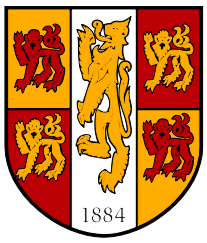
DOCTOR OF PHILOSOPHY

Use of Microwaves for Skin Lesion Treatment

Horwitz, Paul

Award date:
2017

Awarding institution:
Bangor University



PRIFYSGOL
BANGOR
UNIVERSITY

Use of Microwaves for Skin Lesion Treatment

Paul Horwitz

School of Electronic Engineering

Bangor University

March 2017

Acknowledgements

I would like to express my thanks to the Knowledge Economy Skills Scholarship (KESS) programme for funding this research without which it would never have taken place.

I am also deeply appreciative and grateful for a kind and generous donation to the MMSRG that enabled much of the prior skin treatment work to take place. This prior work included studies into potential applications such as BCC and SCC in addition to pre-clinical research that produced valuable reference data. This present work is based on and is a continuation of these prior studies.

Also I would like to thank Malcolm White for his help and depth of knowledge which gave me some insight into the complexities of this subject.

And my thanks and appreciation to Chris Hancock who's enthusiasm, knowledge and consistent optimism helped overcome the most difficult problems.

And finally thanks and love to my family and friends for putting up with my absence for long periods of time while I read countless articles and then putting up with me talking about it endlessly.

Abstract

Against a backdrop of rising incidence of Non-Melanoma Skin Cancer, substantial previous work has resulted in a 14.5GHz Microwave Applicator that ablates small areas of dermal tissue. This previous work has highlighted the need for improvements in uniformity and the estimation and control of heating dosage.

The aim of this work was to improve heating uniformity and control by numerically modelling the thermal characteristics of the existing Microwave Applicator and so gauge the effects of direct thermal feedback, surface cooling, temperature bias and small uncertainties in power delivery to the time and depth of ablation and so provide a guide to thermal dosage and maximum ablation depths achievable for 5.8GHz, 14.5GHz and 24.125GHz; and to produce a simple Electromagnetic numerical Model to identify methods of improving the uniformity of the heating profile at 14.5GHz. And finally, to use these results to guide the design of a New Microwave Applicator using a commercial modeller (CST Design Studio).This New Microwave Applicator operating at 14.5GHz incorporates an integral thermal camera viewing the target tissue which provides direct thermal feedback during the ablation process which is used to control the power delivery. The new mechanical construction incorporates PTFE side sections to flatten the heating profile.

Use of Microwaves for Skin Lesion Treatment

Contents

Abbreviations and Definitions used

General

EM	Electromagnetic
MSA	Microwave Skin Applicator – refers to the prior Applicator
NMSA	New Microwave Skin Applicator – refers to the new design presented here
PEC	Perfect Electric Conductor
PMC	Perfect Magnetic Conductor

Mathematical		Units
A	Current	Amps
α	attenuation constant in	dB or Nepers/metre
β	phase constant in	Radians/metre
b	Perfusion	W/m^3 °C
c	Speed of light in a vacuum	m/s
C_p	Specific heat	J/Kg °C
e_0	Permittivity of free space	8.85×10^{-12} Farads/metre
e	Base of natural logarithm	2.7128
ε_c	Absolute complex permittivity	Farads/metre
ε'_c	Real part of Absolute Complex permittivity	Farads/metre
ε''_c	Imaginary part of Absolute Complex permittivity	Farads/metre
f	frequency	Hz
F	Capacitance	Farad
h	Power Transfer Coefficient	W/m^2 °C
J	Energy	Joules
K	Thermal conductivity	W/m °C
m	Distance	Metres
ρ	Density	Kg/m^3
σ	Conductivity	Siemens/metre
s	Time	Seconds
SAR	Specific absorption rate	W/Kg
T	Tesla for magnetic application	Kg / A s^2
T	Temperature in for thermal application	°C
T_a	Ambient temperature	°C
T_b	Blood temperature	°C
T_s	Surface temperature	°C
tan δ_c	Loss Tangent	Dimensionless
μ_0	Permeability of free space $4\pi \times 10^{-7}$	V s/A m
μ_r	Relative permeability	dimensionless
V	Electrical potential	Volts
W	Power	Watts

Medical

5-FU	5-Fluorouracil
AK	Actinic Keratosis
BCC	Basal Cell Carcinoma
C&E	Curettage and Electrodessication
DICL	Diclofenac
IMIQ	Imiquimod
IngMeb	Ingenol Mebutate
MMS	Mohs micrographic surgery
NMSC	Non-Melanoma Skin Cancer
OCT	Optical Coherence Tomography
PDT	Photodynamic Therapy
RT	Radiotherapy
SCC	Squamous Cell Carcinoma
SE	Surgical Excision

Chapter 1 Background and Rationale

Contents

Introduction

Probably most western households are familiar with the use of microwave ovens, which irradiate food with electromagnetic wave energy which is converted to heat. Apart from direct electrical conduction, at high frequencies this heating effect is due to the oscillating electric field doing work on molecular forces [1]. This heating effect may be used to heat human body tissue for therapeutic reasons and is now widely studied and used in various contexts such as a 915MHz balloon catheter for treating benign prostatic hypertrophy [2], a comparative study of 434 and 915MHz antennae for non-invasive breast cancer therapy [3] and a comparative study of radio frequency and microwave ablation in liver cancer [4]. More recently , higher frequency use of microwaves has been facilitated by developments for 5.8GHz and 14.5GHz communications in Gallium Arsenide and Gallium Nitride devices that have aided the production of microwave power amplifiers capable of in excess of 100W of continuous power at reasonable cost and desktop size [5].

The use of Microwaves to treat skin lesions is a developing technology which requires knowledge of the subtypes and epidemiology of applicable skin lesions and relevant skin structures together with current treatments. Substantial previous microwave work has already been undertaken to evaluate its applicability to the field of skin lesion treatment which has revealed further needs and requirements. These topics are described below.

Skin Structure

In order to describe aspects of skin lesions and treatment, a brief description is presented below together with an image of histology of partially ablated porcine skin which illustrates some dermal structures. The human skin comprises the inner Hypodermis then moving outwards, the Dermis and Epidermis [6] and provides defensive barrier against various threats such as bacteria and fungi [7].

Epidermis The outer layer of human skin is itself multi-layered with thickness varying from 0.06mm (eyelids) to 0.8mm (palms and soles). The epidermal layers consists (from inner to outer) of the Stratum Basale (Basal layer) which consists a single layer mainly of keratinocytes and also melanocytes (melanin), Langerhans (antigen) and Merkel (tactile) cells [6] [7]; The Stratum Spinosum (Squamous cell layer) consisting or 2 to 6 cell layers; The Stratum Granulosum consisting of 1 to 5 cell layers; The Stratum Lucidum which comprises 4-6 layers and is only present in thick (hairless) skin; The Stratum Corneum which comprises 10 to 15 layers of dead, anucleate, flattened cells filled with keratin and held together by Corneodesmosomes [6] [8]. The epidermis can be up to 1.4mm on the soles and the full thickness of thick skin can exceed 5mm whereas the full thickness of thin skin can be 1-2mm [8].

Dermis This vascularised layer is 3-5mm thick and comprises various cell types such as melanocytes, macrophages and fibroblasts. Collagen and elastic fibres provide strength and elasticity [6]. It comprises 2 layers: the outer Papillary and inner Reticular and contains hair follicles, sweat and sebaceous glands, nerves and Meissner corpuscles (touch sensitive receptor), Pacinian corpuscles (pressure sensitive receptor), Ruffini corpuscles (Stretch sensitive) [8].

Hypodermis This is a layer of fat cells proving heat and shock insulation and energy storage and like the dermis, collagen and elastic fibres provide strength and elasticity [6]. It varies in thickness depending on nutrition, gender and location on the body and is highly vascularised permitting for example hypodermal insulin injection [8].

