 0.002053 5.000000
0.000000 0.000000 -3.048706 0.000000 3.048706 0.000000 0.002053 5.000000
0.000000 3.048706 0.000000 0.000000 0.000000 3.048706 0.002053 5.000000
0.000000 0.000000 3.048706 0.000000 -3.048706 0.000000 0.002053 5.000000
0.000000 -3.048706 0.000000 0.000000 0.000000 -3.048706 0.002053 5.000000
3.048000 0.000000 -2.959908 3.048000 2.959908 0.000000 0.002053 5.000000
3.048000 2.959908 0.000000 3.048000 0.000000 2.959908 0.002053 5.000000
3.048000 0.000000 2.959908 3.048000 -2.959908 0.000000 0.002053 5.000000
3.048000 -2.959908 0.000000 3.048000 0.000000 -2.959908 0.002053 5.000000
6.096000 0.000000 -2.908110 6.096000 2.908110 0.000000 0.002053 5.000000
6.096000 2.908110 0.000000 6.096000 0.000000 2.908110 0.002053 5.000000
6.096000 0.000000 2.908110 6.096000 -2.908110 0.000000 0.002053 5.000000
6.096000 -2.908110 0.000000 6.096000 0.000000 -2.908110 0.002053 5.000000
9.144000 0.000000 -2.908110 9.144000 2.908110 0.000000 0.002053 5.000000
9.144000 2.908110 0.000000 9.144000 0.000000 2.908110 0.002053 5.000000
9.144000 0.000000 2.908110 9.144000 -2.908110 0.000000 0.002053 5.000000
9.144000 -2.908110 0.000000 9.144000 0.000000 -2.908110 0.002053 5.000000
0.000000 0.000000 -2.611631 0.000000 2.611631 0.000000 0.002053 5.000000
0.000000 2.611631 0.000000 0.000000 0.000000 2.611631 0.002053 5.000000
0.000000 0.000000 2.611631 0.000000 -2.611631 0.000000 0.002053 5.000000
0.000000 -2.611631 0.000000 0.000000 0.000000 -2.611631 0.002053 5.000000
3.048000 0.000000 -2.533818 3.048000 2.533818 0.000000 0.002053 5.000000
3.048000 2.533818 0.000000 3.048000 0.000000 2.533818 0.002053 5.000000
3.048000 0.000000 2.533818 3.048000 -2.533818 0.000000 0.002053 5.000000
3.048000 -2.533818 0.000000 3.048000 0.000000 -2.533818 0.002053 5.000000
6.096000 0.000000 -2.486993 6.096000 2.486993 0.000000 0.002053 5.000000
6.096000 2.486993 0.000000 6.096000 0.000000 2.486993 0.002053 5.000000
6.096000 0.000000 2.486993 6.096000 -2.486993 0.000000 0.002053 5.000000
6.096000 -2.486993 0.000000 6.096000 0.000000 -2.486993 0.002053 5.000000
9.144000 0.000000 -2.488969 9.144000 2.488969 0.000000 0.002053 5.000000
9.144000 2.488969 0.000000 9.144000 0.000000 2.488969 0.002053 5.000000

```
9.144000 0.000000 2.488969 9.144000 -2.488969 0.000000 0.002053 5.000000
9.144000 -2.488969 0.000000 9.144000 0.000000 -2.488969 0.002053 5.000000
0.000000 0.000000 -2.222408 0.000000 2.222408 0.000000 0.002053 5.000000
0.000000 2.222408 0.000000 0.000000 0.000000 2.222408 0.002053 5.000000
0.000000 0.000000 2.222408 0.000000 -2.222408 0.000000 0.002053 5.000000
0.000000 -2.222408 0.000000 0.000000 0.000000 -2.222408 0.002053 5.000000
3.048000 0.000000 -2.157678 3.048000 2.157678 0.000000 0.002053 5.000000
3.048000 2.157678 0.000000 3.048000 0.000000 2.157678 0.002053 5.000000
3.048000 0.000000 2.157678 3.048000 -2.157678 0.000000 0.002053 5.000000
3.048000 -2.157678 0.000000 3.048000 0.000000 -2.157678 0.002053 5.000000
6.096000 0.000000 -2.119919 6.096000 2.119919 0.000000 0.002053 5.000000
6.096000 2.119919 0.000000 6.096000 0.000000 2.119919 0.002053 5.000000
6.096000 0.000000 2.119919 6.096000 -2.119919 0.000000 0.002053 5.000000
6.096000 -2.119919 0.000000 6.096000 0.000000 -2.119919 0.002053 5.000000
9.144000 0.000000 -2.119919 9.144000 2.119919 0.000000 0.002053 5.000000
9.144000 2.119919 0.000000 9.144000 0.000000 2.119919 0.002053 5.000000
9.144000 0.000000 2.119919 9.144000 -2.119919 0.000000 0.002053 5.000000
9.144000 -2.119919 0.000000 9.144000 0.000000 -2.119919 0.002053 5.000000
0.000000 0.000000 -1.953003 0.000000 1.953003 0.000000 0.002053 5.000000
