ISSN 2310-6697
otoiser—open transactions on independent scientific-engineering research

FUNKTECHNIKPLUS # JOURNAL

Théorie—Expérimentation—Métrologie—Logiciel—Applications

ISSUE 8 — WEDNESDAY 30 SEPTEMBER 2015 — YEAR 3

This small European Journal is
In the Defense of Honesty in Science and Ethics in Engineering

Publisher — otoiser——open transactions on independent scientific engineering research, www.otoiser.org — info@otoiser.org, Hauptstraße 52, 2831 Scheiblingkirchen, Austria

Language — We declare the origins of the Journal by using in the cover page English, German and French, as well as, a Hellenic vignette. However, since we recognize the dominance of US English in the technical literature, we adopted it as the Journal's language, although it is not our native language.

Focus — We consider Radio—FUNK, which still creates a vivid impression of the untouchable, and its Technology—TECHNIK, from an Advanced—PLUS point of view, Plus—PLUS Telecommunications Engineering, Electrical Engineering and Computer Science, that is, we dynamically focus at any related scientific-engineering research regarding Théorie, Expérimentation, Métrologie, Logiciel, ou Applications.

Scope — We emphasize this scope broadness by extending the title of the Journal with a Doppelkreuz-Zeichen # which we use as a placeholder for substitution of our Editorial Team disciplines: # Telecommunications etc. as above, or # High Voltage, # Software Engineering, # Simulation etc. as below.

Frequency — We publish 3 issues per year: on 31st of January, on 31st of May, and on 30th of September, as well as, an extra issue every 3 papers and a volume every 2 years.

Editions — We increase the edition number of an issue only when is needed to reform one or more of its papers——thus to increase their version numbers——but we keep unchanged its 1st edition date shown on its front page and we number its pages sequentially from 1. We count the editions of *About* separately.

Format — We use a fixed-space font, hyphenation, justification, unfixed word spacing, and the uncommon for Journals **A5** (half A4) page size to achieve WYSIWYG printing and clear reading of 2 to 4 side-by-side pages on wide-screen displays

Printing-on-Demand — We can email gratis PDFs at 300-4000 dpi in booklet page scaling of either brochure or book type.

Please download the latest *About* edition from
about.ftpj.otoiser.org

Editorial Team

Technical Support

Information for Authors

This is a small, but independent, low profile Journal, in which we are all——Authors, Reviewers, Readers, and Editors—— free at last to be Peers in Knowledge, without suffering from Journal roles or positions, Professional—Amateur—Academic statuses, or established "impact factorizations", under the following guiding principles:

Authors — We know what Work means, we respect the Work of the Independent Researcher in Science and Engineering and we want to exhibit his Work. Thus, we decided to found this Free and Open Access Journal in which to publish this Work. Furthermore, as we care indeed for the Work of the technical author——especially a young or a beginner one——we strongly support the publication of his Work, as follows:

0 We do not demand from the author to transfer his own copyright to us. Instead, we only consider papers resulting from original research work only, and only if the author can assure us that he owns the copyright of his own paper as well as that he submits to the Journal either an original copy or a revised version of his own paper, for possible publication after review——or even for immediate republication, if this paper has already been published after review——but, in any case under a Creative Commons Attribution, CC-BY 3 Unported or CC-BY 4 International, License, only.
1 We encourage the author to submit his own paper written just in Basic English plus Technical Terminology.
2 We encourage the author even to select a pen name, which may drop it at any time to reveal his identity.
3 We encourage the author to submit an accepted for publication paper, which he was forced to decline that publication because it would be based on a review with unacceptable evaluation or derogatory comments.
4 We encourage the author to submit any paper that was rejected after a poor, impotent, inadequate, unreasonable, irresponsible, incompetent, or "just ticking" review.
5 We encourage the author to submit an unreviewed paper of his own that he uploaded on some Open Access repository.
6 We encourage the author to upload his published papers in our Journal to a truly Free Open repository like viXra.org.
7 We provide the author with a decent, express, peer review process, of up to just 4 weeks, by at least 2, either anonymous or onymous, reviewers.

8 We provide the author with the option to choose from 2 review processes: the traditional, anonymous, close one, as well as, a contemporary, onymous, open review in our private mailing list for Peer Discussion.
9 Under the Clause 0 : We immediately accept for publication a research paper directly resulting from a Project Report, or a Diploma-, Master-, or PhD-thesis, which already the author has successfully defended before a committee of experts, as long as he can mention 2 members of this committee who approved his Work.
10 Under the Clause 0 : We immediately accept for publication any paper which is not Openly Accessible on the Internet.
11 We immediately publish online a paper, as soon as it is accepted for publication in the Journal.
12 We quickly publish an extra issue——that is in excess of the 3 issues we publish a year——as soon as the review process of 3 papers is completed.

Reviewers — Every peer may voluntarily become a reviewer of the Journal in his skillfulness for as long as he wishes. In addition, each author of the Journal must review one paper in his expertness for each one of his published papers.

Readers — Every reader is a potential post-reviewer: we welcome comments and post-reviews in our private mailing list for Peer Discussion.

Editors — Every editor owns a PhD degree——to objectively prove that he really has the working experience of passing through the dominant publishing system. An editor pre-reviews a paper in order to check its compliance to our guiding principles and to select the appropriate reviewers of it. We can accept for consideration papers only in the expertise areas currently shown in the Editorial Team page, above. However, since we are very willing to amplify and extend the Scope of the Journal, we welcome the volunteer expert, in any related subject, who wants to join the Editorial Team as long as he unreservedly accepts our guiding principles.

Electronic Publishing

We regularly use the Free Libre Open Source Software Libre Office with the Free Liberation Mono font and the Freewares PDFCreator and PDF-Xchange Viewer. We also use, with some basic html code of ours: the Free Open Source Software Open Journal System OJS by the Public Knowledge Project PKP installed in our website, and the Free Open Digital Library of Internet Archive website, where we upload Issues, Paper reprints, *About* documents, and Volumes, in both portrait and landscape orientations, for exceptionally clear online reading with the Free Open Source BookReader.

Submissions

We can consider only papers submitted in a format which is fully compatible with LibreOffice——preferably in odt format.

Legal Notice — It is taken for granted that the submitter—correspondent author accepts, without any reservation, the totality of our publication conditions as they are analytically detailed here, in this *About*, as well as, that he also carries, in the case of a paper by multiple authors, the independent will of each one of his coauthors to unreservedly accept all the aforementioned conditions for their paper.

Internet Addresses

Internet Publishing Website : ftpj.otoiser.org
Internet Archive Digital Library : archive.org/details/@ftpj
Printing-on-Demand : pod@ftpj.otoiser.org
Principal Contact : principal-contact@ftpj.otoiser.org
Technical Support : technical-support@ftpj.otoiser.org
Editorial Team—Technical Support List : etts.ftpj.otoiser.org
Peer Discussion List : peers.ftpj.otoiser.org
Submissions : sub@ftpj.otoiser.org
Sample Paper Template : template.ftpj.otoiser.org

ARG NfP AoI
Antennas Research Group
Not-for-Profit Association of Individuals *
arg.op4.eu — arg@op4.eu
Hauptstraße 52, 2831 Scheiblingkirchen, Austria

* The Constitution of Greece, Article: 12(3) 2008: www.hellenicparliament.gr/en/Vouli-ton-Ellinon/To-Politevma
* The Hellenic Supreme Court of Civil and Penal Law: www.areiospagos.gr/en/ — Court Rulings:Civil|A1|511|2008

The Mensurasoft-System: Software Drivers for Scientific Devices in the Most Common Computer Languages on All Major Operating Systems with a Prototype API for Measurement Data Processing Applications - Practical Examples

P. Dieumegard *

Independent Researcher, Orléans, France

Abstract

For years now, although many scientific devices have a port for computer connection, the diversity of various connection types results in lack of software able to use the totality of them. This paper presents how to: (1) build software drivers with a prototype Application Programming Interface API for scientific devices using dynamic libraries under various combinations of connection ports, programming languages and operating systems, (2) employ these drivers in software programs capable of managing more than one device at the same time, and (3) use two specific software implementations of this whole Mensura (Measurement) System to select appropriate built driver, define communication settings, and collect-store-process-present experimental data.

Keywords

Mensurasoft, measurement, devices, dynamic library, programming languages, MS-Windows, Linux, driver

Introduction

In this paper, "scientific devices" are mainly measurement devices: pHmeters, voltmeters, thermometers etc, but sometimes they are actuators: motors, heaters, voltage-generators etc. Many scientific devices can be connected to a computer; in the past, usually through an ISA, PCI or RS-232, but nowadays via USB too.

Sometimes there is software to use them, often there is no software. When a software is sold or given with a device, it can be used only with this device or similar devices from the same supplier. This causes problems to professional scientists, as well to teachers of sciences, and to simple amateurs of sciences. Can we imagine programs able to manage all of them?

This problem is similar to that of office software: how to make programs (spreadsheets, word-processors etc) able to manage all printers - not only already existing printers, but also the printers which will appear in the next years. The solution is similar; by using device drivers. There is a printer driver for each printer, and we can make a driver for each measurement driver.

However, there are two differences between office programs and scientific programs. Scientific devices are more various than printers, and usually produced in small quantities. Scientist often must make himself/herself the program for his/her aim.

In the present work we will show how to use "scientific drivers" from the system Mensurasoft named by the Latin word mensura which means measure. Fig. 1 gives an illustration of the Mensurasoft system architecture: separation between programming for devices and programming for humans is an obvious necessity.

The necessity of Drivers

The construction of scientific devices is not easy. Making a pHmeter is a job for specialists, with special glasses for the electrode, special electronics, correction for temperature, etc. In the same way, building a precise scale to measure weight, or an accurate lux-meter are works of specialists. These specialists can supply a program for their devices, but not for others.