An example of ablated porcine skin is shown below in Figure 1

Example of Ablated Porcine Skin

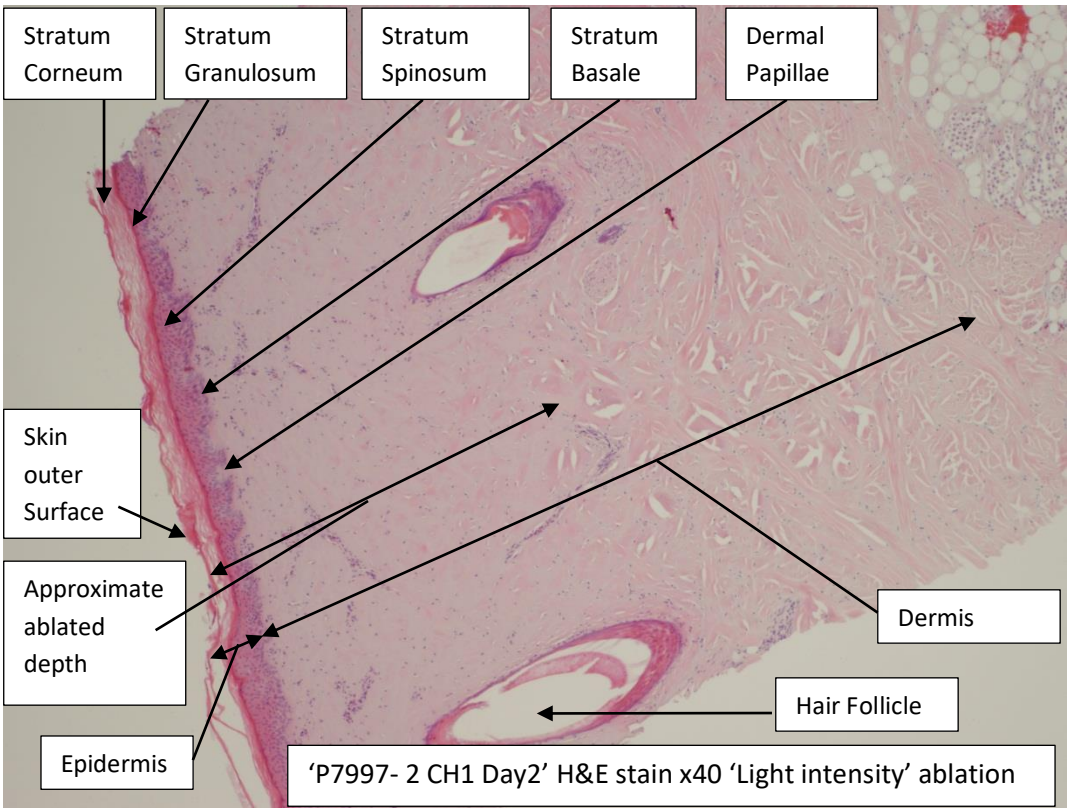


Figure 1 Histology of partially ablated porcine skin.

Reproduced with the permission of Professor C. Hancock on behalf of Bangor University.

The H&E stain has been interpreted and annotations added by this author by reference to [9].

Microwave Heating of Biological Tissue

As is outlined later, the physical extent of the heating effect is important both to destroy unwanted tissue and to preserve viable tissue, so the following useful formulae and their implications are noted below.

From [1] pg. 40 and sec 2.3.1:

{1} $\mathcal{E}_x(z,t) = C_1 e^{-\alpha z}\cos(2\pi ft - \beta z)$ Instantaneous field strength in V/m for real C_1, z = distance travelled by wave in medium.

{2} $\varepsilon_c = \varepsilon_c' - j\,\varepsilon_c''$ Absolute complex permittivity of the material to be heated, a frequency dependant measureable quantity which incorporates conduction.

{3} $\tan\delta_c = \varepsilon''/\varepsilon'$ Loss tangent

{4} $P_{av} = \frac{1}{2}2\pi f\varepsilon''\,|E|^2$ Watts/m^3 Average power at any arbitrary point. |E| = absolute electrical field strength in V/m.

{5} $d = \frac{c\sqrt{2}}{2\pi f\sqrt{\mu_r\varepsilon_r'}\sqrt{\left[\sqrt{1+\tan^2\delta_c}-1\right]}}$ Skin depth in metres

Skin depth is defined as the point where the amplitude falls to 1/e and thus where the field strength falls from 100% at the surface to 37% or using the relationship between Power and $|E|^2$ as shown in {4} to 13.5% of the power, thus 86.5% of the power is dissipated within the skin depth. From [1] d = 1/α and by choosing $e^{-\alpha z}$ = √2 and solving for z, we get z = 0.34d which means that 50% of the power is dissipated in approximately the first 1/3 of the skin depth and the remaining 36.5% is dissipated in the remaining 2/3 of the skin depth.

In summary, by choosing f and using known values of ε_c for a particular material, we can predict the power loss profile for an incident EM wave. Some relevant examples using the calculator 'Dielectric Properties of Body Tissues' [10] for 'DrySkin' and calculating the 50% depth are shown in Table 1 below.

Table 1 Penetration depths for 'DrySkin' at various frequencies

Frequency GHz	Skin Depth (86.5% power) mm	Depth for 50% Power mm
2.45 *	22.6	7.5
5.8 *	8.6	2.76
14.5	2.1	0.7
24.125 *	1.1	0.37
61.25 *	0.5	0.16

* ISM Industrial, Scientific, Medical unlicensed bands [11]

The temperature rise in the treatment area will be determined in part by a combination of energy dissipation and heat conduction, so the required depth of penetration and extent will depend on frequency and heating time.

Temperatures required to destroy and to preserve human keratinocytes

In order to destroy unhealthy cells sufficient temperatures must be reached. A study of thermal injury to *in vitro* human keratinocytes found that cells died immediately on exposure to temperatures above 72°C, after exposure to 62°C for 1 second most cells died by non-apoptotic process and possibly all cells were non-viable within 72 hours; when exposed to 72°C for 1 second all cells died almost immediately by non-apoptotic process [12]. To minimise healthy tissue damage, normal cells need to be spared, a study into thermotolerance of human keratinocytes [13] found that 91.8% of cells were still viable after exposure to 46°C for 20 minutes and using the CEM_{43} method[1] to convert temperature and exposure time described in [14] for *in vitro* human tissue above the breakpoint of 43.5°C, the above temperature and time converts to a CEM_{43} of 255 minutes (equivalent to 40 seconds at 50°C). Consequently it would appear that a suitable aim for ablation temperatures might be to achieve at least 62°C for 1 second while attempting to expose healthy tissue to less than 255 CEM_{43} minutes with the proviso that these figures are based on *in vitro* studies of normal keratinocytes. For *in vivo* applications 60°C may be more favoured by clinicians [15] and as discussed in Chapter 2 page 32, the more conservative 80 and 40 CEM_{43} minutes is used for Muscle/Skin and Fat respectively.

[1] From [14], the CEM_{43} calculation provides a means of establishing a heat dosage related to temperature and the time duration spent at different temperatures:

$$\mathrm{CEM}_{43} = \mathrm{CEM43°C} = \sum_{i=1}^{n} t_i \ . R^{(43-T_i)} \ minutes$$

Where T_i is the average temperature for period i which is of duration t_i minutes. The R-Value differs between species and has different values below and above the breakpoint; For Man *in vitro* , the breakpoint is 43.5°C with R = 0.233 for T<43.5 °C and R = 0.428 for T>43.5°C.

Non-Melanoma Skin Cancer (NMSC)

Common non-melanoma skin cancers:

Squamous Cell Carcinoma (SCC) and Basal Cell carcinoma (BCC) are the most common types of NMSC, with NMSC being the most common cancer affecting fair skinned people with rates increasing in parts of Europe, North America and Australia at 3-8% p.a., SCC and BCC occur mostly in sun-exposed areas of the body due to UV radiation [16] and there may be an association with diet especially with previous SCC cancer [17].

Incidence

There is wide incidence variation between various parts of the world with the highest incidence of BCC in Australia with more than 1000 per 100,000 person-years, and other areas such as England in the UK with BCC at 76.21 per 100,000 person-years and SCC at 22.65 per 100,000 person-years, New Hampshire in the USA in 1980 had a BCC rate of 170 per 100,000 person-years [18]. BCC accounts for approximately 30% of new cancer diagnoses in the USA with NMSC comprising 80-85% BCC and SCC 15-20%; There are 1.3 million cases of NMSC in the USA per annum, a figure which may double in 30 years; BCC rarely metastasizes unlike SCC which is more likely to be invasive and both types can cause substantial disfigurement [19].

Possible suitability for microwave treatment from dimensional perspective

A study in the Netherlands on invasive SCC over 1989-2008 found that 73% of all newly diagnosed SCCs over this period were at Stage I defined as less than 2cm diameter [20]. Nodular BCC which appears as a pearly flesh nodule often with small dilated blood vessel s and rolled border accounts for 50-60% of diagnoses [21] [22]. BCC is slow growing attaining 1-2cm diameter after several years [23]. It would consequently appear that a substantial proportion of both SCC and BCC present as lesions less than 2cm in diameter, a proportion of which may be suitable candidates for microwave therapy. Further, In a study of prognosis of SCC [24] of 615 patients, it was found that only tumours with a thickness greater than 2mm were at a significant risk of metastasizing (0 metastases out of 207), 2.1-6mm was a low risk (12/318) and greater than 6mm presented a medium risk (14/90) – which would lend weight for early treatment and tumour dimensions more suitable for Microwave therapy.

Diagnosis of NMSC and depth of tumour

Diagnosis can be difficult and is usually achieved by biopsy or excision biopsy [21] the gold standard [19]. Excision biopsy would likely exclude the need for subsequent microwave treatment however new potential diagnostic methods are being investigated as outlined below.