0.000000 1.953003 0.000000 0.000000 0.000000 1.953003 0.002053 5.000000
0.000000 0.000000 1.953003 0.000000 -1.953003 0.000000 0.002053 5.000000
0.000000 -1.953003 0.000000 0.000000 0.000000 -1.953003 0.002053 5.000000
1.524000 0.000000 -1.895862 1.524000 1.895862 0.000000 0.002053 5.000000
1.524000 1.895862 0.000000 1.524000 0.000000 1.895862 0.002053 5.000000
1.524000 0.000000 1.895862 1.524000 -1.895862 0.000000 0.002053 5.000000
1.524000 -1.895862 0.000000 1.524000 0.000000 -1.895862 0.002053 5.000000
3.048000 0.000000 -1.856693 3.048000 1.856693 0.000000 0.002053 5.000000
3.048000 1.856693 0.000000 3.048000 0.000000 1.856693 0.002053 5.000000
3.048000 0.000000 1.856693 3.048000 -1.856693 0.000000 0.002053 5.000000
3.048000 -1.856693 0.000000 3.048000 0.000000 -1.856693 0.002053 5.000000
6.096000 0.000000 -1.862798 6.096000 1.862798 0.000000 0.002053 5.000000
6.096000 1.862798 0.000000 6.096000 0.000000 1.862798 0.002053 5.000000
6.096000 0.000000 1.862798 6.096000 -1.862798 0.000000 0.002053 5.000000
6.096000 -1.862798 0.000000 6.096000 0.000000 -1.862798 0.002053 5.000000
9.144000 0.000000 -1.863196 9.144000 1.863196 0.000000 0.002053 5.000000
9.144000 1.863196 0.000000 9.144000 0.000000 1.863196 0.002053 5.000000
9.144000 0.000000 1.863196 9.144000 -1.863196 0.000000 0.002053 5.000000
9.144000 -1.863196 0.000000 9.144000 0.000000 -1.863196 0.002053 5.000000

>>
```

The above wire table is exported from MATLAB to EZNEC as an ASCII file named qallw. Directions for exporting are described at the end of the program listing for quadmodBL.m in Section 3.

3.) MATLAB PROGRAM LISTINGS

The listing of MALAB program sixMTR6B.m follows:

```
% M-file sixMTR6B.m
% Base line design
% 6 MTR six band quad EZNEC performance data taken 7-4-05
% m columns: 1=FREQ in MHZ (@EZNEC), 2=Gain in dBi, 3=FB in dB, 4=FBR in dB
m=[49.00 15.27 11.13 10.05
   49.05 15.35 11.40 10.31
   49.10 15.42 11.67 10.57
   49.15 15.50 11.96 10.84
   49.20 15.57 12.26 11.12
   49.25 15.64 12.58 11.41
   49.30 15.70 12.90 11.70
   49.35 15.77 13.25 12.00
   49.40 15.82 13.60 12.31
   49.45 15.88 13.99 12.62
   49.50 15.94 14.38 12.95
   49.55 15.99 14.82 13.29
   49.60 16.04 15.27 13.64
   49.65 16.09 15.74 13.98
   49.70 16.14 16.26 14.36
   49.75 16.19 16.81 14.73
   49.80 16.23 17.40 15.11
   49.85 16.27 18.06 15.50
   49.90 16.31 18.75 15.90
   49.95 16.35 19.55 16.31
   50.00 16.39 20.43 16.73
   50.05 16.42 21.41 17.16
   50.10 16.46 22.55 17.58
   50.15 16.49 23.87 17.99
   50.20 16.53 25.45 18.40
   50.25 16.56 27.48 18.80
   50.30 16.60 30.12 19.19
   50.35 16.63 34.10 19.52
   50.40 16.66 41.70 19.83
   50.45 16.69 48.90 20.05
   50.50 16.73 36.22 20.19
   50.55 16.76 31.10 20.26
   50.60 16.79 27.85 20.21
   50.65 16.82 25.41 20.05
   50.70 16.85 23.51 19.78
   50.75 16.89 21.88 19.39
   50.80 16.91 20.51 18.91
```

```
   50.85 16.94 19.27 18.36
   50.90 16.96 18.15 17.72
   50.95 16.99 17.13 17.02
   51.00 17.00 16.21 16.21
   51.05 17.01 15.31 15.31
   51.10 17.01 14.50 14.50
   51.15 17.00 13.71 13.71
   51.20 16.98 12.96 12.96
   51.25 16.93 12.24 12.24
   51.30 16.88 11.56 11.56
   51.35 16.79 10.91 10.81
   51.40 16.68 10.25 9.99
   51.45 16.54 9.65 9.21
   51.50 16.36 9.03 8.40
   51.55 16.15 8.43 7.62
   51.60 15.89 7.85 6.84
   51.65 15.59 7.28 6.06
   51.70 15.25 6.73 5.29];
%
delF=0.306; % EZNEC real world offset frequency in MHZ
f=m(:,1)-delF;
g=m(:,2); % dBi gain
dplg=7.8; % 6 MTR Reference dipole dBi gain at 55 ft
fb=m(:,3); % FB
fbr=m(:,4); % FBR
plot(f,g-dplg,'LineWidth',2)
grid