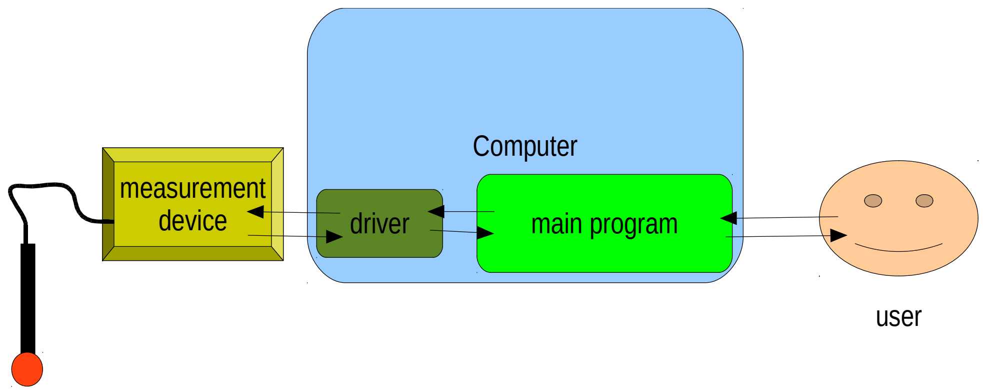

Fig. 1: Mensurasoft System

A professional scientist needs several scientific devices. For example, in ecology, we must measure accurately temperature AND pH AND light: there are not programs from devices makers for all them. A teacher of sciences needs to show to the students how to make an experiment. If there are several models of measurement devices in the classroom, with different software, the explanations will be very difficult for the teacher, who dreams about one piece of software for all these devices.

Drivers for measurement devices must be simple and easy to use, because science workers may not have great skills in programming. They must be usable with a lot of programming languages, in order to be used by many people.

Dynamic Libraries as Drivers

Dynamic libraries are pieces of compiled program, with functions callable by others programs; they are standardized and available in major operating systems: MS-Windows, Linux (named "shared objects") and Mac-OS [1]. Many programming languages can make dynamic libraries, often available as free software, but they must be compiled ones, such as: C/C++, Pascal, and some kinds of Basic.

On the other hand, almost all programming languages can use dynamic libraries, including office software such as Openoffice/LibreOffice, or the calculation specialists for example FreeMat [2], Scilab [3] Julia [4], R [5], and tools for general programing as Basic, C/C++, Pascal, Python, Ruby, Rebol.

In order to use dynamic libraries for all scientific devices, we must standardize the functions, giving them the same name in all the drivers. "Mensurasoft system" is a proposal for a possible standard: the names are short and consist of combination of letters.

Fundamental functions and additional properties

Computer must receive measures from the measurement device, and the driver needs a function for this input. In Mensurasoft system, the name of this input function has the letter "e" (as "entrée" in French, "eingang" in German, "entrada" in Spanish, "entrata" in Italian and "entry" in English). This is the main job.

If the device has an actuator, computer must send a value to the device, in order to change its state. In Mensurasoft system, the name of this output function has the letter "s" (as "sortie" in French, "salida" in Spanish).

Inputs and outputs can be digital or binary if there are only two values as "yes/no", "1/0", "on/off", represented with the letter "b" in Mensurasoft. They are analog if there are a great number of values, for example when we measure temperature, voltage or pH: in Mensurasoft, the letter is "a" for them.

Computers use different type of numbers: integers or reals ("floating"). Reals are often coded as "double" (double precision), using 8 bytes (64 bits), and integers can be coded on 4 bytes (32 bits). When the result of an input function (or the value to be set by an output function) is an integer, there is no special letter, but when the result or the value is a "double", a letter "d" is added.

Sometimes (often) there are several inputs. For example, the little Arduino Uno has 6 analog inputs; a simple thermometer only measures temperature at one sensor, but we could have the results either in kelvins or in Celsius or even in Fahrenheit system, thus three (3) analog inputs are needed. So the input functions have one parameter meaning the channel for measurement which is usually coded as integer without a special letter; 0 corresponds to the first channel, 1 to the second and so on.

Output functions have two parameters. The first is the number of the channel, as for input functions, and the second is the value to be set by the device (integer or "double").

In programming languages, there are several ways to pass parameters to a function: stdcall, cdecl, pascal, and others. "stdcall" is more used in Windows world, "cdecl" in Linux world, but this is no mandatory. In order to use all of them, we must put a prefix "std" if the declared function is "stdcall" and "c" if the function is "cdecl".

For example, in Mensurasoft system, the first analog input channel sending a real (double) is stdead(0) with the calling convention "stdcall"; the second binary output is stdsb(1,1) if it is set to "On" and stdsb(1,0) if it is set to "Off"; with "cdecl" convention, the third binary input is ceb(2) and the fourth analog output set to 3.44 is csad(3, 3.44).

That is enough for main programming languages, but there are some among them which want other functions. Some languages accept parameters only as "double", even for the channel number: instead of "d", we put "double" to show that all parameters and results are in double precision. Some languages deals with parameters and results only as "strings": we can add "str" at the end, and then the word stdeastr (0), for example, stands for the first analog input. Others need special functions (R, Scilab) in the dynamic library, but they are not included in the present work. A detailed outline of the naming procedure for functions in the Mensurasoft system is shown in Fig. 2.

Additional string functions

In Mensurasoft system each numerical function as already has been mentioned above has its name, coded by a string of characters. This string is a null-terminated string, as pchar in Pascal, zstring in FreeBasic, s in PureBasic or LPTSTR in C++. The name of the function begins by a "n" (as "name", "nom", "nomo", nome", "nombre"), e.g. nead(1) is the name of the second analog input. If a numerical function does not exist, its name is an empty string, i.e. a string whose length is 0; for example, cnead(133) is usually an empty string, because normal devices have only a few channels for measurement.

These "name functions" are very useful since:

a. when we do not know the channels of the device: we can ask cnead(0), cnead(1), cnead(2) and so on in order to find out them

b for general purpose software with menus and dialog boxes: to show all analog inputs the programmer is able to use a loop, asking the name of the analog input until this name is empty.

Additional string functions can be added for more comfortable use. A function "detail" could provide information about the device such as the company identity, the device name, the programmer's name, the date of programming etc. Even if there are not parameters, the name will be cdetail() and stddetail(). We could also define a title function, which will send a smaller string, for example only the name of the device, as: stdtitre() and ctitre(). This small string could be useful, e.g., as a title in a dialog box.

An optional function for calibration stdcalibration(ch) and ccalibration(ch) takes a string as input and gives a string as output. This function is useful only for specific devices, and the programmer can put anything in it.

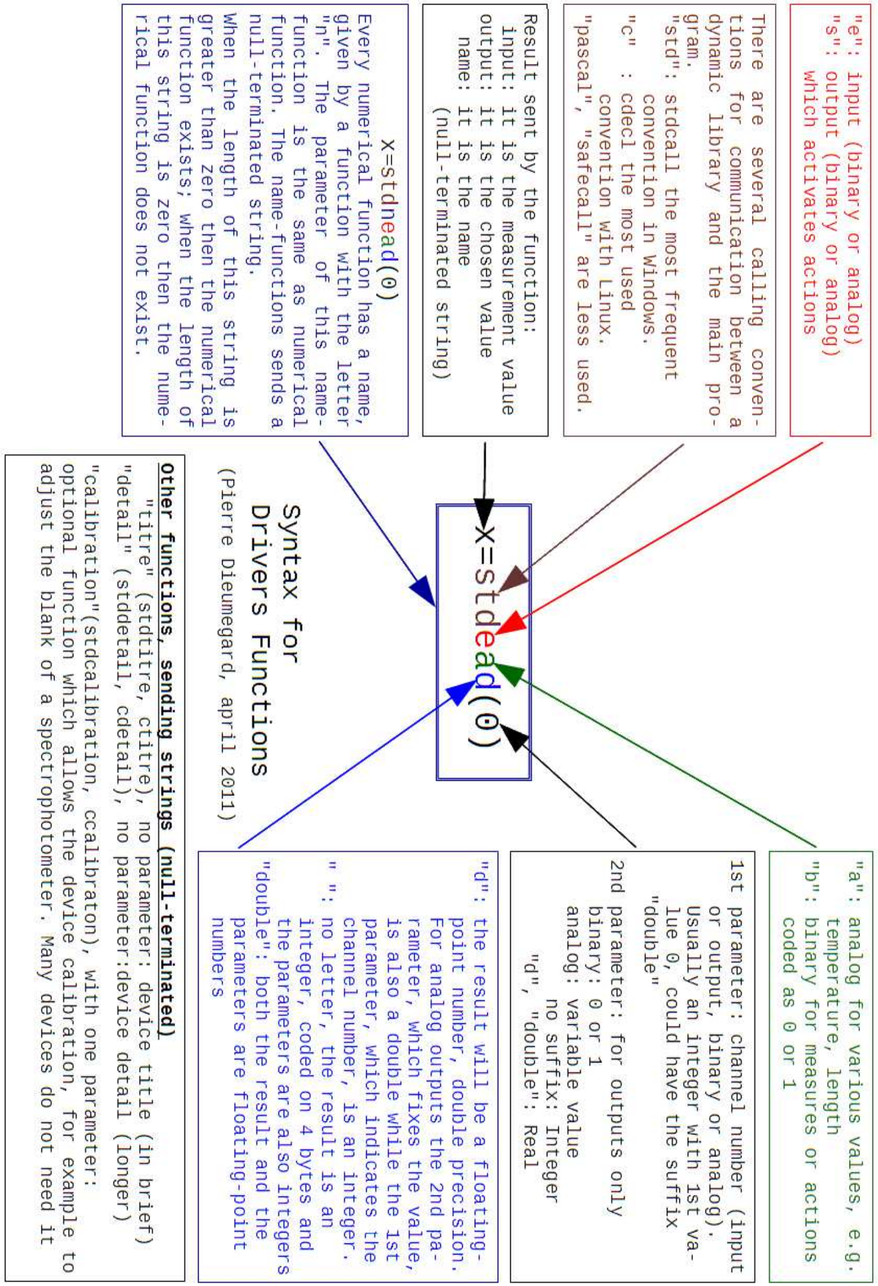

Fig. 2: Syntax of measurement function for Mensurasoft

A free project

Dynamic libraries are well known, and well standardized in programming languages. Mensurasoft is only a proposal for the adoption of a standard for the functions' names in order to allow all programmers to be able to use dynamic libraries of others.