Potential Non-Invasive *in-vivo* diagnostic methods

In order to filter suitable candidates for microwave treatment, it is necessary to determine the width and depth of the tumour so that adequate ablated margins can be achieved.

Ultrasound Imaging is a non-invasive and useful aid to BCC skin lesion dimensions and diagnosis [25] [26] and has been shown to provide detailed images of facial BCC and also provide information about blood flow in the tumour – which may be useful for assessing the likelihood of other malignancies [27] and any cooling effects relevant to the microwave treatment. Ultrasound has limitations in that it may not be able to detect epidermal or small (less than 0.1mm) lesions and also some SCCs (hyperkeratotic) [28]. Ultrasound depth of imaging is dependent on frequency with 7.5MHz used for >4mm depth and 13.5-100MHz for 0.015-3.0mm [28]. Combination of Ultrasound and Spatial frequency domain imaging has been used in a study of two patients, one with BCC (tumour thickness 1.79mm) and the other with SCC (tumour thickness (1.86mm) to map the tumours preoperatively non-invasively is described in [29].

Optical Coherence Tomography (OCT) A study of images of AK, BCC and normal skin by skilled observers demonstrated a 86-95% sensitivity and 81-98% specificity in the diagnosis of BCC lesions and 94-100% sensitivity with 28-76% specificity in differentiating NMSC from normal skin; OCT can image up to 2mm depth; the authors conclude that further technological and study improvements are likely to improve accuracy [30].

Reflectance Confocal Microscopy can image epidermis and papillary dermis to 0.2mm depth and lateral resolution of 1µm, is comparable to X30 histology but has a narrow field of view and equipment was reported as costing US$69,500 for a handheld device in a 2015 review [31] and has been evaluated for NMSC and melanoma in a study of 47 patients with reported sensitivity of 100% and specificity 69.2% [32].

Common Treatments for NMSC

Surgical Excision (SE)

SE achieves a clearance of 95% for well-defined BCCs of less than 2cm with a 4mm surgical margin; for well-defined SCCs of less than 2cm in non-complex areas a 4mm margin achieves 95% clearance – for complex areas (such as scalp, lips, nose) a 6mm margin is advised [21]**.** A guide to surgical management of NMSC noted that a surgical margin of 3mm for primary BCC lesions of less than 2cm resulted in a 2-5% recurrence rate [33]. For SCC, the European Dermatology Forum - European Association of Dermato-Oncology - European Organization for Research and Treatment of Cancer recommends at least 5mm margin on tumours less 2mm thick and 2.01- 6mm thick without high risk factors [34].

Mohs micrographic surgery (MMS)

MMS is described in a 1998 clinical review and summarised here as a surgical procedure that takes successive horizontal sections of the tumour site, each section is then histologically examined for tumour presence and surgical margins are re-evaluated with each section - this permits the minimization of surgical margins and so maximises the retention of viable tissue; BCC is the most common MMS procedure with 5 year cure rates of better than 99% and 96% for primary and recurrent BCC respectively; for SCC the 5 year cure rate was reported at 98% of patients without metastases and 16% with – the metastatic rate being estimated at 2-3%; MMS requires a high level of skilled surgical, histology and nursing staff [35]. A 2010 Argentinian study found 9 recurrences amongst 1957 patients with 2412 lesions after 5 years, representing a 0.37% rate; initial defect areas were 0.02-19.4cm^2 and final 0.06-42.37cm^2 [36]. However, a commentary [37] suggests that there is very little difference between the 5 year recurrence rates of MMS and Excision, that MMS is advantageous for recurrent and high risk NMCs but that possibly most NMSCs are not high risk, the commentary draws from a study [38] that gives 5 year recurrence rates of 4.9% for C&E, 3.5% for excision and 2.1% for MMS.

Curettage and Electrodessication (C&E)

This is the second most common BCC treatment after SE in the UK, with 5 year cure rates achieving 92.3% for selected BCCs and is indicated for use on low risk lesions on the trunk or extremities, although very high cure rates have been achieved with certain medium and high risk facial BCCs, excellent results have also been achieved with certain small primary SCCs [39].

Surgical excision depth

The above treatment descriptions cite surgical margins around the tumour, for the purposes of treating tumours with topical Microwave therapy the depth of the margin beneath the tumour would be of great importance. There is little information regarding excision depth but for SCCs the excision should extend into the hypodermis but not aponeurosos, periosteum and perichondrium unless these structures are involved according to [40], and as outlined above this could be 2mm for the epi and 5mm for the dermis assuming that the tumour has replaced the tissues involved rather than added to them.

A study of 1539 BCCs published in 2007 [41] that used excised tissue to assess margins concluded that 1-2mm histological margins peripherally and deeply may be adequate but notes that 2-3mm may be required to account for histological shrinkage ; the site distribution of lesions and (% with histological margin depth 0.1-4.9mm) comprised nose 22%(94%), temple 14%(96%), forehead

10%(90%), limbs 10%(60%),Peri-Orbital 9%(94%), cheek 9%(87%), Peri-Auricular 8%,(87%), trunk 5%(47%), scalp 5%(81%), neck/chin 5%(80%) and lip 3%(91%) these figures derived from table 5 and 6 from the study. Excluding the nose, peri-orbital, peri-auricular and lip this accounts for 57% of the BCCs studied and 46% of BCCs with histological depth margin of 0.1-4.9mm. Although this study does not include any follow up of recurrences, it does illustrate an example of clinical practice and suggests a histological margin of 1-2mm.

It would appear that a form of treatment that mimicked simple excision in terms of ablation and margins might be a suitable alternative, although this might mean that only a subset of smaller NMSC lesions would be suitable.

A selection of other current NMSC Treatments

Cryotherapy as described below is a good treatment for low risk BCCs but it relies heavily on expertise and selection of lesions for high risk BCCs [42]. **Radiotherapy (RT**) is not recommended for either SCC or BCC as first-line treatment if surgery is possible; Surgery results in better long term cosmesis compared to RT; RT can be used as adjuvant treatment for areas of high recurrence risk of SCC [43]. **Laser CO_2** [44] cites several small studies that demonstrate efficacy in superficial and nodular BCCs and some SCCs and conclude that there is an urgent need for prevention of NMSCs and alternative treatments especially in cases of multiple lesions and where surgery would be difficult. **Imiquimod** may be useful for adjuvant treatment of nodular BCC after C&E and has demonstrated efficacy for SCC *in situ* [44].

Cost of treating NMSC

For the purposes of comparing a new treatment cost to existing practices it is necessary to establish the cost to the clinicians of the existing practices, this information has been difficult to obtain. The estimated total cost to the NHS of treating all skin cancers in England was £108-£112m in 2008 using bottom-up and top-down methodology respectively and expected to rise to £180m by 2020; The estimated bottom-up cost of topical treatment (Imiquimod 30 treatments and 2 GP visits) £200, cryotherapy was £204 (based on 3 GP visits for wart treatment), £137 for C&E (outpatient procedure) and £885 for surgical excision (inpatient or day-case); The cost of NMSC per case was £889 with an estimated 73593 NMSC cases [45]. Although it is extremely difficult to predict with any accuracy the cost impact of a new treatment, there may be cost advantage in earlier treatments that result in previous inpatient procedures being performed in a primary care setting.

Actinic Keratosis (AK)

AKs (meaning sunlight-induced thickened skin) are typically rough scaly, warty or horny intra-epidermal papules or plaques up to 1-2 cm in diameter and are skin coloured, red or pigmented; Initially they may easier to feel (being sandpaper-like) rather than seen; they are caused by chronic exposure to sunlight (i.e. UV) and more common in the elderly with light skinned, blue-eyed, blond or red haired individuals most at risk [46] [47] [48]; Various images of AKs may be found on [46]. AKs may present as single or multiple lesions or an area of visible and non-visible lesions known as 'Field Cancerization [49]. Further, AK is thought to be a closely related genetically and a precursor to SCC [50]. Diagnosis may be made by appearance but biopsy may be required in certain circumstances such as doubt or suspicion of cancer [51].

Incidence

The Incidence of AK is high - in a study of 2061 Dutch participants spanning 2010 – 2012 (The Rotterdam Study, RS) [47], 610 participants (29.6%) were found to have 1-9 AK lesions and 163 (8%) with 10+ AK Lesions. Of these 610 participants with 1-9 AK lesions, the 1181 lesion sites were on the scalp (17.0%), face (40.3%), ears (7.96%), neck (0.85%) , back of hands (12.95%), forearms (8.98%) ,chest (7.62%) or elsewhere (4.32%). Note that the RS study breaks the AK count into 1-3 and 4-9, but these figures have been recalculated here to 1-9 AKs to illustrate the potential microwave treatment for up to 9 AK lesions; nevertheless, 433 participants exhibited 1-3 AKs in the RS study. A study in South Florida USA, estimates that the number of AK treatments in 1996 as 115,214.9 per 100,000 population and notes that the worldwide incidence appears to have increased substantially over the previous 20 years [52]. A 1996 UK South Wales study of 1034 subjects aged 60+ years found an incidence of AK of 23% [53]. A small (n=96) Queensland, Australia follow-up study found an initial AK incidence of 46% but with high rates of regression, recurrence and new AKs [54].