title('FIG 1 6 MTR SIX BAND QUAD GAIN IN dBd VERSUS FREQUENCY')
xlabel('FREQUENCY IN MHZ')
ylabel('GAIN OVER A DIPOLE IN dBd')
keyboard
%
plot(f,fb,'LineWidth',2)
grid
title('FIG 2A 6 MTR SIX BAND QUAD FRONT TO BACK (FB) RATIO VS
FREQUENCY')
xlabel('FREQUENCY IN MHZ')
ylabel('FRONT TO BACK RATIO IN dB')
keyboard
%
plot(f,fbr,'LineWidth',2)
grid
title('FIG 2B 6 MTR SIX BAND QUAD FRONT TO BACK REGION (FBR) RATIO
VS FREQ')
xlabel('FREQUENCY IN MHZ')
ylabel('FRONT TO BACK REGION RATIO IN dB')
```

```
%
fspl=(min(f):.001:max(f))';
%
gspl=spline(f,g,fspl);
[Y I]=sort(-gspl);
gmax=-Y(1); % max dBi
Fgmax=fspl(I(1));
disp(' ')
disp(['MAX GAIN FREQ IN MHZ=',num2str(Fgmax),'      MAX GAIN IN
dBi=',num2str(gmax),'      MAX GAIN IN dBd=',num2str(gmax-dplg),])
disp(' ')
%
fbspl=spline(f,fb,fspl);
[Y1 I1]=sort(-fbspl);
fbmax=-Y1(1); % max dB
Ffbmax=fspl(I1(1));
disp(['MAX FB FREQ IN MHZ=',num2str(Ffbmax),'    MAX FB IN
dB=',num2str(fbmax)])
disp(' ')
%
fbrspl=spline(f,fbr,fspl);
[Y2 I2]=sort(-fbrspl);
fbrmax=-Y2(1); % max dB
Ffbrmax=fspl(I2(1));
disp(['MAX FBR FREQ IN MHZ=',num2str(Ffbrmax),'    MAX FBR IN
dB=',num2str(fbrmax)])
disp(' ')
```

The listing of MATLAB program zcon.m follows:

```
% zcon.m
% MATLAB program for converting EZNEC SWR data file output
% Converts EZNEC z matrix data to condensed form for input to
%  MATLAB program quadk1.m
% Plots real and imaginary parts of Z versus frequency
%  For either case of one (square quad loops) or two
%   (diamond quad loops) sources used to drive antenna.
% Run EZNEC SWR tab, get z data from: File, View File, LastZ
% Cut and paste LastZ data to z=[paste data, then ]; in MATLAB work space.
global z zout % Copy this line to MATLAB work space before running this program
% Run this program (zcon) in MATLAB work space
% Cut and paste zout data from work space to quadk1 Q section SWR program
%  as a data matrix.
nz=length(z);
if z(1,1)==z(2,1)  % split EZNEC sources case with two rows per frequency
zout=zeros(nz/2,3);
for g=1:nz/2
   k=2*g-1;
   zout(g,1)=z(k,1);
   zout(g,2)=z(k,3);
   zout(g,3)=z(k,4);
end
else             % single EZNEC source case. One row per frequency
   zout=zeros(nz,3);
   zout(:,1)=z(:,1);
   zout(:,2)=z(:,3);
   zout(:,3)=z(:,4);
end
zout % Converted file output F, R, X columns
%
plot(zout(:,1),zout(:,2),'LineWidth',1)
hold on
plot(zout(:,1),zout(:,3),':','LineWidth',2)
grid
text(zout(3,1),zout(3,2),'REAL R (SOLID)')
text(zout(3,1),zout(3,3),'IMAGINARY X (DASHED)')
xlabel('FREQUENCY IN MHZ')
ylabel('IMPEDANCE IN OHMS')
title('FIG 1 IMPEDANCE VERSUS FREQUENCY')
%
prevz=zout(1,3);
j=0;
clear nref
for i=2:nz
```

```
    r=zout(i,3)/prevz;
    if r<0
        j=j+1;
        nref(j,1)=i; % Loop detects all resonant frequencies
    end
    prevz=zout(i,3);
end
for j=1:length(nref)
nrefj=nref(j,1);
Fr=polyval(polyfit(zout(nrefj-2:nrefj+1,3),zout(nrefj-2:nrefj+1,1),2),0);
Rr=polyval(polyfit(zout(nrefj-2:nrefj+1,1),zout(nrefj-2:nrefj+1,2),2),Fr);
disp(' ')
disp(['Resonant Frequency Fr in MHZ=',num2str(Fr)])
disp(' ')
disp(['Resonant Resistance in Ohms=',num2str(Rr)])
disp(' ')
disp(' ')
end
```

The listing of MATLAB program swrQ.m follows.

```
% M-file swrQ.m
% Computes SWR "swrq" versus frequency for a quarter wave Q match.
% Program as coded uses a 75 Ohm RG11 coax Q section followed by
%   any length of 52 Ohm coax.
% Program can be set for other Q section line types by changing Rq and
%   Cprime (and computed velocity factor "vf") to values for the line type.
% Program also computes SWR "swr52" for a 52 Ohm coax feed without the Q
%   section.