Until now, all drivers and all application programs of Mensurasoft system are free software. The best license for drivers seems to be LGPL [6], because the drivers are small programs, and the best license for application programs seems to be GPL [7]. Nevertheless, we can imagine non-free drivers and non-free application programs, co-operating with free drivers and applications.

Three frequently asked questions need to be answered:

a. how to get existing drivers

b. how to make new drivers,

c. how to build programs calling them

a. The repository of the developed drivers for a lot of devices is at:

http://sciencexp.free.fr

mainly in French [8], but Google Translate may provide a decent translation for other languages. In this site, there are also application programs for Windows and Linux. They can use three (3) channels, for three (3) different drivers, make a graph of measured values, and save the values in files. They can transform the measured values by a formula (for example, to transform tension in volts from a pH-meter with analog output to pH). They are multilingual by language files.

Mensurasoft-PB [9] is written in PureBasic. A booklet in pdf is available in French, English, Spanish and Esperanto. Another program, Oscillo-PB, is very similar to Mensurasoft-PB, but is designed for fast measurement, like an oscilloscope. Fig. 3 shows a window from Mensurasoft-PB.

Mensurasoft-LZ is written in FreePascal/Lazarus. A booklet in pdf is available in French, English and Esperanto language. Fig. 4 shows an example of its usage in Linux illustrating the behavior of a common capacitor.

b. The best way to build a new driver is to modify an existing driver found on [8]. You can modify a "true driver" for a real device, or a "demonstration driver" [10].

Available drivers are in Delphi, FreePascal and Lazarus, FreeBasic, PureBasic, Oxygen-Basic, CodeBlocks, Dev-C++, C++Builder, Visual-C++.

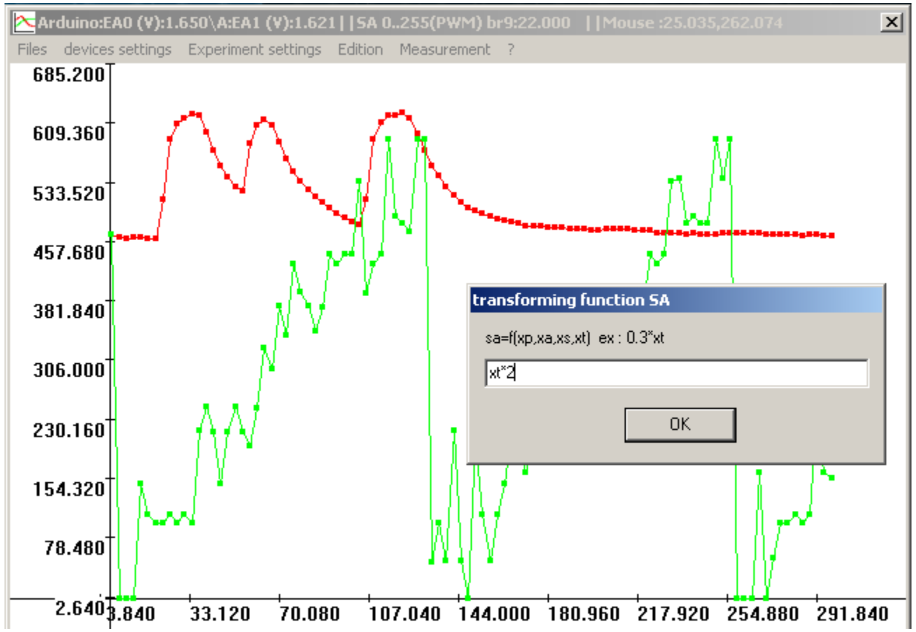

Fig. 3: An example window from Mensurasoft-PB

For example, if you have a new pH-meter, you can modify a driver only for a few functions: stdead, stdnead, stddetail, stdtitre. After compilation, you will be able to use Mensurasoft-PB, Mensurasoft-LZ and others with your new driver, and measure pH easily. Details of programming is available with booklets in French [11], English and Esperanto.

c. The booklets for programming also show how to use several languages to call these dynamic libraries: a lot of Basic, C/C++, Pascal, Logo, Python, Rebol, Ruby, Freemat, Scilab, Julia, R ...

Small programs could also be found at [12].

Conclusion

Mensurasoft system evolved for twenty years, and works well, at least for slow measurement, and for Linux and MS-Windows operating systems. For fast measurement, like an oscilloscope, limitations are due to the speed transfer of USB. A rather simple but systematic notation of functions implemented in dynamic libraries has been adopted and successfully applied and tested. The same system is available for Mac OS, but never has been tested.

The author encourage those who would like to be involved to this continuing effort and to contribute their driver for another device to the public domain via a link on the website [8] or to share a new written related application program. Do not hesitate to contact the author for any comment or related work.

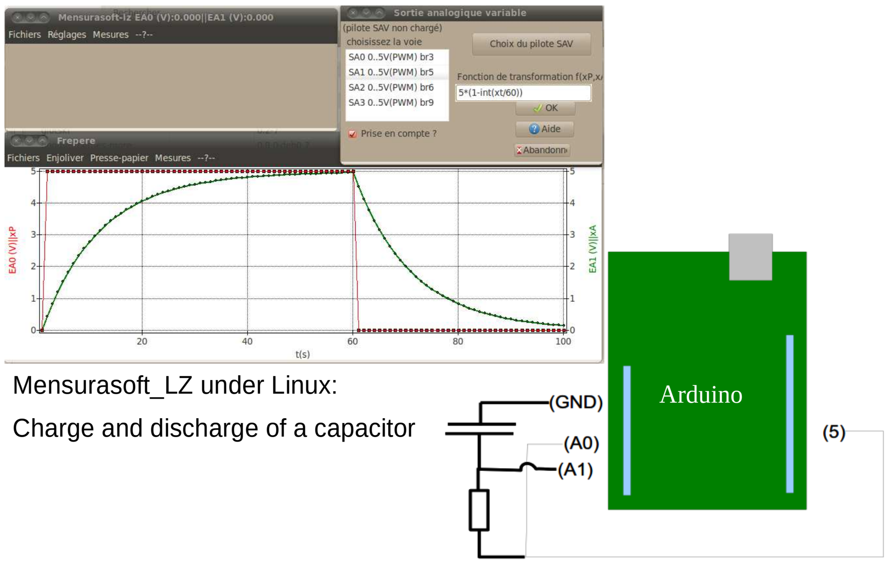

Fig. 4: Application of Mensurasoft-LZ in Physics

References

[1] Dynamic-link library: https://en.wikipedia.org/wiki/Dynamic-link_library

[2] FreeMat: http://freemat.sourceforge.net/

[3] Scilab: Scilab Enterprises, www.scilab.org

[4] Julia: http://julialang.org/

[5] R: www.stats4stem.org/r-math-functions.html

[6] GNU Lesser General Public License: "https://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License"

[7] GNU General Public License: https://en.wikipedia.org/wiki/GNU_General_Public_License

[8] Sciencexp - the site of the Experimental Sciences: http://sciencexp.free.fr

[9] Mensurasoft-PB: http://sciencexp.free.fr/index.php?perma=Mensurasoft-PB

[10] Pilot demonstration: "http://sciencexp.free.fr/index.php?perma=pilotes_demonstration"

[11] Mensurasoft: drivers of measurement devices and actuators system under MS-Windows and Linux: http://sciencexp.free.fr/index.php?perma=Programmation

[12] Demonstration applications: "http://sciencexp.free.fr/index.php?perma=Applications_demonstration"

Previous Publication in FUNKTECHNIKPLUS # JOURNAL

None

* About The Author

Pierre Dieumegard, was born in Limoges, France, in 1955. He teaches biology, geology and computer science in Lycée Pothier in Orléans, France. Since his thesis in plant genetics (1984), he started to use computers. After that, for many years, he was involved in research on the use of computers in science education.

pierre.dieumegard@ac-orleans-tours.fr

Antenna Design to Extremes: Practicable Suggestions from Pattern Analysis and RICHWIRE Simulation of Special Delta-Cross Loops

N.I. Yannopoulou, P.E. Zimourtopoulos *

Antennas Research Group, Austria

Abstract

In order to investigate the ability of pattern analysis and RICHWIRE simulation we usually use, to further handle antenna designs under certain extreme geometric conditions, we set up a number of limiting cases of our Delta-Cross loop, an antenna shape which we devised for the additional purpose of this research. Specifically, we vary the wire radius, segment vicinity, apex angle, and central configuration, while we deliberately keep constant its total length at almost two wavelengths, and we demonstrate the results for its main antenna characteristics: the complex vector radiation pattern and the complex scalar input impedance, as well as, the real scalars directivity and SWR, respectively. In this way, we conclude practicable suggestions for managing antenna problems which particularly include similar geometric arrangements.