Regression and Progression of AK to SCC

There is much uncertainty regarding this topic, for example, a review [55] cites single AK general regression rates of 20-30% but up to 63%, and a corresponding recurrence of 15-53%, a progression of AK to SCC of between 0% and 0.075% per lesion year and up to 0.53% where there is a history of NMSC; the review concludes that the potential of AK to progress to SCC is beyond doubt, but difficult to estimate reliably. A 2013 study acknowledging that the individual conversion rates of AK to SCC is low, writes that 60-80% of SCCs arise from AK with a mean progression time of 2 years [56]. Due to this uncertainty, perhaps all AKs require treatment [57].

Current Treatments and Costs for AK

Topical Treatments

Photodynamic Therapy (PDT) uses a sensitizer such as Methyl Aminolevulinate (MAL) and an intense red light in one or two treatment sessions; **5-Fluorouracil(5-FU)** cream with a possible average clearance rate of 52% at 12 months, the patient is required to apply the cream once-daily for up to 4 weeks; **Imiquimod(IMIQ)** cream which has a possibly higher clearance rate of 70% at 12 months, two treatment regimes are reported: Once daily for 2 weeks repeated for a second 2 week period, Once-daily for 3 days/week for up to 16 weeks ; **Diclofenac(DICL)** gel which requires either a 60 or 90 day twice-daily application and a clearance rate of 33-50% [57]**; Ingenol Mebutate(IngMeb)**

gel which requires a once-daily 25cm^2 application for 2 or 3 consecutive days, a clearance rate of 42.2% for lesion on the face and scalp has been reported [58].

Cryosurgery

Cryosurgery is a procedure utilising extreme cold to affect tissue either to produce inflammation or destruction. The mechanism of cellular destruction is complex and due mainly to vascular stasis resulting in ischemia and direct cellular damage caused by ice crystal growth during the subsequent thaw; The cooling rate is not as important as the temperature reached where -20 to -30°C will cause substantial tissue damage but -40 to -50°C or even -60°C may be needed for destruction of tumour cells; The rate of thawing greatly affects the destruction, with slower thawing producing the most destructive effects due to increased ice crystal growth with damaging shearing forces produced by large crystals, and other solute effects; A second quicker freeze-thaw procedure produces a larger frozen volume with more extensive destruction – the interval between procedures may be significant; Different cell types show differing resistance to freezing, Keratinocytes surviving to -30°C, tumour cells may be highly resistant to freezing; Fibroblasts and collagen resist freezing and later provides a wound repair matrix [59]. The above illustrates the importance of accurate temperature control however with common methods such as a liquid nitrogen spray, it is difficult to control the ablated area according to a 2009 article which further states the requirement for better control of the ablated region while preserving adjacent regions [60].

Liquid nitrogen, a common cryogen is extremely dangerous [61] and requires special procedures for transport and use [62]. Accidents range in severity for example: a case (1982) of inhalation is described in [63], Inadequate gloves (1997) [64], a kitchen accident in 2009 [65], drink accident (2012) [66].

Cosmesis. Cryosurgery may lead to persistent changes in skin pigmentation consisting of hypopigmentation (with melanocytes present but melanosomes not present in keratinocytes) and peripheral hyperpigmentation [67] although a later study assumes hypopigmentation is due to loss of melanocytes [68]. In a comparison of keloid scar treatment between an Argon and a Liquid Nitrogen device, the results suggested that a degree of hypopigmentation was inevitable [69]. A 1979 study on rats comparing full thickness dermal burns from 100°C with freeze injuries at liquid nitrogen temperature concluded that the healed wound from freezing was larger due to lack of contraction in the healing process possibly due to comparative preservation of collagen and consequently there may be non-obvious scarring [70].

Cost of treating AK

For topical treatments, the costs of the medications vary with regime and dosage, at the time of writing NICE estimates these as: DICL 3% £76.60, 5-FU 5% £32.83, 0.5% 5FU/10% salicylic acid £76.60, IMIQ 5% £48.50 plus additional £97.00 if repeat treatment required, IMIQ 3.75% £113.00 or £226.00 depending on dosage, IngMeb £65.00 [71]. A lecture by the British Columbia College of Family Physicians gives the cost of methyl aminolevulinate PDT as Canadian $500 [72] . No information of the cost of cryotherapy for AK could be found but it may be similar to NMSC above (£204 for 3 GP sessions).

Prior Microwave work

Two pre-clinical trials November 2012 [73] and March 2013 [74] were conducted using the Microwave Skin Applicator (MSA) and equipment described in [75] .

Figure 2 and Figure 3 below were reproduced from [75]. Figure 2 shows the aperture view of the MSA and the resonant outer cavity and inner cavity with the microwave energy launcher may be seen. Figure 3 shows an example of the result of heating *in-vitro* porcine liver – in this case for 6s at 40W. The depth of tissue denaturing may be clearly seen along the subsequent incision, which appears to be parallel to the vertical axis and at the horizontal centre of the MSA.

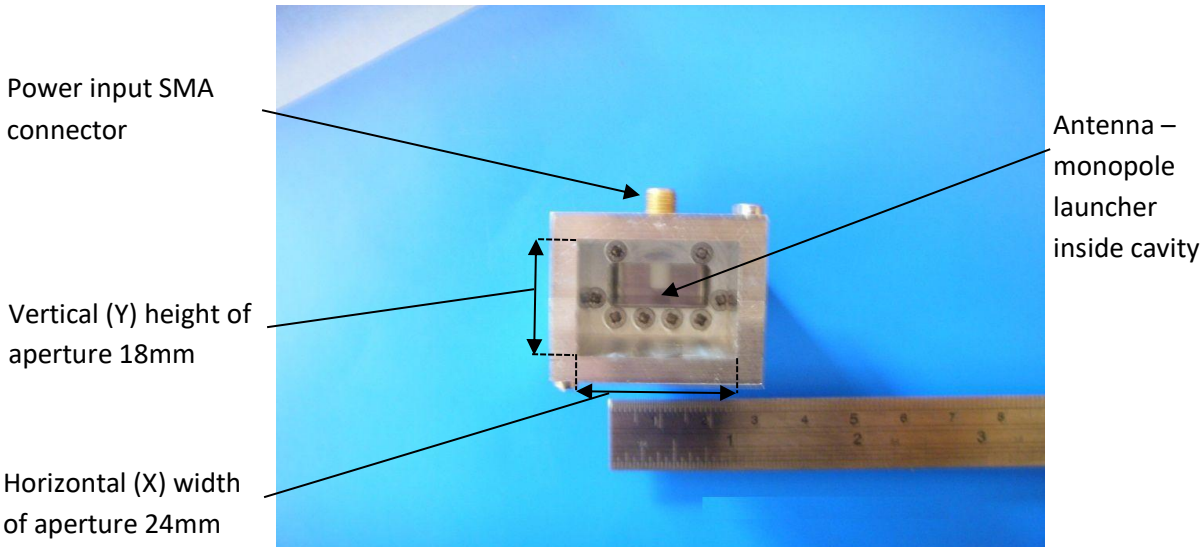


Figure 2 Aperture View of MSA. Note that camera and ruler position appear to distort the ruler measurement.

Reproduced with the permission of Professor C. Hancock on behalf of Bangor University.

.