format short
clear i
clear j
global zout % from zcon.m run
f=zout(:,1);
z=zout(:,2)+i*zout(:,3);
%global f z % inputs
global swrq swr50  % outputs
% f=column matrix of Mhz frequencies
% z=column matrix of complex antenna driving point impedances (Ohms)
Rq=75; % Q match line Zo value in Ohms (For RG11 coax it is 75 Ohms)
Cprime=20.5; % Q match line capacitance per foot in pf/FT units
%             (For RG11 coax it is 20.5 pF/FT)
vf=1016/(Rq*Cprime); % Velocity factor of Q section line
%Rq=Rq/2; % @ Two pieces of RG11 75 ohm coax in parallel Q section
Fdmat=[14.174 18.118 21.224 24.94 28.4 50.1]; % Q section design frequencies
fm=mean(f); % mean of input frequencies (Mhz)
for b=1:6 % Auto detect Q section design frequency loop
   if abs(fm-Fdmat(b))<1.5
      Fd=Fdmat(b);
   end
end
lambdaQ=vf*983.5712/Fd; % One wavelength on Q section line in feet @ Fd Mhz
xQ=0.25*lambdaQ; % Length of quarter wave Q section in FT for @ Fd
lambda=vf*983.5712./f; % One wavelength on Q section line in feet @ f Mhz
theta=2*pi*xQ./lambda; % Q section line length in radians of phase shift @ f Mhz
gammax=j*theta;
% z1=complex impedance looking into Q section input port (Ohms)
z1=Rq*(z.*cosh(gammax)+Rq*sinh(gammax))./(Rq*cosh(gammax)+z.*sinh(gammax));
R50=50; % swrq @ 50 Ohm coax of any length after quarter wave Q section
rhoq=(z1-R50)./(z1+R50); % Reflection coefficient on 52 Ohm coax line with Q section
swrq=(1+abs(rhoq))./(1-abs(rhoq));
rho50=(z-R50)./(z+R50); % Reflection coefficient on 52 Ohm coax line without Q
section
swr50=(1+abs(rho50))./(1-abs(rho50)); % swr52 without Q section match
%
```

```
table=0; % Set table=1 for table printout
if table==1
disp(' ')
disp('Quarter Wave Q Section Made Of RG11 Coax')
disp(' ')
disp('   Zo Ohms   Design F  L in FT  L in Inch')
disp([Rq Fd xQ 12*xQ])
disp(' ')
end

%
% SUPPLY SWR(f) data >3 range data for below intercept points
fadj=-0.306; % EZNEC adjusted freq to real world
disp(' ')
disp(['ADJUSTMENT TO EZNEC TUNING IN MHZ=',num2str(fadj)])
F=f+fadj;
fspl=(min(F):.001:max(F))';    % 1.0 khz freq step fit frequencies
swr50spl=spline(F,swr50,fspl); % swr50 every 1.0 khz
nf=length(fspl); % number of freq steps
%
ref=3.0;
x=swr50spl;
oldsign=sign(x(2)-ref);
for i=2:nf
   newsign=sign(x(i)-ref);
   if newsign~=oldsign
      if x(i)-x(i-1)<0
         Flow3=fspl(i);
         disp(['LOWER SWR=',num2str(ref),'   POINT FREQUENCY=',num2str(fspl(i))])
      end
   end
   oldsign=newsign;
end
%
ref=2.5;
x=swr50spl;
oldsign=sign(x(2)-ref);
for i=2:nf
   newsign=sign(x(i)-ref);
   if newsign~=oldsign
      if x(i)-x(i-1)<0
         Flow2p5=fspl(i);
         disp(['LOWER SWR=',num2str(ref),'   POINT FREQUENCY=',num2str(fspl(i))])
      end
   end
oldsign=newsign;
```

```
end
%
x=swr50spl;
[Y I]=sort(x);
Fmin=fspl(I(1));
swrMIN=Y(1);
disp(['MINIMUM SWR FREQUENCY=',num2str(fspl(I(1))),'    MIN
SWR=',num2str(Y(1))])
%
ref=2.5;
x=swr50spl;
oldsign=sign(x(2)-ref);
for i=2:nf
   newsign=sign(x(i)-ref);
   if newsign~=oldsign
      if x(i)-x(i-1)>0
         Fhigh2p5=fspl(i);
         disp(['UPPER SWR=',num2str(ref),'   POINT FREQUENCY=',num2str(fspl(i))])
      end
   end
   oldsign=newsign;
end
%
ref=3.0;
x=swr50spl;
oldsign=sign(x(2)-ref);
for i=2:nf
   newsign=sign(x(i)-ref);
   if newsign~=oldsign
      if x(i)-x(i-1)>0
         Fhigh3=fspl(i);
         disp(['UPPER SWR=',num2str(ref),'   POINT FREQUENCY=',num2str(fspl(i))])
      end
   end
   oldsign=newsign;
end
disp(' ')
disp([Flow3 Flow2p5 Fmin Fhigh2p5 Fhigh3])
disp([3.0 2.5 swrMIN 2.5 3.0])
swrpers=100*[(Flow3/Fmin-1) (Flow2p5/Fmin-1) 0 (Fhigh2p5/Fmin-1) (Fhigh3/Fmin-
1)];
disp([swrpers])
disp(['SWR<2.5 PERCENT BANDWIDTH=',num2str(swrpers(1,4)-swrpers(1,2))])
disp(['SWR<3.0 PERCENT BANDWIDTH=',num2str(swrpers(1,5)-swrpers(1,1))])
%
plot(fspl,x)
```

```
grid
xlabel('FREQUENCY IN MHZ')
ylabel('SWR')
title('FIG 1 SWR VERSUS FREQUENCY PLOT')
%
disp(' ')
disp('   FREQ      REAL      IMAG       SWR50')
disp([(f+fadj) real(z) imag(z) swr50])
disp(' ')
```

The listing of the MATLAB program quad6mtr.m follows.