Keywords

Antenna, pattern, RICHWIRE, simulation, extreme, delta-cross

Introduction

The analytical study of the cross wire antenna of four λ/2 delta elements, as equilateral triangles with side length λ/6, was carried out with the available theoretical tools of a first semester university course. Thus, it was considered as a 5λ/4 two wired open circuit transmission line, bended appropriately to form the four delta elements with total length 2λ. The remaining two λ/4 segments formed a transmission line segment of opposite currents and so, they were not taking into account as they do not radiate at all [1]. The current was then taken as the standing wave current of the corresponding segment of a 2λ equivalent symmetric, centered fed, dipole antenna. The antenna is shown in Fig. 1.

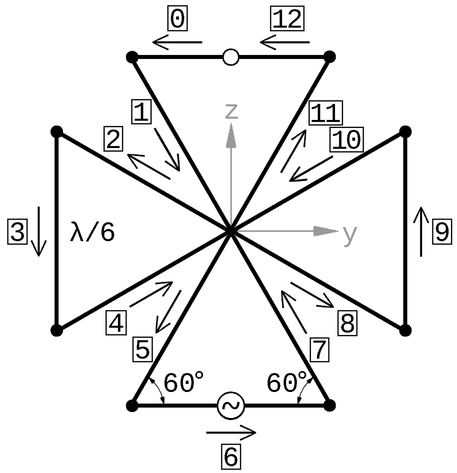

Fig. 1: The Planar Delta-Cross Shaped Loop antenna

The question is, was that approach adequately close to the current real situation, or accurately, near the simulated one and how can anyone verify it. From another point of view is the adopted theoretical model for the antenna geometry close enough to the simulated one which is undoubtedly closer to a possible built model. Some relative remarks are given in this paper. Additionally, the behavior of the antenna is presented, in detail, as the base angles of each delta loop varies between the two possible extremes values from 45° to 90°. The same tools will be used for the investigation, that is, the antenna theory for pattern analysis [2], the [RadPat4W] computer program of 2D/3D antenna patterns plots [3], the [RICHWIRE] simulation program [4] and the mini-Suite of software tools [5].

First Approach

The detailed procedure for the radiation pattern determination as a complex vector quantity had already be given in [1]. We repeat here the general relation that gives the total radiation pattern of all the 13 segments according to the law of superposition

$$\bar{\mathbb{E}} = \sum_{\nu=0}^{12} \begin{bmatrix}\dot{\mathbb{E}}_{\nu\theta}\\ \dot{\mathbb{E}}_{\nu\varphi}\end{bmatrix} = \begin{bmatrix}\dot{\mathbb{E}}_{\theta}\\ \dot{\mathbb{E}}_{\varphi}\end{bmatrix} = \dot{\mathbb{E}}_{\theta}\vec{\theta} + \dot{\mathbb{E}}_{\varphi}\vec{\varphi} = (\mathbb{E}_{\theta_R} + i\mathbb{E}_{\theta_I})\vec{\theta} + (\mathbb{E}_{\varphi_R} + i\mathbb{E}_{\varphi_I})\vec{\varphi} \quad (1)$$

which after a lot of manipulation and due to the antenna's symmetry was simplified as

$$\bar{\mathbb{E}} = \mathbb{E}_{0,12,6} + \mathbb{E}_{1,7} + \mathbb{E}_{2,8} + \mathbb{E}_{3,9} + \mathbb{E}_{4,10} + \mathbb{E}_{5,11} \quad (2)$$

were the real parts were eliminated. In [1], an upper bound of the absolute radiation pattern was illustrated. Actually it was the greatest possible upper bound since it resulted from the sum of the 13 absolute radiation patterns given by,

$$\left|\bar{\mathbb{E}}\right|_U = \sum_{\nu=0}^{12} \left|\bar{\mathbb{E}}_\nu\right| \tag{3}$$

while the absolute radiation pattern of the antenna is

$$\left|\bar{\mathbb{E}}\right| = \left|\sum_{\nu=0}^{12} \bar{\mathbb{E}}_\nu\right| = \left|\sum_{\mu=1}^{6} \bar{\mathbb{E}}_\mu\right| \tag{4}$$

where μ stands for each complex vector in (2). For the sake of research (5) gives another possible upper bound as the sum of the six absolute radiation patterns

$$\left|\bar{\mathbb{E}}\right|_u = \sum_{\mu=1}^{6} \left|\bar{\mathbb{E}}_\mu\right| \tag{5}$$

Fig. 2 shows the absolute radiation pattern of the antenna (4) among with its upper bounds of (3) and (5), all normalized to their maximum values in the three main planes, 1.27, 0.072 and 0.43, respectively for (3), (4) and (5). There is a noticeable difference, especially with the maximum upper bound given in [1]. The generalized triangle inequality dictates the relationship between them

$$\left|\bar{\mathbb{E}}\right| \le \left|\bar{\mathbb{E}}\right|_u \le \left|\bar{\mathbb{E}}\right|_U \tag{6}$$

as it is shown in Fig. 3, where all patterns are normalized with the max value of 1.27. Actually we have determine, in that way, the minimum and the maximum limit of the radiation pattern among which will be located every other pattern of any possible combination of the antenna elements.

The comparison between the analytical evaluated radiation pattern of the antenna and the one result in from simulation is given in Fig. 4(a). The significantly different shape of the radiation patterns lead us to investigate the simulated model.

The first thought is that a theoretical model assumes that instead of a wire antenna in space with current distribution on its surface, there is only a filament current distribution that follows the same shaped curve in space. Thus, we rerun the simulation program with variable wire radius from $1\text{x}10^{-3}$ [m], that was initially, to $1\text{x}10^{-9}$ [m], that is, reducing the wire radius as much as possible in order to achieve a filament "ExtraThin" wire, near to the theoretical prototype.

There are two factors that must taken into account:

i. The constraints imposed by the simulation program itself, relative to the used mathematical functions and

ii. The need of a quantity that will provide a criterion to stop reducing the radius.

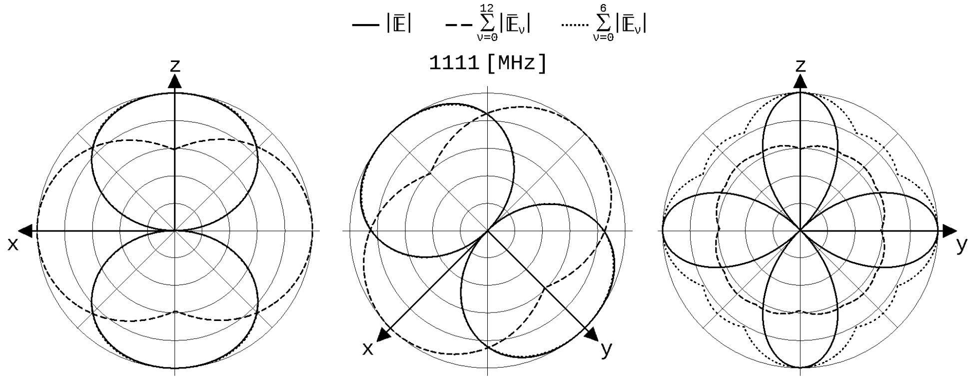

Fig. 2: Radiation patterns in 1111 [MHz]

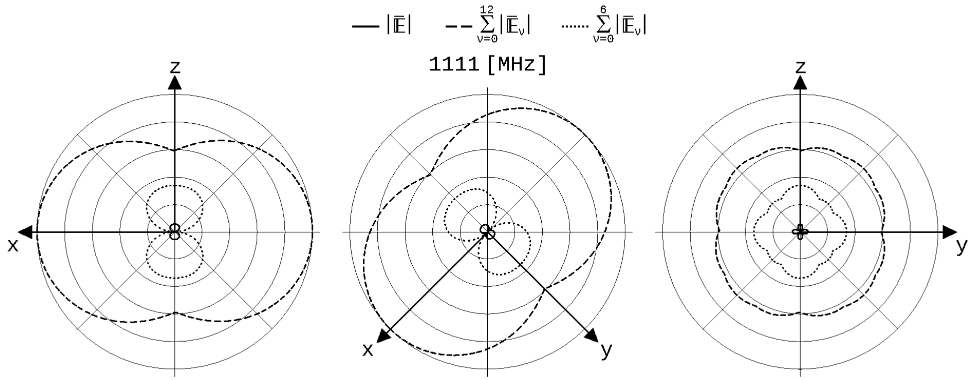

Fig. 3: Normalized patterns with common maximum value

The antenna Directivity is the right quantity to follow up since our concern is the radiation and it is shown in Fig. 5 with respect to the Wire Radius. The red line corresponds to the value 1 which is, as known, the directivity of an isotropic source, practically such an antenna does not exist, and it is the lower limit of Directivity. The two open circular points corresponds to the initial 1 [mm] radius (1.76) and to the radius of 0.5 [nm] (1.77) which represents the most thin wire that we may consider in the [RICHWIRE] simulation program. In that radius Directivity has almost the same value as it shown with the straight black line in the figure. Beyond that radius there is an unstable situation.

Fig. 4(b) illustrates all the patterns corresponding to the antenna with wire radius from 1x10^{-3} [m], to 1x10^{-9} [m]. The dotted black line is for the initial 1 [mm] and solid black line for the 0.5 [nm], which we named it as "ExtraThin". In Fig. (4c) the comparison between theory and simulation for the radiation pattern is shown, in the case of this ExtraThin wire.

Second Approach

The next idea is to find out if there is an explanation of the revealed difference between theory and simulation of the antenna radiation, shown in Fig. 4(a), relative to the analytical current assumption we considered.

As we already referred in [1] the simulation model was inevitable a little different from the theoretical one. Fig. 6 shows the three following steps:

(a) the initial theoretical model,

(b) the displacement of the four equilateral triangles by (2a+a/10)/2 away from their intersection point. Thus, a square is formed with side equal to 2a+a/10, where "a" is the used wire radius.