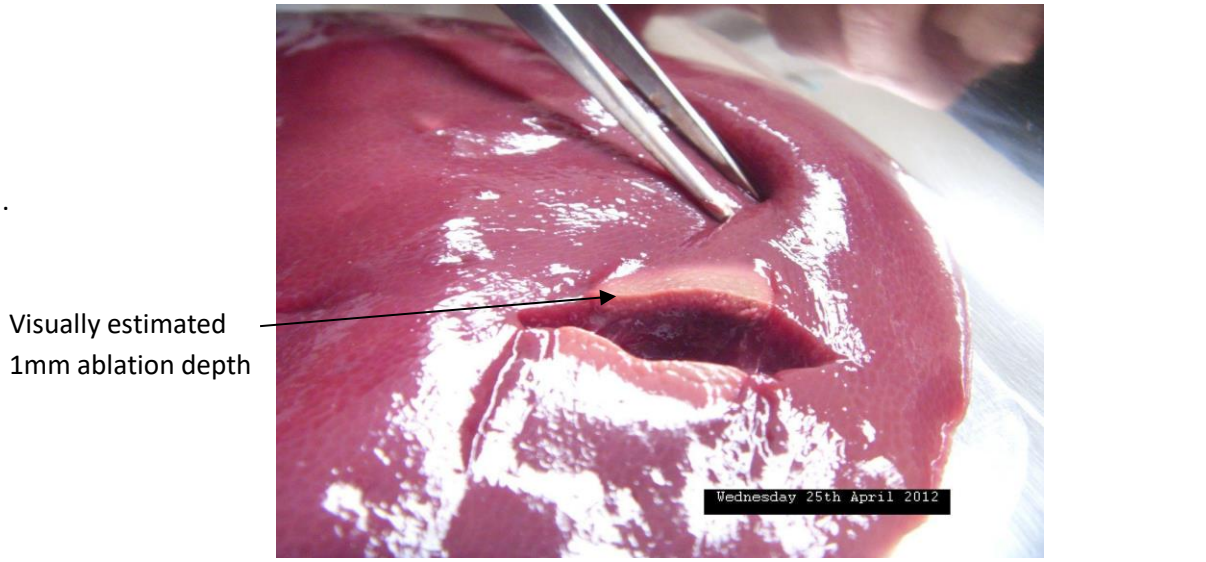


Figure 3 *in-vitro* Porcine liver Tissue heating profile resultant from 40W for 6s.

Reproduced with the permission of Professor C. Hancock on behalf of Bangor University.

The two pre-clinical trials evaluated the epi/papillary dermal/dermal effect of the MSA *in vivo* at various microwave powers and durations over 8 treatment sites per trial, 3 histological evaluations were subsequently taken over a period of 14 days for [73] and 28 days for [74]. With reference to [75], in brief, It may be seen that the microwave source and applicator system consists of an 14.5GHz oscillator and power amplifier providing 0-50W, a directional coupler and power meter to monitor the power to the MSA, and the MSA itself which is a closed (at one end) waveguide structure with monopole launcher designed to resonate when the open end is in full contact with human skin. The system provides approximately 29W at the distal end of the MSA when set to 40W, and 27W when set to 35W. This is shown below in Figure 4 where the heating profile is constant in the vertical (Y) direction and cosine[2] in the horizontal (X) direction with the maxima at the horizontal centre.

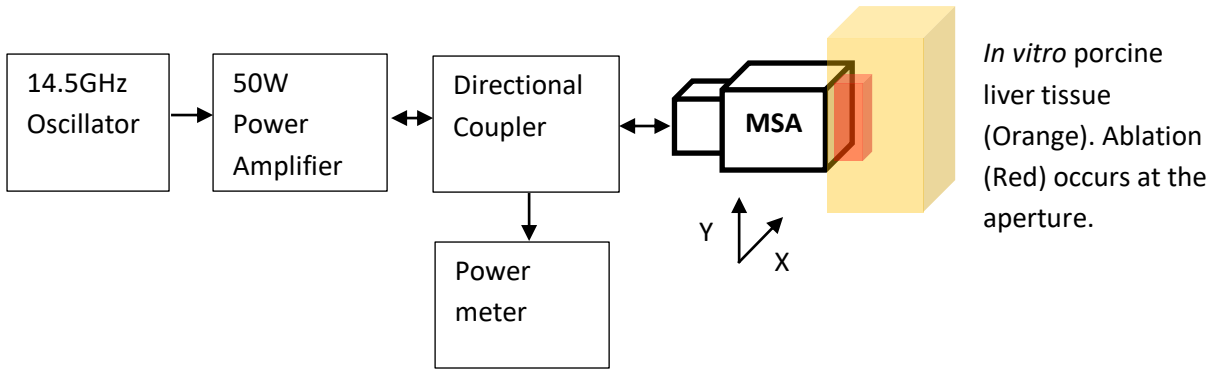


Figure 4 Simplified block digram of system. User interface and microcontroller not shown.

Relevant Main points of 2012 pre-clinical trial

The treatment sites had consistent energy deposition between them and individually demonstrated uniformity and 'fairly well defined lateral borders' with greater effect in the centre of the treated area. The histology reported a graduated ablation lessening from the surface, an unusually low amount of wound contraction[2] possibly due to continuity between ablated and non-ablated collagen.

Relevant Main points of 2013 pre-clinical trial

This trial produced consistent coagulation and ablation of epidermal, papillary dermal and dermal tissues which all repaired well. The histology report section outlines several dermatologic uses and specifically notes that it is essential that ablated/coagulated neoplasia epi and upper-papillary dermal areas are non-viable.

The reported ablation profile corresponds to the cosine shape E-Field of the waveguide across the aperture, but the vertical direction would be uniform and so it would be useful to compare these in future histology. In addition, this original system has some provision for simultaneous gaseous cooling of the skin surface for possible preservation of the epidermis or for greater ablation depth without overheating the surface.

[2] Contraction is an important phase of wound healing which occurs when myofibroblasts attach to the wound edges and contract resulting in reduction of the wound size [85].

Cost Feasibility of Microwave Applicator

As a very rough estimate, to heat up an area of human skin 1cmx1cmx5mm can be approximated as 0.5cc water (0.5gram; 1x1cm x 5mm deep) by 30°C (from 37°C to 67°C) assuming 50% efficiency and ignoring heat conduction to adjacent areas as a first approximation, would require 125.4 Watt-Seconds or Joules, so for 10W of available power, 12 seconds would be required i.e. a short treatment time. The microwave oscillator and power amplifier and associated control circuitry could be built into the applicator itself, so leading to greater power efficiency and lower cost as no interconnecting cable between power source and applicator would be required.

At the time of writing and based on several UK and US online component suppliers the Triquint/Qorvo TGF2023-2-02 GaN 12W HEMT DC-18Ghz power transistor [76] is available from [77] for US $42.02 in 300+ quantities – several may be required depending on amplifier topology - a quantity of 4 is assumed. Other components needed such as the oscillator can be built using a phase lock loop (e.g. Analog devices ADF41020 18GHz PLL synthesiser [78] US $11.53 100-499) and VCO (e.g. Analog Devices HMC736 [79] US $27.75 100-499) and 125MHz Reference crystal Oscillator (e.g. RS672-0918 [80] £1.93 25+). Further, a single chip microcontroller such as the PIC16F18325 (Microchip Inc. [81] US $1.03 100+), LCD display (e.g. RS758-8718 [82] £13.77 20+), IR temperature sensor (e.g. Smartec SMTIR9902SIL from Farnell [83] £10.27 500+), discrete components and PCB (estimated $50), PSU or battery (estimated $30), Enclosure (estimated $20).

In sum and using £1 = US$1.24009 [84] the component cost may be under US$350.00 perhaps implying a total manufactured cost of perhaps $550.00 which does not include the applicator itself or design costs, ongoing quality control, certification and any other manufacturing, testing, legal and regulatory costs. This would indicate that it may be possible to produce a complete system with user costings which when amortised over many treatments which could be comparable with current treatments, although it must be emphasised that this based on an extremely rough estimate and uncertainty as to existing costs.

Discussion and Aims of this work

The existing MSA produces an ablation that is well-defined but greater at the centre which is consistent with the microwave cosine-shaped field intensity, and, importantly for NMSC treatment, to reduce the risk of recurrence or metastases it is essential that apparently ablated tissue is not viable and that the ablation depth can be accurately controlled and maximised in order to obtain the required margins.

For AK, the microwave applicator would appear to provide an alternative to cryosurgery for individual or small number of lesions without the risks associated with the cryogen. For 'field cancerization' it would be necessary to produce a uniform ablation area so that a mosaic treatment approach could emulate the existing topical treatments. Should a mosaic approach be adopted, consideration would have to be given to ensure that the regrowth was healthy tissue. With reference to Table 1 above, at 14.5GHz 50% of available power is dissipated in the outer 0.7mm of skin implying that the epidermis may be well suited to this form of treatment as this layer may be heated rapidly with lesser reliance on heat conduction.

For SCC and BCC, the microwave applicator may be suitable where the tumour is in the early stages and where either biopsy and/or other non-invasive diagnostic techniques have confirmed a shallow invasion. For lesions less than 2cm and with a margin of 5mm an approximate circular area of 3cm diameter would ultimately be required but a mosaic approach could be adopted. In order to achieve destructive temperatures in the Hypodermis and using 2mm for thin skin and 5mm for thick (hairless) skin, a maximum of 5mm ablation depth would be required in addition to some extra margin. Consequently a reasonable aim for ablation depth might be at least 4-5mm with the recognition that thick skin may not be suitable for this technology. Also it should be kept in mind that there is an assumption that there is equivalence between surgical removal (or other types of established treatments) and this type of ablation of tissue in terms of prognosis, and that this is not addressed here.