```
% M-file quad6mtr.m
% 6 MTR 4 element quad design program
clear all
k=1005.4238; % ft/MHZ design constant
f=50.30; % MHZ design frequency for minimum SWR
de=k/f; % de=driven element length in ft
per=[3 0 -1.9 -1.7]'; % percent change for ref, de, dir1, dir2 elements relative to de
boom=[16 20 25 30]'; % boom locations of 6 MTR quad ref, de, dir1 dir2 in ft
%                       note: 20 MTR REF is at zero position on boom
% note: de is on 20MTR DIR1 spider, dir2 is on 20 MTR DIR2 spider
%       ref and dir1 are on separate 6 MTR band only spiders
lengths=de*(1+per/100); % lengths of ref, de, dir1, dir2 in ft
sides=lengths/4; % sides of ref, de, dir1, dir2 in ft
arms=sides/sqrt(2);
w=[0.7 1.3 0.7 1.3]'/12; % flat corner widths in ft for ref, de, dir1 dir2
%w=[0.7 0.7 0.7 0.7]'/12; % @ experimental quad all arms 0.5 inch dia
slack=(2/16)/12; % allowed slack for arm wire holes in ft
holes=sides/sqrt(2)-w*(sqrt(2)-1)/2-slack; % quad arm wire holes measured from center
of boom in ft
bb=(3+14/16)/12; % opposing quad arm pair butt to butt separation in ft
disp(' ')
disp(['6 MTR QUAD OPPOSING ARM PAIR BUTT TO BUTT SEPARATION IN
INCHES=',num2str(12*bb)])
holesbutt=holes-bb/2; % wire holes from arm butt end in ft
names=[' REF'
       '  DE'
       'DIR1'
       'DIR2'];
global ft INTft INTin INT16ths
ft=holesbutt;
ftin16ths % program that converts decimal ft to integer ft, inches, and 1/16 inch units
disp(' ')
disp('6 MTR QUAD ARM WIRE HOLES FROM BUTT ENDS')
disp(' ')
disp('EL         FT        INCH      1/16 INCH')
disp(' ')
for i=1:4
fprintf(1,'%s\t %5.0f %10.0f %10.0f\n',names(i,:),INTft(i),INTin(i),INT16ths(i));
end
disp(' ')
%
ft=lengths;
ftin16ths % program that converts decimal ft to integer ft, inches, and 1/16 inch units
disp('6 MTR QUAD ELEMENT WIRE LENGTHS')
```

```
disp(' ')
disp('  EL     DEC FT          FT         INCH      1/16 INCH')
disp(' ')
for i=1:4
fprintf(1,'%s\t %5.5f %10.0f %10.0f
%10.f\n',names(i,:),lengths(i),INTft(i),INTin(i),INT16ths(i));
end
disp(' ')
%
%
disp(' ')
%disp('Wire Marks for Experimental 6 MTR  4 EL quad')
disp(' ')
disp('NON DRIVEN ELEMENT WIRE MARK POINTS FOLLOW')
for i=1:4 % quad spider loop 6 MTR REF DE DIR1 DIR2 order
   if i==2 % 6 mtr DE case
      continue
   end
      SIDE=sides(i); % side in ft
      p1=1;
      p2=p1+1.0; % Note change
      p3=p2+0.5;
      p4=p3+1;
      p5=p4+SIDE;
      p6=p5+SIDE;
      p7=p6+SIDE;
      p8=p7+SIDE-2; % Note change
      p9=p8+0.5; % Note change
      p10=p9+1.0; % Note change
      pts1=[p1 p2 p3 p4 p5 p6 p7 p8 p9 p10]';
      ft=pts1;
      ftin16ths
      disp(' ')
      disp(' ')
      disp(['QUAD SPIDER NUMBER=',num2str(i),'    6 MTR ELEMENT=',names(i,:)])
      disp(' ')
      disp(['(p9-p2) in FT=',num2str(p9-p2),'       ELEMENT LENGTH in
FT=',num2str(lengths(i))])
      disp(' ')
      disp('    PT#   FT    INCH   1/16th INCH UNITS')
      disp([(1:10)' INTft INTin INT16ths])
end
disp(' ')
disp(' ')
disp(' ')
disp('6 MTR DRIVEN ELEMENT (SPIDER 2) WIRE MARK POINTS FOLLOW')
```

```
delta=0/12; % lower arm holes space +/-7/16 inch about holes for other arms (FT)
coax=0; % Lower diamond sides shortening for coax feed in feet
%     Above @ p2 p4 p5 p6 and p8 wire mark points centered in driven element quad
arms
%clear ft
SIDE=sides(2); % 6 MTR DE side in ft
p1=1;
p2=p1+1.0; % Note change
p3=p2+1.0; % Note change
p4=p3+SIDE-1.0; % Note change
p5=p4+SIDE;
p6=p5+SIDE;
p7=p6+SIDE-1.0; % Note change
p8=p7+1.0; % Note change
p9=p8+1.0; % Note change
pts2=[p1 p2 p3 p4 p5 p6 p7 p8 p9]';
ft=pts2;
ftin16ths
disp(' ')
disp(['(p8-p2) in FT=',num2str(p8-p2),'       DRIVEN ELEMENT LENGTH in
FT=',num2str(lengths(2))])
disp(' ')
disp('    PT#   FT    INCH   1/16th INCH UNITS')
disp([(1:9)' INTft INTin INT16ths])
```

The listing of a MATLAB subprogram named ftin16ths.m used in quad6mtr.m to convert decimal feet to integer feet, integer inches, and rounded integer 1/16th inch increments follows.

```
% M-file ftin16ths.m
% Program input is matrix "ft" in feet units
% Program outputs are: matrix INTft in integer feet
%                      matrix INTin in integer inches
%                      matrix INT16ths in integer 1/16th inch increments
global ft INTft INTin INT16ths
INTft=floor(ft); % Integer feet
FRACin=12*(ft-INTft); % Fractional inches
INTin=floor(FRACin); % Integer inches
INT16ths=round((FRACin-INTin)*16); % Integer 1/16 th inch increments
[r c]=size(ft);
for ii=1:r
   for jj=1:c
   if INT16ths(ii,jj)==16
      INT16ths(ii,jj)=0;
      INTin(ii,jj)=INTin(ii,jj)+1;
   end
end
end
```

A listing of MATLAB program quadmodBL.m follows. Comment statements show how it is structured and how to use it.