(c) the necessary connection of the neighboring sides of the delta elements by moving their apexes from the center of each side of the square to its corners.

A slight increase of the length of the base side of each delta element is necessary in order to reserve the 60° angle (60.006°) with its either sides. Therefore, the total length of each truncated triangle is by 2a+a/10 greater than the initial λ/2. Fig. 7 shows the small dashed square of Fig. 6, as a zoomed window.

This difference could result in a 1.55% to 4.67% change on the considered h quantity of the applied current distribution, if we take its ratio with respect to the total antenna length or to the length of the base of each delta element, respectively.

The significance of the base sides is due to the fact that the input source is located at the center of one of them. An investigation of the antenna radiation behavior in terms of h was carried out. It was varied in the range [4.5, 5.0] λ/4. Fig. 8 shows the horizontal maximum value of the normalized radiation pattern, i.e. on y axis for the selected arrangement. The noted point indicates the horizontal maximum value (0.51) of radiation pattern result in from [RICHWIRE].

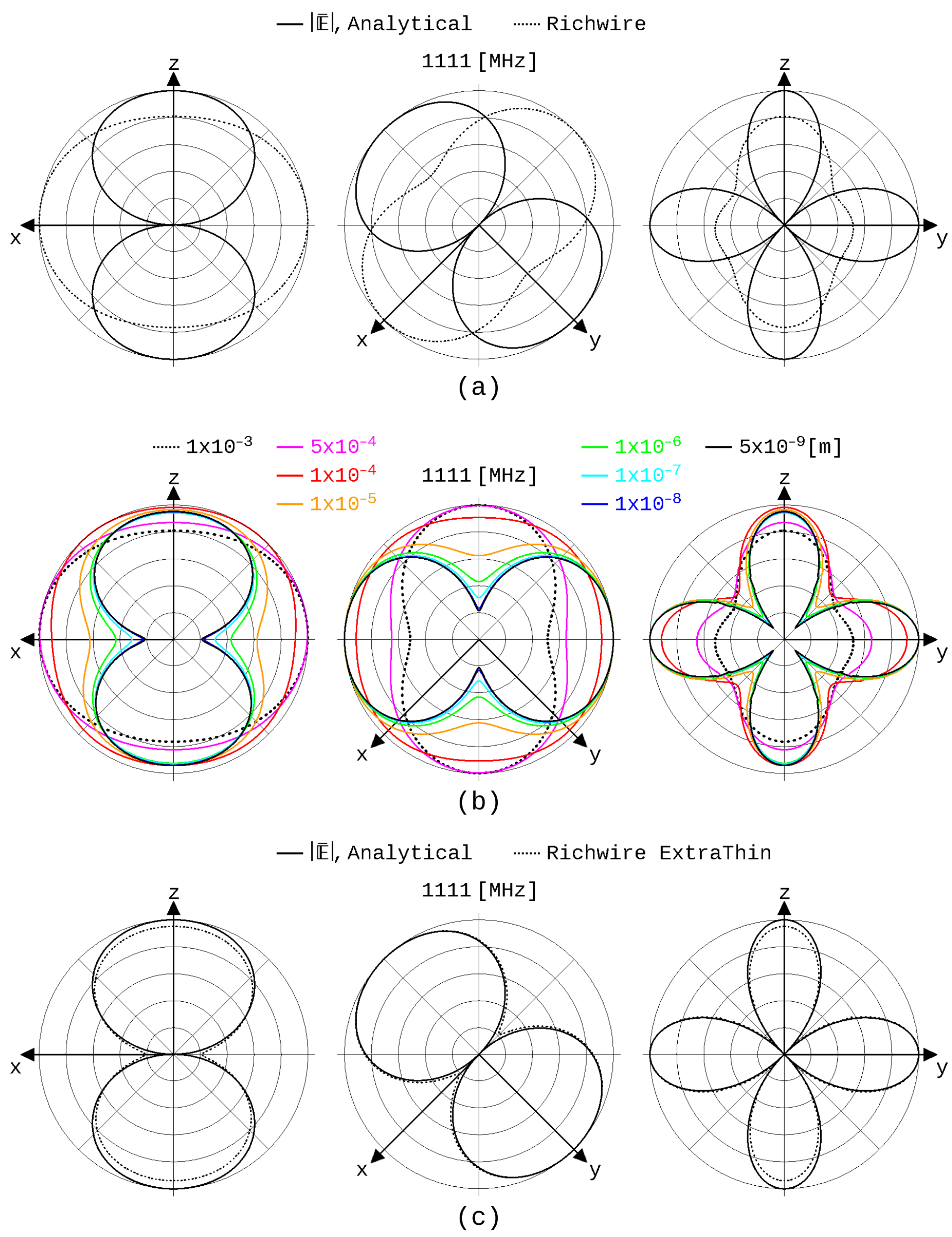

Fig. 4: Radiation pattern comparison

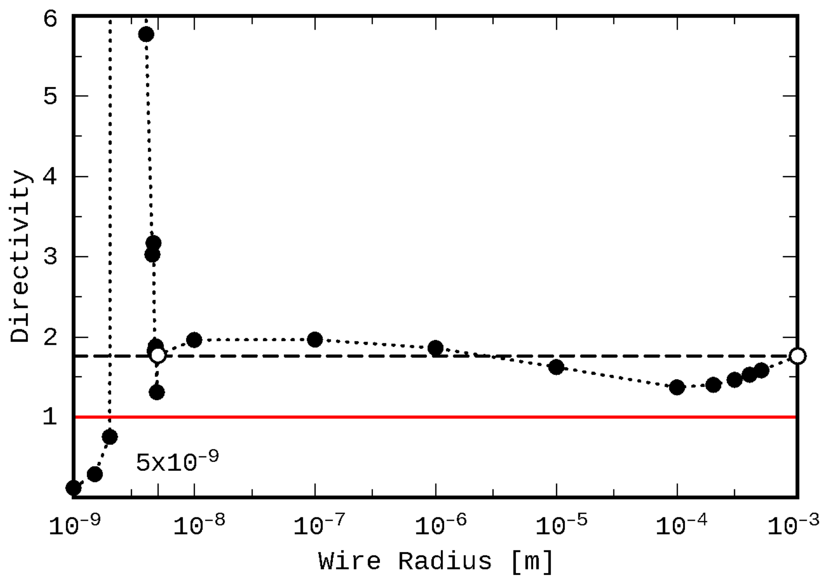

Fig. 5: Directivity versus Wire Radius

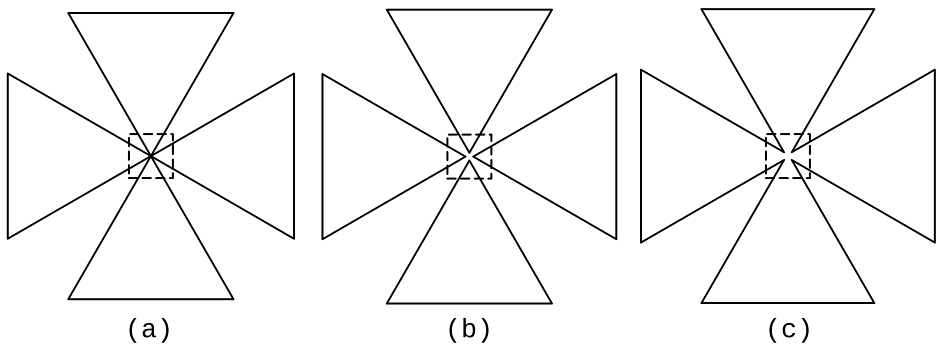

Fig. 6: Construction of the simulation model

This value corresponds to a h equal to 4.73λ/4, which is close enough to the 4.77 λ/4 as expected for the second percentage mentioned above ((100 – 4.67)% x 5.0 = 4.77). Since the other percentage leads to the value 4.92λ/4 of h, we decided to plot the radiation patterns in the three main planes for the range [4.5, 5.0] λ/4 of Fig. 9. From this figure is obvious that indeed the closest form

of the two patterns, theoretical and simulated, is achieved if we assume a sinusoidal current distribution with h equals to 4.73λ/4, as shown in Fig. 10.

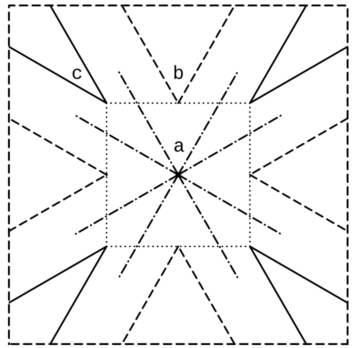

Fig. 7: Center detail

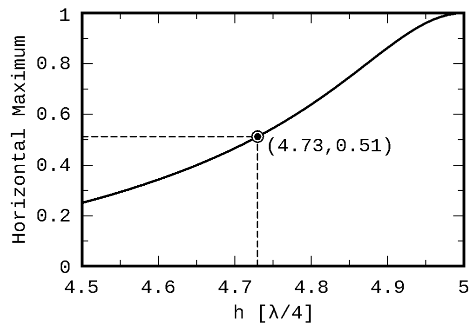

Fig. 8: Maximum value of radiation pattern on y axis

Modified Delta Element

The antenna was modified only with respect to the base angles of each delta element, keeping the perimeter constant, slightly greater than λ/2, as we already mentioned in the previous section, in order to improve its electric and electromagnetic characteristics [1], [7]. The result is a cross antenna of four isosceles triangles. We extent here the investigation to the whole possible range for the base angle, which is from 45° to 90°. Fig. 11 shows six representative geometries for 45°, 48°, 60°, 75°, 87° and 90° base angles. The base length of each antenna per wavelength is noted except in (f) where the side length is given instead.