So, for the possible future use of this technology in a clinical setting against a background of growing incidence of NMSC and AK this present work focusses on flattening the power distribution profile to achieve even ablation and to facilitate a mosaic treatment approach, maximising the heating effect and employing some direct feedback of the temperatures reached in order to achieve consistent reliable ablation with maximum penetration, minimising damage to healthy tissue, avoiding gross overheating or vaporisation of the ablated area and also to facilitate automatic dosing.

Works Cited

[1] U. S. Inan and A. S. Inan, Electromagnetic waves, Upper Saddle River, New Jersey: Prentice Hall, 2000.

[2] F. Sterzer, "Microwave Medical Devices," *IEEE Microwave magazine,* vol. 3, no. 1, pp. 65-70, 2002.

[3] A. Garcia-Miquel, S. Curto, N. Vidal, J. M. Lopez-Villegas and P. Prakash, "Compact microwave applicator for thermal therapy of breast cancer: Comparative assessment of arrays operating at 434 and 915 MHz," in *2016 10th European Conference on Antennas and Propagation (EuCAP)*, Davos, 2016.

[4] L. S. Poulou, E. Botsa, I. Thanou, P. D. Ziakas and L. Thanos, "Percutaneous microwave ablation vs radiofrequency ablation in the treatment of hepatocellular carcinoma," *World Journal of Hepatology,* vol. 7, no. 8, pp. 1054-1063, 2015.

[5] C. Hancock, "A New Wave in Electrosurgery - Therapeutic Applications of Microwave/RF Energy and Novel Antenna Structures," in *1EEE MTT-S International Microwave Workshop Series on RF and Wireless Technologies for Biomedical and Healthcare Applications (IMWS-Bio 2013)*, Singapore, 2013.

[6] T. Haque, K. M. Rahman, D. E. Thurston, J. Hadgraft and M. E. Lane, "Topical therapies for skin cancer and actinic keratosis," *European Journal of Pharmaceutical Sciences,* no. 77, pp. 279-289, 2015.

[7] A. Baroni, E. Buominno, V. De Gregorio, E. Ruocco, V. Ruocco and R. Wolf, "Structure and function of the epidermis related to barrier properties," *Clinics in Dermatology,* no. 30, pp. 257-262, 2012.

[8] O. Arda, N. Goksugur and Y. Tuzun, "Basic histological structure and functions of facial skin," *Clinics in Dermatology,* no. 32, pp. 3-13, 2014.

[9] "Blue Histology - Integumentary System," School of Anatomy and Human Biology - The University of Western Australia, 6 August 2009. [Online]. Available: www.lab.anhb.uwa.edu.au/mb140/corepages/integumentary/integum.htm. [Accessed 20 August 2016].

[10] D. Andreuccetti, R. Fossi and C. Petrucci, "An Internet resource for the calculation of the dielectric properties of body tissues in the frequency range 10 Hz - 100 GHz, Based on data published by C.Gabriel et al. in 1996.," IFAC-CNR, Florence (Italy), 1997. [Online]. Available: Website at http://niremf.ifac.cnr.it/tissprop/. [Accessed 08 9 2016].

[11] "THE EUROPEAN TABLE OF FREQUENCY ALLOCATIONS AND APPLICATIONS IN THE FREQUENCY RANGE 8.3 kHz to 3000 GHz (ECA TABLE)," 2016. [Online]. Available: www.erodocdb.dk/docs/doc98/official/pdf/ERCRep025.pdf. [Accessed 10 9 2016].

[12] N. P. Matylevitch, S. T. Schuschereba, J. R. Mata, G. R. Gilligan, D. F. Lawlor, C. W. Goodwin and P. D. Bowman, "Apoptosis and Accidental Cell Death in Cultured Human Keratinocytes after Thermal Injury," *American Journal of Pathology,* vol. 153, no. 2, pp. 567-577, 1998.

[13] E. V. Maytin, J. M. Wimberly and R. R. Anderson, "Thermotolerance and the Heat Shock Response in Normal Human Keratinocytes in Culture," *The Journal of Investigative Dermatology,* vol. 95, pp. 635-642, 1990.

[14] M. W. Dewhirst, B. L. Viglianti, M. Lora-Michiels, M. Hanson and P. J. Hoopes, "Basic principles of thermal dosimetry and thermal thresholds for tissue damage from hyperthermia," *International Journal of Hyperthermia, DOI:10.1080/0265673031000119006,* vol. 19, no. 3, pp. 267-294, 2003.

[15] *Telephone conversation with Chris Hancock, 30 Oct 2016.*

[16] D. L. Narayanan, R. N. Saladi and J. L. Fox, "Ultraviolet radiation and skin cancer," *International Journal of Dermatology,* vol. 49, pp. 978-986, 2010.

[17] T. I. Ibiebele, J. C. van der Pols, M. C. Hughes, G. C. Marks, G. M. Williams and A. C. Green, "Dietary pattern in association with squamous cell carcinoma of the skin: a prospective study," *American Journal of Clinical Nutrition,* no. 85, pp. 1401-1408, 2007.

[18] A. Lomas, J. Leonardi-Bee and F. Bath-Hextall, "A systematic review of worldwide incidence of nonmelanoma skin cancer," *British Journal of Dermatology,* vol. 166, pp. 1069-1080, 2012.

[19] M. C. F. Simões, J. J. S. Sousa and A. A. C. C. Pais, "Skin cancer and new treatment perspectives: A review," *Cancer Letters,* vol. 357, no. 1, pp. 8-42, 2015.

[20] L. M. Hollestein, E. de Vries and T. Nijsten, "Trends of cutaneous squamous cell carcinoma in the Netherlands: Increased incidence rates, but stable relative survival and mortality 1989–2008.," *European Journal of Cancer,* vol. 48, no. 13, pp. 2046-2053, 2012.

[21] S. Chummun and N. R. McLean, "The Management of malignant skin cancers," *Surgery(Oxford),* vol. 32, no. 9, pp. 485-490, 2014.

[22] K. E. Sharquie and A. A. Noaimi, "Basal cell carcinoma: Topical therapy versus surgical treatment," *Journal of the Saudi Society of Dermatology & Dermatoligic Surgery,* vol. 16, no. 2, pp. 41-51, 2012.

[23] J. F. McGuire, N. N. Ge and S. Dyson, "Nonmelanoma skin cancer of the head and neck I:histopathology and clinical behavior," *American Journal of Otolaryngology–Head and Neck Medicine and Surgery,* vol. 30, no. 2, pp. 121-133, 2009.

[24] K. D. Brantsch, C. Meisner, B. Schonfisch, B. Trilling, J. Wehner-Caroli, M. Rocken and H. Breuninger, "Analysis of risk factors determining prognosis of cutaneous squamous-cell carcinoma: a prospective study," *The Lancet,* vol. 9, no. 8, pp. 713-720, 2008.

[25] M. Nassiri-Kashani, B. Sadr, F. Fanian, K. Kamyab, P. Noormohammadpour, M. M. Shahshahani, H. Zartab, M.-M. Naghizadeh, M. Sarraf-Yazdy and A. Firooz, "Pre-Operative assessment of basal cell carcinoma dimensions using high frequency ultrasonography and its correlation with histopathology," *Skin Research and Technology,* vol. 19, no. 1, pp. 132-138, 2013.

[26] X. Wortsmann and J. Wortsman, "Clinical Usefulness of variable-frequency ultrasound in localised lesions of the skin," *Journal of the American Academy of Dermatology,* vol. 62, no. 2, pp. 247-256, 2010.

[27] X. Wortsmann, "Sonography of Facial Cutaneous Basal Cell Carcinoma," *Journal of Ultrasound in Medicine,* vol. 32, no. 4, pp. 567-572, 2013.

[28] A. Mandava, P. R. Ravuri and R. Konathan, "High-resolution ultrasound imaging of cutaneous lesions," *Ultrasonography/Muskuloskeletal Radiology,* vol. 23, no. 3, pp. 269-277, 2013.

[29] D. J. Rohrbach, D. Muffoletto, J. Huihui, R. Saager, K. Keymel, A. Paquette, J. Morgan, N. Zeitouni and U. Sunar, "Preoperative Mapping of Nonmelanoma Skin Cancer Using Spatial Frequency Domain and Ultrasound Imaging," *Academic Radiology,* vol. 21, no. 2, pp. 263-270, 2014.

[30] J. Olsen, L. Themstrup, N. De Carvalho, M. Mogensen, G. Pellacani and G. B. E. Jemec, "Diagnostic accuracy of optical coherence tomography in actinic keratosis and basal cell carcinoma.," *Photodiagnosis and Photodynamic Therapy DOI:http://dx.doi.org/10.1016/j.pdpdt.2016.08.004,* 8 2016.

[31] S. K. T. Que, N. Fraga-Braghiroli, J. M. Grant-Kels, H. S. Rabinovitz, M. Oliviero and A. Scope, "Through the looking glass: Basics and principles of reflectance confocal microscopy," *Journal of the American Academy of Dermatology,* vol. 73, no. 2, pp. 276-284, 2015.