```
% M-file quadmodBL.m
% Base Line quad design 2005
% MATLAB program designed to create an exportable wire table for the EZNEC or
EZNEC-PRO
%   antenna modeling programs for any mono band or multi band
%   multi element Cubical Quad antenna in either the diamond or square loop
%   shape configuration.
%
% A note for radio amateurs not familiar with the MATLAB programming
% language follows. MATLAB is a powerful high level scientific programming
% language commonly used by college students and professional engineers.
% The student version of MATLAB can be downloaded from the Mathworks web
% site for $100. The professional version of MATLAB currently costs $1900.
% Both PC and MAC versions are available.
%
% Written by Bob Hume KG6B on 5/15/2003  (310) 376-4192 (H) 814-7557 (W)
%
% Final EZNEC export file wire end locations and sizes are in meter units
%       with zero antenna height
% Export wire file includes the number of EZNEC segments used to model
%       each wire.
% See detailed instructions on how export the quad wire table file generated
%       by this program to EZNEC at the end of this program listing.
%
%square=1; % Activate this line (remove leading %) for a square quad loop
configuration.
%           EZNEC should use a source at the middle of wire #5 for the
%           driven band.
square=0; % Activate this line for a diamond quad loop configuration.
%           EZNEC should use a split source at the 0% end of wire #5
%           and the 100% end of wire #8 for the driven band.
%
```

```
% Select all bands common bare copper wire diameter in feet "dia"
%       on following line(s).
%       Note that EZNEC can not properly model wire with a thick layer of
%       insulation. Enamel covered magnet wire can be properly modeled
%       since the insulation layer is very thin.
%dia=.06408/12; % #14 wire diameter in feet
dia=.08081/12; % #12 wire diameter in feet
%dia=.09074/12; % #11 wire diameter in feet
%
% Select Meter bands in quad on next line(s) that define matrix "bandset"
%bandset=[20 17 15 12 10]'; % MTR bands in quad. Choose one or all of the 20, 17,
%        15, 12, 10, or 6 MTR bands in any order except that the first band listed is
%        the driven band for which the antenna is evaluated. Consider the 500 wire
%        segment limit of EZNEC ($100 cost) when choosing the number of bands and
%        elements in the quads. The driven band uses 7 segments per wire. The
%        non driven bands use 5 segments per wire. There are four wires per
%        quad loop. EZNEC may give a warning using 5 segments per wire but
%        this is OK since the currents in the non driven band element wires are
%        small. (Or spend $400 for EZNEC-PRO version with 10,000 wire
%        segment modeling limit).
%bandset=[17 20 15 12 10]'; % Activate the desired banset line by removing
%                             the leading %
%bandset=[15 17 12 20 10]';
%bandset=[12 10 15 17 20]';
%bandset=[10 12 15 17 20]';
%bandset=[6 10 12 15 17 20]';
bandset=[6 20 17 15 12 10]';
%bandset=[20 17]';
%bandset=[6]';
%bandset=[20]';
%
%
NRbands=length(bandset);
wnr=zeros(NRbands,7);
wnr(:,1)=bandset;
nt=0;
segtotal=0;
%
disp(' ')
if square==1
   disp('MONO OR MULTI BAND CUBICAL QUAD DESIGN CONSTANTS @
SQUARE ELEMENT SHAPES')
else
   disp('MONO OR MULTI BAND CUBICAL QUAD DESIGN CONSTANTS @
DIAMOND ELEMENT SHAPES')
end
```

```
disp(' ')
disp('FIRST BAND LISTED IS THE DRIVEN BAND. "DE" STANDS FOR DRIVEN
ELEMENT')
disp('DATA ELEMENT ORDER IS REF, DE, DIR1, DIR2, ...DIRn')
for bandNR=1:NRbands % Band case loop
MTRband=bandset(bandNR); % Selected MTR band in loop
%
% MODEL THE QUAD DESIGN CONSTANTS FOR EACH BAND ON THE
FOLLOWING LINES.
%   THE PROGRAM QUAD MODEL ASSUMES THAT ONE REFLECTOR PER
BAND IS USED.
%   ONLY QUAD METER BANDS USED IN THE MATRIX "bandset" NEED BE
MODELED
if MTRband==20
% 20 MTR Quad design constants follow
k=997.6767; % Driven Element (DE) Length*Frequency Design Product in FT*MHZ
units
f=14.15;% experimental monobander
%f=14.055;    % DE Design Frequency in Mhz @ 2 el CW end
if bandNR==1
   segs=11; % segs=EZNEC segments per wire. segs must be odd for square quad loops
else
   segs=5;
end
elper=[2.976 0 -1.704 -1.725]'; % Percent change from driven element (DE)
%                              each element.
%                              Order: REF, DE, DIR1, DIR2, ...DIRn etc
elspace=[0 10 20 30]'; % Element locations along boom in ft (@ Reflector=0)
%                       Order: REF, DE, DIR1, DIR2, ...DIRn etc
disp(' ')
disp('20 MTR QUAD DESIGN CONSTANTS')
end
%
%
if MTRband==17
% 17 MTR Quad design constants follow
k=994.848; % DE Length*Frequency Design Product in FT*MHZ units
f=18.11;  % DE Design Frequency in Mhz
if bandNR==1
   segs=11; % segs=EZNEC segments per wire
else
   segs=5;
end
elper=[3 0 -1.75 -1.75]'; % Percent change from driven element (DE) size for
%                              each element.