The lower and upper limits lead to specific, degenerated cases, as it is shown in (a) and (f) of Fig. 11. In (a), a square is formed with the diagonal elements to be double as it is indicated by the small circular points near the center of the square. In fact, as the currents in those elements are of opposite direction it is expected that they do contribute almost nothing to the total radiation of such an antenna. At the upper limit, a cross is formed of double perpendicular elements. Since the distance between the λ/4 elements is too smaller than λ, and because their currents are again of opposite direction, no radiation is expected from them.

The remaining four small base elements of length (2a+a/10), where "a" is the used wire radius, i.e. 0.0021 [m], are small enough, as

$$\frac{0.0021}{\lambda} = 0.0078 < \frac{1}{10}\lambda \qquad (7)$$

and since the input source is on the one of the two parallel to y-axis elements while the other two parasitic elements are parallel to z-axis, λ/2 apart, we do expect a radiation like a small dipole on y-axis. These almost equivalent antennas are depicted in Fig. 12. Fig. 13 shows the comparison between the radiation patterns of the 45° Delta-Cross shaped loop antenna and of the equivalent simple Square loop one, with the same perimeter, on the three main planes, at 1111 [MHz].

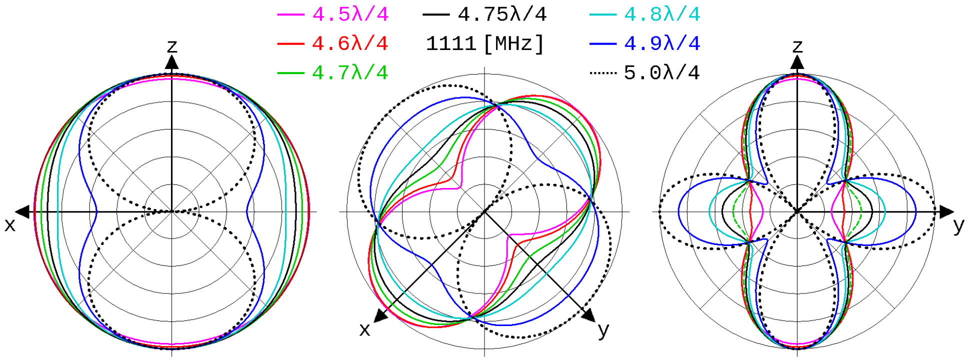

Fig. 9: Radiation patterns versus h

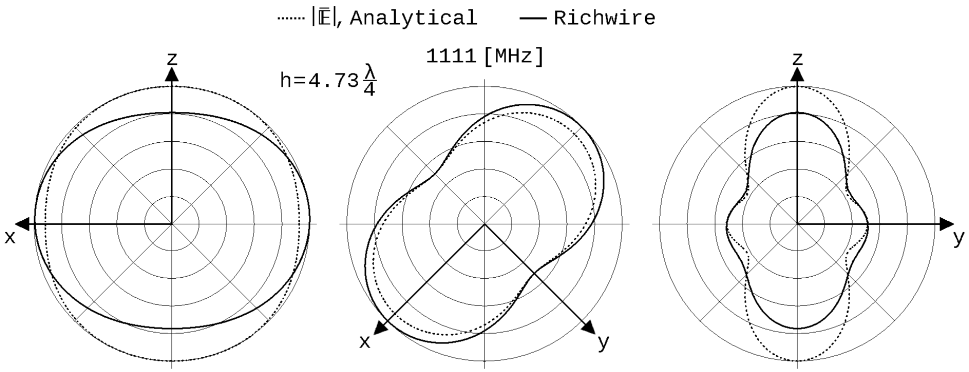

Fig. 10: Comparison of normalized patterns

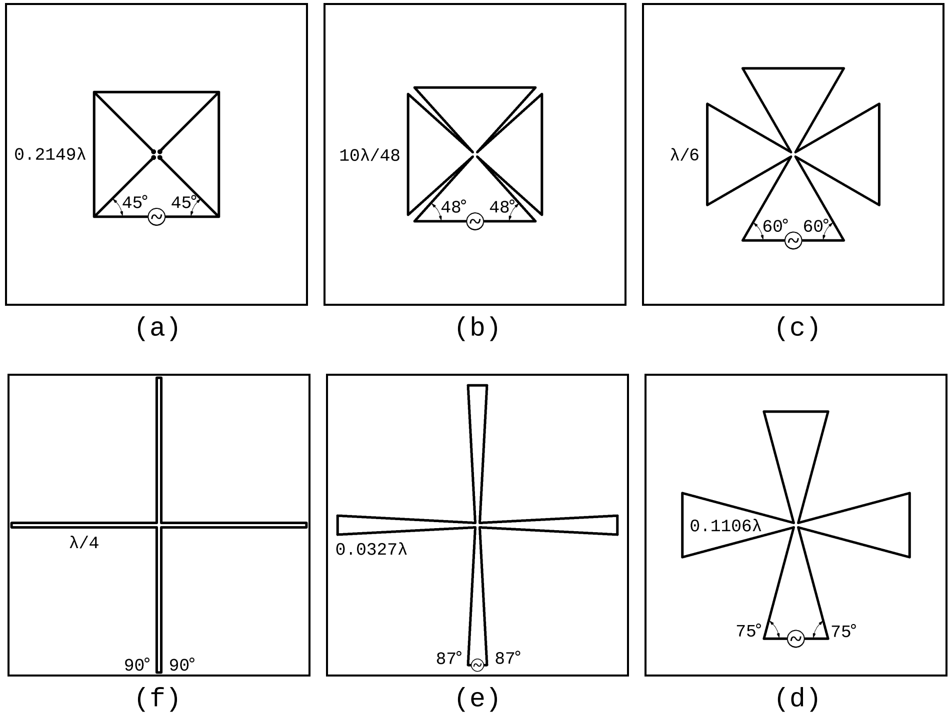

Fig. 11: Planar Delta-Cross shaped loop antennas

Fig. 14 contains the corresponding comparison between the 90° Delta-Cross shaped loop antenna, i.e. a rectangular shaped cross with λ/4 double sides, with a simple short dipole antenna parallel to y-axis. The agreement is obvious for both arrangements.

The investigation was performed in the range [45°, 90°] with 1° step at the frequencies 800, 1010 and 1111 [MHz], as these are the most interesting ones [1]. Fig. 15 shows the Directivity in [dB] with respect to the base angle. It is obvious that near the two limits of the degenerated cross antennas, the directivity shows a relative sharp change while at the frequency of 800 [MHz] has the more stable behavior. The two gray vertical lines correspond to 48° and 60° for the base angles of each delta element. The Directivity of the 48° Delta-Cross shaped loop antenna is greater at the higher frequencies.

SWR is given in Figs. 16, 17 and 18 for 50 [Ω], 75 [Ω] αnd 300 [Ω] respectively. The standing wave ratio is much better for the 48° not only for the 50 [Ω] but even for the 75 [Ω] characteristic impedance, at 1010 [Mhz].

Fig. 19 illustrates the Input Impedance [Z_{inp}] in terms of the base angle, in separate graphs for the Input Resistance [R_{inp}] and the Input Reactance [X_{inp}]. A narrower range for both the Resistance and the Reactance is given in Fig. 20, in order to clarify the behavior of the Input Impedance in lower values. It is obvious that the curve for 800 [MHz] is relative smooth both for the real and imaginary parts of the impedance while in the higher frequencies sharp changes appears as the base angle varies between 45° and 60°. In 1010 [MHz] a resonance is achieved for the 48° Delta-Cross antenna.

Conclusion

Two practical suggestions are given, in the present work, in order to explain the radiation pattern results from pattern analysis and simulation. The first is based on the assumption of the ExtraThin wire for simulation and the second to the inevitable change of the ideal geometrical representation in analysis to a more realizable model in simulation.

Both these different paths lead us to a good agreement between the, analytical and simulated, produced radiation patterns. Although, the simulation prototype could be closer to a constructed antenna and so the radiation pattern is expected to be as the result of simulation, the final step of the construction and measurement it is absolutely necessary to justify, as always, our expectations.

During this procedure the lower and the upper bound of the absolute radiation pattern for any possible deconstruction and reformation of the antenna in its elements was explicitly determined.

The detailed investigation on the range of the variable base angles reveals the two degenerated cases of 45° and 90° as the apex lower and upper bound, that is the design extremes, where the antenna is actually transformed to an equivalent common square loop and a short center fed dipole on y-axis respectively. It also becomes evident why the antenna of 48° base angle was selected as the improved Delta-Cross shaped loop antenna.