[32] E. Cinotti, J. L. Perrot, N. Campolmi, B. Labeille, M. Espinasse, D. Grivet, G. Thuret, P. Gain, C. Douchet, F. Forest, M. Haouas and F. Cambazard, "The role of in vivo confocal microscopy in the diagnosis of eyelid margin tumors: 47 cases," *Journal of the American Academy of Dermatology,* vol. 71, no. 5, pp. 912-918, 2014.

[33] J. Divine, L. Stefaniwksy, R. Reddy, P. Padilla, T. Hagele, N. S. Patel and B. S. Cherpelis, "A comprehensive guide to the surgical management of nonmelanoma skin cancer," *Current Problems in Cancer,* vol. 39, no. 4, pp. 216-225, 2015.

[34] A. Stratigos, C. Garbe, C. Lebbe, J. Malvehy, V. del Marmol, H. Pehamberger, K. Peris, J. C. Becker, I. Zalaudek, P. Saiag, M. R. Middleton, L. Bastholt, A. Testori and J.-J. Grob, "Diagnosis and treatment of invasive squamous cell carcinoma of the skin: European consensus-based interdisciplinary guideline," *European Journal of Cancer,* vol. 51, pp. 1989-2007, 2015.

[35] D. L. Shriner, D. K. McCoy, D. J. Goldberg and R. F. Wagner jr, "Mohs micrographic surgery," *Journal of the American Academy of Dermatology,* vol. 39, pp. 79-97, 1998.

[36] G. Galimberti, A. Pontón Montaño, D. Ferrario, A. Kowalczuk and R. Galimberti, “Mohs Micrographic Surgery for the Treatment of Basal Cell Carcinoma,” *Actas Dermosifiliogr,* vol. 101, no. 10, pp. 853-857, 2010.

[37] R. S. Stern, “Cost Effectiveness of Mohs Micrographic Surgery,” *Journal of Investigative Dermatology,* vol. 133, pp. 1129-1131, 2013.

[38] M.-M. Chren, E. Linos, J. S. Torres, S. E. Stuart, R. Parvataneni and W. J. Boscardin, “Tumor Recurrence 5 Years after Treatment of Cutaneous Basal Cell Carcinoma and Squamous Cell Carcinoma,” *Journal of Investigative Dermatology,* vol. 133, pp. 1188-1196, 2013.

[39] V. Madan, J. T. Lear and R.-M. Szeimies, “Non-melanoma skin cancer,” *The Lancet,* vol. 375, no. 9715, pp. 673-685, 2010.

[40] J. J. Bonerandi, C. Beauvillain, L. Caquant, J. F. Chassagne, V. Chaussade, P. Clavere, C. Desouches, F. Garnier, J. L. Grolleau, M. Grossin, A. Jourdain, J. Y. Lemonnier, H. Maillard, N. Ortonne, E. Rio, E. Simon, J. F. Sei, J. J. Grob and L. Martin, “Guidelines for the diagnosis and treatment of cutaneous squamous cell carcinoma and precursor lesions,” *Journal of the European Academy of Dermatology and Venereology,* vol. 25, pp. 1-51, 2011.

[41] R. W. Griffiths, S. K. Suvarna and J. Stone, “Basal cell carcinoma histological clearance margins: an analysis of 1539 conventionally excised tumours. Wider still and deeper?,” *Journal of Plastic, Reconstructive & Aesthetic Surgery,* vol. 60, pp. 41-47, 2007.

[42] N. R. Telfer, G. B. Colver and C. A. Morton, “Guidelines for the management of basal cell carcinoma,” *British Journal of Dermatology,* vol. 159, pp. 35-48, 2008.

[43] M. Fort, S. Guet, L. Colson-Durand, C. Auzolle and Y. Belkacemi, “Role of radiation therapy in non-melanoma cancers, lymphomas and andsarcomas of the skin: Systematic review and best practice in 2016,” *Critical Reviews in Oncology/Hematology,* vol. 99, pp. 200-213, 2016.

[44] L. Brightman, M. Warycha, R. Anolik and R. Geronemus, “Do Lasers or Topicals Really Work for Nonmelanoma Skin Cancers?,” *Seminars in Cutaneous Medicine and Surgery,* vol. 30, no. 1, pp. 14-25, 2011.

[45] L. Vallejo-Torres, S. Morris, J. M. Kinge, V. Poirier and J. Verne, “Measuring current and future cost of skin cancer in England,” *Journal of Public Health,* vol. 36, no. 1, pp. 140-148, 2013.

[46] D. A. Oakley, “DermNet New Zealand, Actinic Keratotis,” December 2015. [Online]. Available: http://www.dermnetnz.org/topics/actinic-keratosis/. [Accessed 23 August 2016].

[47] S. C. Flohil, R. J. T. van der Leest, E. A. Dowlatshahi, A. Hofman, E. de Vries and T. Nijsten, “Prevalence of Actinic Keratosis and Its Risk Factors in the General Population: The Rotterdam Study,” *Journal of Investigative Dermatology,* vol. 133, pp. 1971-1978, 2013.

[48] British Association of Dermatologists, “ACTINIC KERATOSES - ALSO KNOWN AS SOLAR KERATOSES,” September 2013. [Online]. Available: http://www.bad.org.uk/shared/get-file.ashx?id=66&itemtype=document. [Accessed 24 August 2016].

[49] C. V. Asche, P. Zografos, J. M. Norlin, B. Urbanek, C. Mamay, C. Makin, S. Erntoft, C.-C. Chen, D. M. Hines and D. M. Siegel, “Evaluation of Resource Utilization and Treatment Patterns in Patients with Actinic Keratosis in the United States,” *Value in Health,* vol. 19, pp. 239-248, 2016.

[50] R. S. Padilla, S. Sebastian, Z. Jiang, I. Nindl and R. Larson, “Gene expression patterns of normal human skin, actinic keratosis, and squamous cell carcinoma: A spectrum of disease progression,” *Archives or Dermatology,* vol. 146, no. 3, pp. 288-293, 2010.

[51] “Actinic keratoses (solar keratoses),” NHS Choices, 30 7 2014. [Online]. Available: http://www.nhs.uk/conditions/solar-keratosis/pages/introduction.aspx. [Accessed 20 11 2016].

[52] M. S. Nestor and M. B. Zarraga, “The Incidence of Nonmelanoma Skin Cancers and Actinic Keratoses in South Florida,” *The Journal of Clinical and Aesthetic Dermatology ,* vol. 5, no. 4, pp. 20-24, 2012.

[53] I. Harvey, S. Frankel, R. Marks, D. Shalom and M. Nolan-Farrell, “Non-Melanoma skin cancer and solar keratoses. I. Methods and descriptive results of the South Wales skin cancer study,” *British Journal of Cancer,* vol. 74, pp. 1302-1307, 1996.

[54] C. Frost, G. Williams and A. Green, “High Incidence and Regression Rates for Solar Keratoses in a Queensland Community,” *The Journal of investigative Dermatology,* vol. 115, pp. 273-277, 2000.

[55] R. N. Werner, A. Sammain, R. Erdmann, V. Hartmann, E. Stockfleth and A. Nast, “The natural history of actinic keratosis: a systematic review,” *British Journal of Dermatology,* vol. 169, pp. 502-518, 2013.

[56] D. S. Rigel and L. F. Stein Gold, “The importance of early diagnosis and treatment of actinic keratosis,” *Journal of the American Academy of Dermatology,* vol. 68, no. 1, pp. S20-S27, 2013.

[57] A. R. Schmitt and J. S. Bordeaux, “Solar keratoses: Photodynamic therapy, cryotherapy, 5-fluorouracil, imiquimod, diclofenac, or what? Facts and controversies,” *Clinics in Dermatology,* vol. 31, pp. 712-717, 2013.

[58] M. Lebwohl, N. Swanson, L. L. Anderson, A. Melgaard, Z. Xu and B. Berman, “Ingenol Mebutate Gel for Actinic Keratosis,” *The New England Journal of medicine,* vol. 366, no. 11, pp. 1010-1019, 2012.

[59] A. A. Gage, J. M. Baust and J. G. Baust, “Experimental cryosurgery investigations in vivo,” *Cryobiology,* no. 59, pp. 229-243, 2009.

[60] H. Takeda, S. Maruyama, J. Okajima, S. Aiba and A. Komiya, “Development and estimation of a novel cryoprobe utilizing the Peltier effect for precise and safe cryosurgery,” *Cryobiology,* no. 59, pp. 275-284, 2009.

[61] J. Meulenbelt, “Nitrogen and nitrogen oxides,” *Medicine,* vol. 40, no. 3, p. 139, 2012.