%                              Order: REF, DE, DIR1, DIR2, ...DIRn etc
```

```
elspace=[0 10 20 30]'; % Element locations along boom in ft (@ Reflector=0)
%                                Order: REF, DE, DIR1, DIR2
disp(' ')
disp('17 MTR QUAD DESIGN CONSTANTS')
end
%
%
if MTRband==15
% 15 MTR Quad design constants follow
k=996.9452; % DE Length*Frequency Design Product in FT*MHZ units
f=21.2;  %  DE Design Frequency in Mhz
if bandNR==1
   segs=11; % segs=EZNEC segments per wire
else
   segs=5;
end
elper=[3.071 0 -1.848 -1.770]'; % Percent change from driven element (DE) size for
%                           each element.
%                           Order: REF, DE, DIR1, DIR2, ...DIRn etc
elspace=[0 10 20 30]'; % Element locations along boom in ft (@ Reflector=0)
%                                Order: REF, DE, DIR1, DIR2
disp(' ')
disp('15 MTR QUAD DESIGN CONSTANTS')
end
%
%
if MTRband==12
% 12 MTR Quad design constants follow
k=998.3181; % DE Length*Frequency Design Product in FT*MHZ units
f=24.93;  % DE Design Frequency in Mhz
if bandNR==1
   segs=11; % segs=EZNEC segments per wire
else
   segs=5;
end
elper=[3 0 -1.75 -1.75]'; % Percent change from driven element (DE) size for
%                           each element.
%                           Order: REF, DE, DIR1, DIR2, ...DIRn etc
elspace=[0 10 20 30]'; % Element locations along boom in ft (@ Reflector=0)
%                                Order: REF, DE, DIR1, DIR2
disp(' ')
disp('12 MTR QUAD DESIGN CONSTANTS')
end
%
%
if MTRband==10
```

```
% 10MTR Quad design constants follow
k=1001.0343; % DE Length*Frequency Design Product in FT*MHZ units
f=28.45;  % DE Design Frequency in Mhz
if bandNR==1
   segs=11; % segs=EZNEC segments per wire
else
   segs=5;
end
elper=[3.014 0 -2.066 -1.744 -1.723]'; % Percent change from driven element (DE) size
for
%                               each element.
%                               Order: REF, DE, DIR1, DIR2, ...DIRn etc
elspace=[0 5 10 20 30]'; % Element locations along boom in ft (@ Reflector=0)
%                                    Order: REF, DE, DIR1, DIR2, DIR3
disp(' ')
disp('10 MTR QUAD DESIGN CONSTANTS')
end
%
%
if MTRband==6
% 6 MTR Quad design constants follow
k=1005.4238; % DE Length*(Fmin swr) Design Product in FT*MHZ units @ 6 bander
design at 55 ft
%k=1006.3793; % DE Length*(Fmin swr) Design Product in FT*MHZ units @ mono
%                    bander at 55 ft
% Above based based on actual measurements on a 4 el mono band 6 mtr quad
f=50.300;  % Minimum SWR design frequency in Mhz (@ real world)
%           Note:EZNEC will give tuning results 0.306 Mhz higher than real world
if bandNR==1
   segs=11; % segs=EZNEC segments per wire
else
   segs=5;
end
elper=[3.0 0 -1.9 -1.7]'; % Percent change from driven element (DE) size for
%                                    Order: REF, DE, DIR1, DIR2, DIR3
elspace=[16 20 25 30]'; % Element locations along boom in ft (@ 20 MTR Reflector=0
point)
%                                    Order: REF, DE, DIR1, DIR2, DIR3
disp(' ')
disp('6 MTR QUAD DESIGN CONSTANTS')
end
%
if MTRband==30
% 30 MTR Quad design constants follow
k=997.6767; % Driven Element (DE) Length*Frequency Design Product in FT*MHZ
units
```

```
f=10.125;    % DE Design Frequency in Mhz
if bandNR==1
   segs=11; % segs=EZNEC segments per wire. segs must be odd for square quad loops
else
   segs=5;
end
elper=[2.976 0]'; % Percent change from driven element (DE) size for
%                              each element.