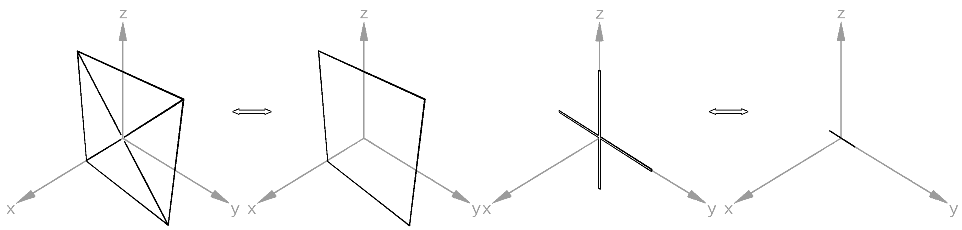

Fig. 12: Almost equivalent antennas in terms of radiation

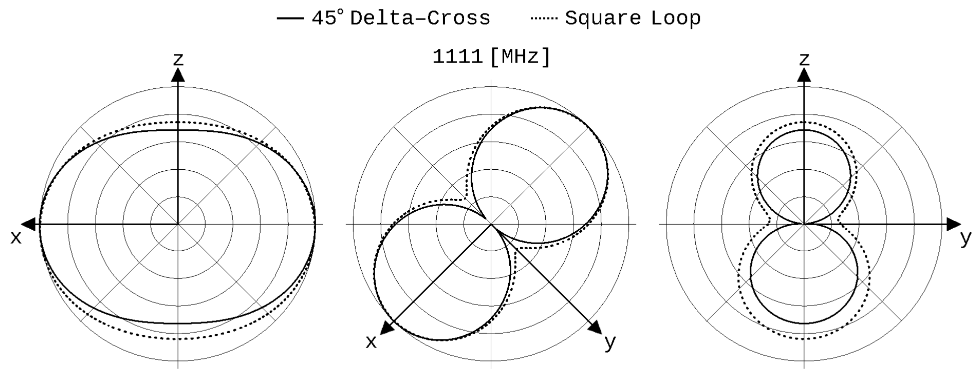

Fig. 13: 45° Delta-Cross and Square loop radiation

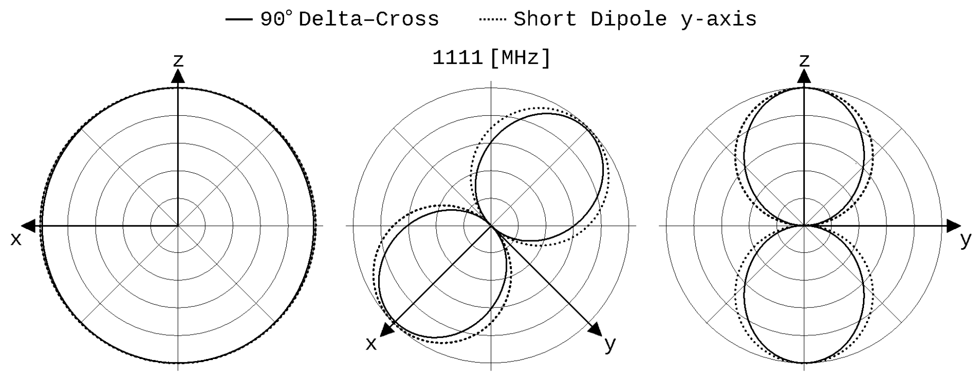

Fig. 14: 90° Delta-Cross and Short dipole on y-axis radiation

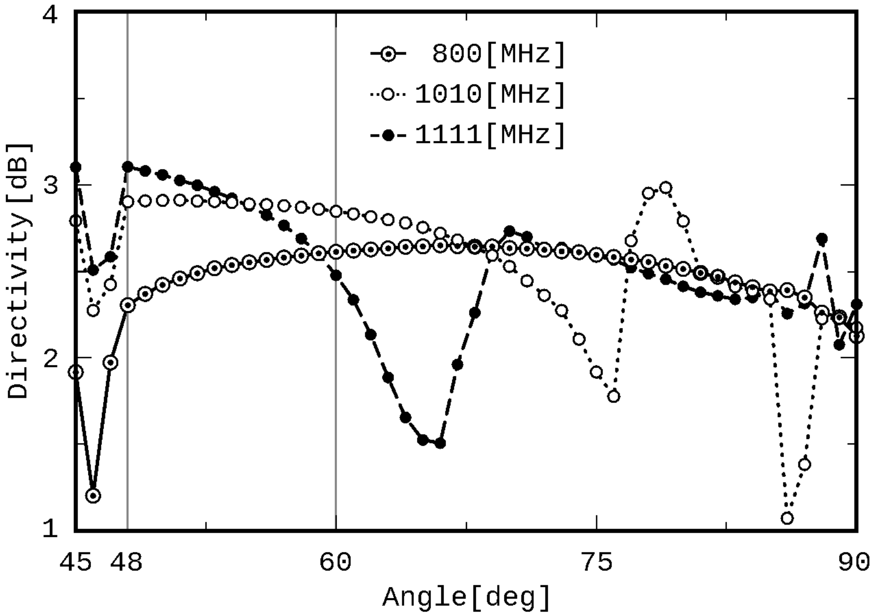

Fig. 15: Directivity versus base angle

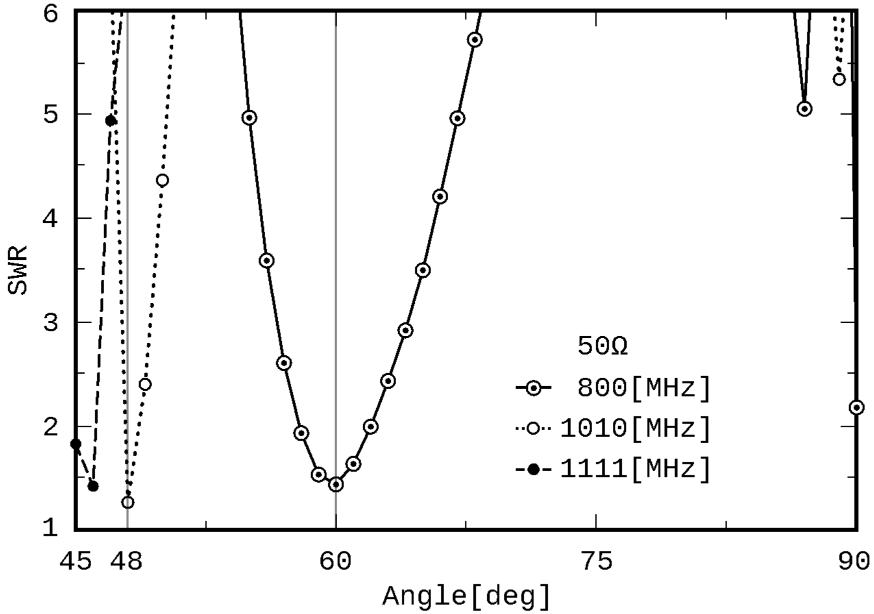

Fig. 16: SWR for 50 [Ω] characteristic impedance

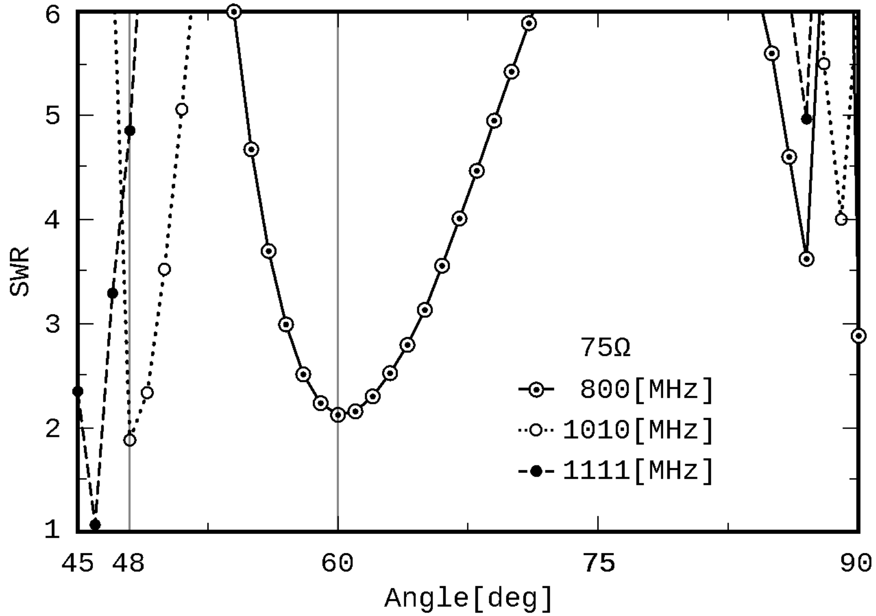

Fig. 17: SWR for 75 [Ω] characteristic impedance

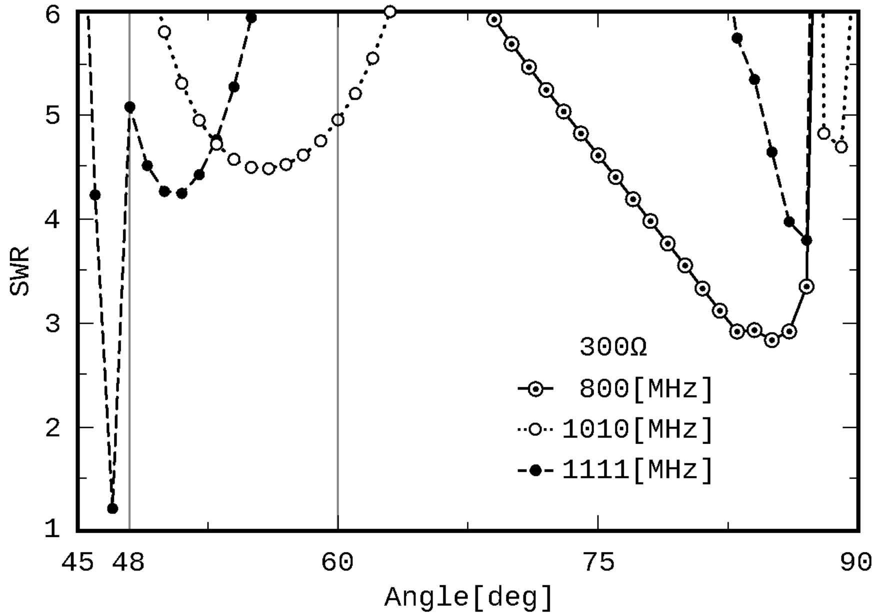

Fig. 18: SWR for 300 [Ω] characteristic impedance

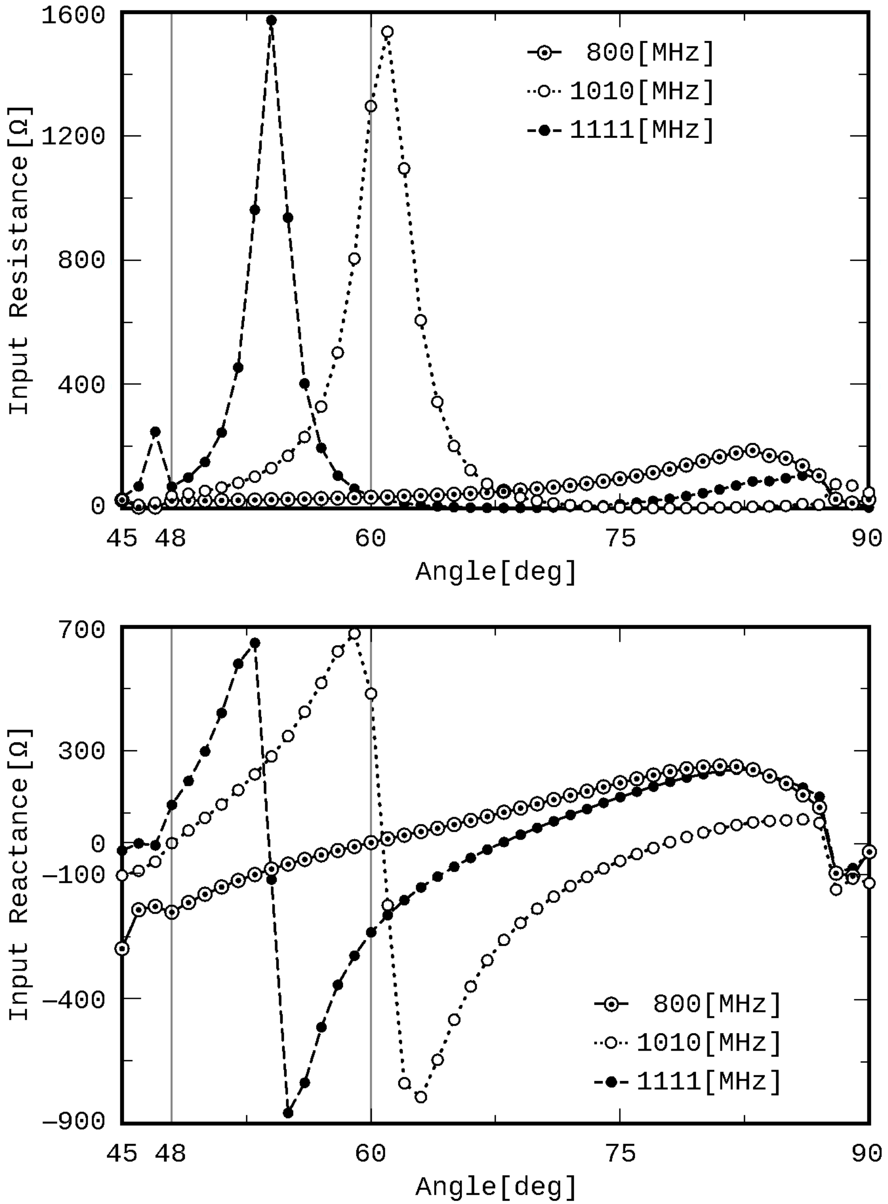

Fig 19: Input Impedance versus base angle

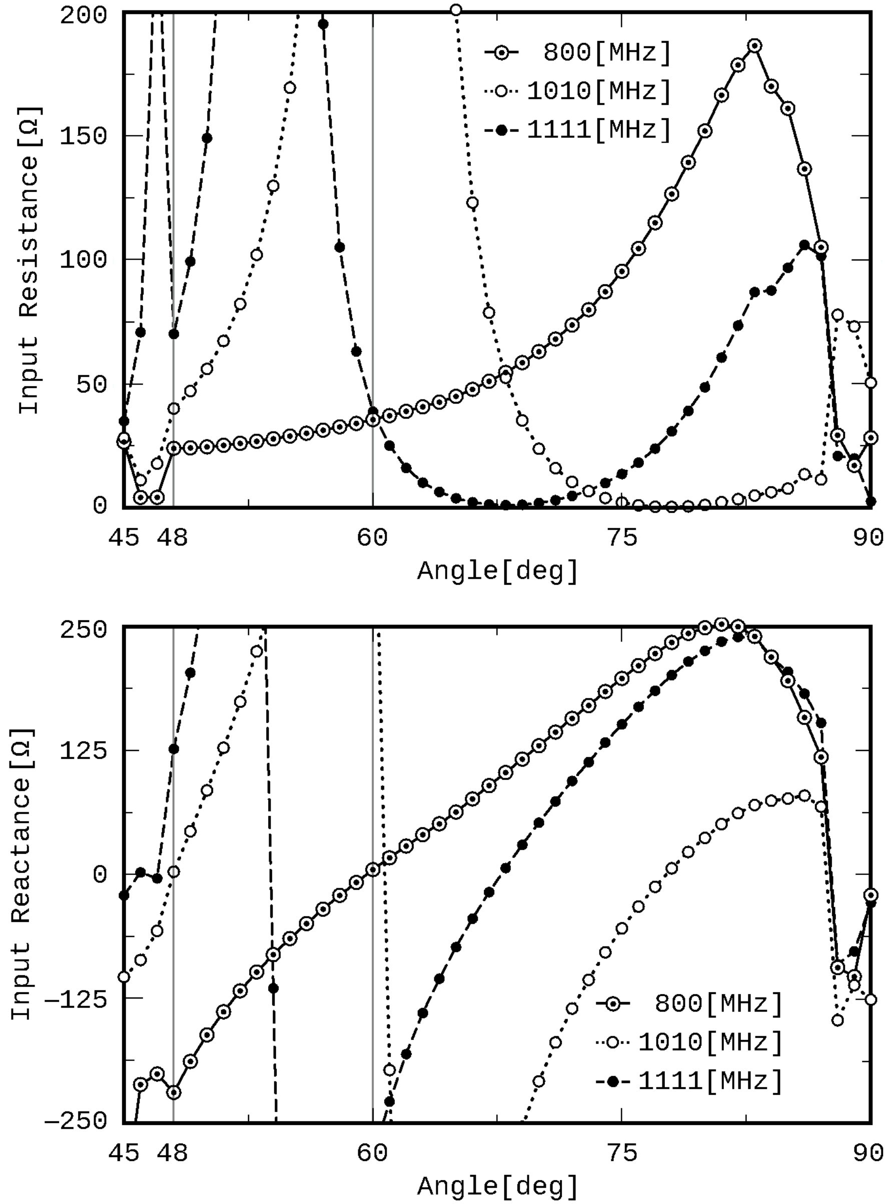

Fig 20: Input Impedance in a narrower range

References *

[1] Papadaniel G.A., Yannopoulou N.I., Zimourtopoulos P.E., "A Planar Delta–Cross Shaped Loop Antenna: Analysis and Simulation – The 2 WL Case", FunkTechnikPlus # Journal, Issue 7, Year 2, February 2015, pp. 19-36
www.otoiser.org/index.php/ftpj/article/view/53
www.otoiser.org/index.php/ftpj/issue/view/2015.05.31

[2] Zimourtopoulos P., "Antenna Notes 1999-", "Antenna Design Notes 2000-", www.antennas.gr/antennanotes/ (in Greek)

[3] Yannopoulou N., Zimourtopoulos P., "A FLOSS Tool for Antenna Radiation Patterns", Proceedings of 15th Conference on Microwave Techniques, COMITE 2010, Brno, Czech Republic, pp. 59-62

[4] Richmond J.H., "Computer program for thin-wire structures in a homogeneous conducting medium", Publication Year: 1974, NTRS-Report/Patent Number: NASA-CR-2399, ESL-2902-12, DocumentID: 19740020595, http://ntrs.nasa.gov

[5] Yannopoulou N., Zimourtopoulos P., "Mini Suite of Antenna Tools, Educational Laboratories, Antennas Research Group, 2006,
www.antennas.gr/antsoft/minisuiteoftools/

[6] Yannopoulou N.I., "Study of monopole antennas over a multi-frequency decoupling cylinder", PhD Thesis, EECE, DUTh, February 2008 (in Greek), pp. (2-6)-(2-9)

[7] Papadaniel G., "The Eisernes Kreuz Antenna [Prototype ARG-06]", Diploma Thesis #40, ARG–Antennas Research Group, DUTh, 2007 (in Greek)

*Active Links: 27.09.2015 - Inactive Links : FTP#J Link Updates: http://updates.ftpj.otoiser.org/

Previous Publication in FUNKTECHNIKPLUS # JOURNAL

"A Planar Delta–Cross Shaped Loop Antenna: Analysis and Simulation – The 2 WL Case", Issue 7, Year 2, pp. 19-36

*** About The Authors**

Nikolitsa Yannopoulou, Issue 1, Year 1, p. 15

Petros Zimourtopoulos, Issue 1, Year 1, p. 15

[This Page Intentionally Left Blank]

[This Page Intentionally Left Blank]

In case of any doubt,
download the genuine papers from
genuine.ftpj.otoiser.org

FRONT COVER VIGNETTE

A faded synthesis of an anthemion rooted in a meandros

The thirteen-leaf is a symbol for a life tree leaf.
"Herakles and Kerberos", ca. 530—500 BC,
by Paseas, the Kerberos Painter,
Museum of Fine Arts, Boston.

www.mfa.org/collections/object/plate-153852

The simple meandros is a symbol for eternal immortality.
"Warrior with a phiale", ca. 480—460 BC,
by Berliner Maler,
Museo Archeologico Regionale "Antonio Salinas" di Palermo.

commons.wikimedia.org/wiki/File:Warrior_MAR_Palermo_NI2134.jpg