%                              Order: REF, DE, DIR1, DIR2, ...DIRn etc
elspace=[0 12.142]'; % Element locations along boom in ft (@ Reflector=0)
%
%Order: REF, DE, DIR1, DIR2, ...DIRn etc
disp(' ')
disp('30 MTR QUAD DESIGN CONSTANTS')
end
%
disp(['DE LENGTH CONSTANTS:   k=',num2str(k),'   f=',num2str(f),'   DE in
FT=',num2str(k/f)])
disp(['ELEMENT LENGTHS AS A % FROM DE=',num2str(elper')])
disp(['ELEMENT BOOM LOCATIONS IN FT=',num2str(elspace')])
disp(['SEGMENTS PER WIRE=',num2str(segs)])
%
elcirc=(k/f)*(1+elper/100); % Element total length (i.e. of all four sides) matrix in ft
elarm=elcirc/(4*sqrt(2)); % Diamond Quad arm length matrix in ft
%
n=length(elper); % Number of elements in Quad
A=zeros(4*n,8); % Blank EZNEC wire table. Column 8 for number of segments per wire
%
if square==0 % Diamond quad loop configuration
for i=1:n % Quad element number index i
   s=elspace(i,1);
   a=elarm(i,1);
   m=[s 0 -a s a 0 dia segs; % Wire coordinates matrix for diamond Quad element i
      s a 0 s 0 a dia segs;
      s 0 a s -a 0 dia segs;
      s -a 0 s 0 -a dia segs];
   A(4*(i-1)+1:4*(i-1)+4,:)=m;  % Wire coordinate accumulation for all n Quad elements
end
end
%
if square==1 % Square quad loop configuration
for i=1:n % Quad element number index i
   s=elspace(i,1);
   c=elarm(i,1)/sqrt(2); % Half side dimension of loop
   m=[s -c -c s c -c dia segs; % Wire coordinates matrix for square Quad element i
      s c -c s c c dia segs;
```

```
        s c c s -c c dia segs;
        s -c c s -c -c dia segs];
    A(4*(i-1)+1:4*(i-1)+4,:)=m;  % Wire coordinate accumulation for all n Quad elements
end
end
%
A(:,1:7)=(12*2.54/100)*A(:,1:7); % Convert wire dimensions from Feet to Meters
%
nt=nt+length(A);
segtotal=segtotal+segs*length(A);
wnr(bandNR,2)=length(A);
wnr(bandNR,3)=segs;
wnr(bandNR,4)=nt;
wnr(bandNR,5)=segtotal;
wnr(bandNR,6)=nt-length(A)+5;
wnr(bandNR,7)=nt-length(A)+8;
%
if bandNR==1
    B=A;
else
    Bold=B;
    nB=length(Bold);
    nA=length(A);
    B=zeros((nB+nA),8);
    B(1:nB,:)=Bold;
    B((nB+1):(nB+nA),:)=A;
end
end % End of bands loop
%
qall=B; % EZNEC wire table matrix for use in other MATLAB programs.
% The next three lines of MATLAB code create an ASCII text file for
%  wire table file "qall" which is compatible with the EZNEC wire
%  table import file requirements.
fid = fopen('qallw','wt'); % Open and write to ASCII text file qallw
fprintf(fid,'%f %f %f %f %f %f %f %f\n',B'); % ASCII text file of B
fclose(fid); % close file
%
if square==1
disp(' ')
disp('              SEGS       TOTAL  DRIVEN ELEMENT WIRE NUMBER')
disp('   MTR   BAND   PER  TOTAL #WIRE  MIDDLE OR 50% POINT IN WIRE')
disp('   BAND  WIRES  WIRE WIRES SEGS   DE#')
disp([wnr(:,1:6)])
disp(' ')
    disp('For the square quad loop configuration EZNEC must use a single source')
    disp('  at the center (50%) of wire number 5')
```

```
else
disp(' ')
disp('              SEGS       TOTAL  DRIVEN ELEMENT WIRE NUMBERS')
disp('   MTR   BAND   PER  TOTAL #WIRE   0%   100%')
disp('   BAND  WIRES  WIRE WIRES SEGS   DEa#  DEb#')
disp([wnr])
disp(' ')
   disp('For the diamond quad loop configuration EZNEC must use a split source')
   disp('  at the junction of wire numbers 5 (0% end) and 8 (100% end)')
end
disp(' ')
disp('The above table also lists the driven element wire number(s) for the non driven')
disp('  bands in case impedance termination effects are to be modeled in EZNEC')
disp(' ')
disp('EZNEC can work with up to 500 wire segments (SEGS) total')
disp('EZNEC-M Pro version can work with up to 10,000 wire segments total')
disp(' ')
disp(' ')
disp('EZNEC wire table output in Meter units with zero antenna height follows')
type qallw % EZNEC Wire table file in export compatible ASCII text file form
%
% To export the ASCII wire table file "qall" to EZNEC follow these steps.
% 1.) Run program quadmod.m in the MATLAB work space to create file "qall"
% 2.) Open EZNEC
% 3.) Click on the "WIRES" tab
% 4.) Click on the "Other" button
% 5.) Select "Import Wires From ASCII File"
% 6.) Select "Replace Existing Wires"
% 7.) Locate file "qall" on the path C:\MARLAB6p5\work\qall
% 8.) Double click file "qall"
% 9.) Repeat step 4
% 10.) Click on "Change units"
% 11.) Select feet and click OK
% 12.) Repeat step 4
% 13.) Select "Change Height by ..."
% 14.) Enter antenna height in feet and click OK
% 15.) In EZNEC window click the "Ground Type" tab
% 16.) Select real or perfect ground option and click OK
% 17.) In EZNEC window click the "Sources tab"
% 18.) Enter the source as follows for the square or diamond loop
%      For square quad loops EZNEC should use a source at the middle of wire #5
%      For diamond quad loops EZNEC should use a split source at the
%           0% end of wire #5 and the 100% end of wire #8
%      The source only needs to be set up one time for all "bandset" case
%      runs
% The above steps 1 to 17 can be performed in about a minute for each
```

```
%   "bandset" case. The program thereby makes it possible to evaluate large
%   multiband multielement quad arrays very quickly using EZNEC. Manual
%   wire table entry errors and tedium are avoided using this program.
%
% Also see MATLAB programs zcon.m and quadk1.m which use the EZNEC
%  antenna impedance versus frequency data table output "LastZ.txt"
%  obtained from an EZNEC SWR plot run
%  to plot SWR versus frequency using a 75 Ohm RG11AU quarter wave Q
%  section match to a RG213U 50 Ohm coaxial feed line.
```