[62] Royal Cornwall Hospitals NHS Trust, “Safe transport, storage, use and disposal of Liquid Nitrogen (Policy and Guidance),” January 2014. [Online]. Available: http://www.rcht.nhs.uk/DocumentsLibrary/RoyalCornwallHospitalsTrust/Estates/HealthAndSafety/HSP103PolicyOnTheControlOfLiquidNitrogen.pdf. [Accessed 10 July 2016].

[63] G. Rockswold and D. J. Buran, “Inhalation of Liquid Nitrogen Vapor,” *Annals of Emergency Medicine,* no. 11, pp. 553-555, 1982.

[64] P. Roblin, A. Richards and R. Cole, “Liquid Nitrogen injury: a case report,” *Burns,* vol. 23, no. 7/8, pp. 638-640, 1997.

[65] “German Heston Blumenthal blows off both hands in liquid nitrogen kitchen accident,” Telegraph, 14 July 2009. [Online]. Available: http://www.telegraph.co.uk/news/worldnews/europe/germany/5821433/German-Heston-Blumenthal-blows-off-both-hands-in-liquid-nitrogen-kitchen-accident.html. [Accessed 10 July 2016].

[66] A. Gladwell and D. Crookes, “Teenager's stomach removed after drinking cocktail,” BBC, 15 Oct 2012. [Online]. Available: http://www.bbc.co.uk/newsbeat/article/19866191/teenagers-stomach-removed-after-drinking-cocktail. [Accessed 10 July 2016].

[67] S. M. Burge, M. Bristol, P. R. Millard and R. P. R. Dawber, “Pigment Changes in Human Skin after Cryotherapy,” *Cryobiology,* no. 23, pp. 422-432, 1986.

[68] G. F. Graham and K. L. Barham, “Cryosurgery,” *Current Problems in Dermatology,* vol. 15, no. 6, pp. 229-250, 2003.

[69] M. C. E. Van Leeuwen, A. E. J. Bulstra, A. J. Van der Veen, W. B. Bloem, P. A. M. Van Leeuwen and F. B. Niessen, “Comparison of two devices for the treatment of keloid scars with the use of intralesional cryotherapy: An experimental study,” *Cryobiology,* no. 71, pp. 146-150, 2015.

[70] A. K. C. Li, H. P. Ehrlich, R. L. Trelstad, M. J. Koroly, M. E. Schattenkerk and R. A. Malt, “Differences in Healing of Skin Wounds Caused by Burn and Freeze Injuries,” *Annals of Surgery,* vol. 191, no. 2, pp. 244-8, 1980.

[71] “Actinic keratosis: ingenol mebutate gel,” NICE National Institute for Health and Care Excellence, March 2013. [Online]. Available: https://www.nice.org.uk/advice/esnm14/chapter/Context. [Accessed 09 09 2016].

[72] M. Wong, “Management of Actinic Keratosis,” Youtube, October 2013. [Online]. Available: https://www.youtube.com/watch?v=gWEVsh1fIhg. [Accessed 27 August 2016].

[73] P. Wall, “Pre-Clinical Study Report, A summary of pre-clinical study CH260912 and recommendation for future work, IHCR/12/11/01_CH_NPPCR_v1.0,” Isca Healthcare Research, 2012.

[74] P. Wall, “Pre-Clinical Study Report A summary of pre-clinical study CH121212 and recommendation for future work, IHCR/13/03/01_CH_NPPCR_v1.0,” Isca Healthcare Research, 2013.

[75] C. Hancock, “Skin Cancer Treatment System, Medical Microwaves Systems Research Group,” Bangor University, Bangor, 2012.

[76] “12 Watt Discrete Power GaN on SiC HEMT,” TriQuint/Qorvo, [Online]. Available: http://www.triquint.com/products/p/TGF2023-2-02. [Accessed 21 11 2016].

[77] Mouser Electronics, [Online]. Available: http://www2.mouser.com/ProductDetail/TriQuint-Qorvo/TGF2023-2-02/?qs=1vfYvNkoS0X8pE0PgA6nqQ==. [Accessed 21 11 2016].

[78] “ADF41020,” Analog Devices, [Online]. Available: http://www.analog.com/en/products/rf-microwave/pll-synth/integer-n-plls/adf41020.html#product-samplebuy. [Accessed 21 11 2016].

[79] “HMC736,” Analog Devices, [Online]. Available: http://www.analog.com/en/products/rf-microwave/vcos-plos/vcos-plos/hmc736.html#product-discussions. [Accessed 21 11 2016].

[80] “FXO-HC736R-125, Crystal Oscillator, 125 MHz, ±25ppm HCMOS 15pF, 4-Pin SMD, 7 x 5 x 1.4mm,” RS Components, [Online]. Available: http://uk.rs-online.com/web/p/crystal-oscillators/6720918/?sra=pstk. [Accessed 21 11 2016].

[81] “PIC16F18325,” Microchip Inc, [Online]. Available: http://www.microchip.com/wwwproducts/en/PIC16F18325. [Accessed 21 11 2016].

[82] “Electronic Assembly EA DOGL128L-6 Graphic Reflective LCD Monochrome Display Yellow-Green, LED Backlit, 128 x 64pixels,” RS Components, [Online]. Available: http://uk.rs-online.com/web/p/lcd-monochrome-displays/7588718/. [Accessed 21 11 2016].

[83] “SMARTEC SMTIR9902SIL INFRARED SENSOR + TEMP SENSOR + LENSE,” Farnell, [Online]. Available: http://uk.farnell.com/smartec/smtir9902sil/infrared-sensor-temp-sensor-lense/dp/2543404. [Accessed 21 11 2016].

[84] “XE Currency Converter: GBP to USD,” XE, 21 11 2016. [Online]. Available: www.xe.com. [Accessed 21 11 2016].

[85] Wikipedia, “Wound Healing,” [Online]. Available: https://en.wikipedia.org/wiki/Wound_healing. [Accessed 2 March 2017].

Chapter 2 Thermal Modelling of the Microwave Skin Applicator (MSA)

Contents

Introduction

This section aims to compare the simulated heating effect of microwave energy in a volume of tissue comprising Dermis, Fat and Muscle vs. various parameters such as Heating time, Surface cooling, Frequency of operation and Tolerance in total power. The main goals are to establish methods of providing accuracy to Ablation Depth, minimising damage to viable tissue and gaining an estimate of the dosage required.

In order to simplify the comparisons the maximum temperature permissible is set to 75°C with ablation assumed achieved at 60°C and the CEM_{43} [1] method is used to set a limit for heating damage.

Thermal Modelling Theory

Many studies such as [2] use a form of the Pennes Bioheat equation [3] to model and solve a thermal distribution problem. One form of this partial differential equation is given by [4] as:

$$\rho C_p \frac{\partial T}{\partial t} = K \nabla^2 T - \rho . SAR - b(T - T_{b)} \quad \{ 1 \}$$

With the convective boundary conditions:

$$K \frac{\partial T_s}{\partial n} = -h(T_s - T_{a)} \quad \{ 2 \}$$

Where ρ is the density in Kg/m^3, C_p is the Specific Heat in J/Kg °C, K is the thermal conductivity in W/m °C, SAR is the specific absorption rate in W/Kg, b is a constant describing power flow due to perfusion in W/m^3 °C, T = T(x,y,z,t) is the temperature within the modelled volume at time t (seconds), T_b is the blood temperature in °C, T_s is the skin surface temperature in °C, T_a is the ambient temperature in °C, and n = unit normal on the surface interface of the medium with air or heat conducting fluid with power transfer coefficient h in W/m^2 °C .

Heat from metabolic processes is assumed to be insignificant compared to incident power and has been omitted. The FDTD update equations are given in [4] but are worked through here to outline the process.

Using the FDTD method using discrete time step Δ_t and distance Δ_x and from [5] Pgs. 49 & 52:

1st Order forward difference in time:

$$\frac{\partial f_i^n}{\partial t} \simeq \frac{f_i^{n+1} - f_i^n}{\Delta_t} \quad \{ 3 \}$$

2nd Order centred difference in space:

$$\frac{\partial^2 f_i^n}{\partial x^2} \simeq \frac{f_{i+1}^n - 2f_i^n + f_{i-1}^n}{(\Delta_x)^2} \quad \{ 4 \}$$

Where the superscript and subscript of *f* refer to the time point and location in space respectively.

Substituting { 3 } into { 1 } where f =T(x,y,z,t) and { 4 } into { 1 } for x,y,z with $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ and spatial discretisation $\Delta_x = \Delta_y = \Delta_z = \Delta_L$

$$\rho C_p \frac{T_{x,y,z}^{n+1} - T_{x,y,z}^n}{\Delta_t} = K \left(\frac{T_{x+1,y,z}^n - 2T_{x,y,z}^n + T_{x-1,y,z}^n}{(\Delta_L)^2} + \frac{T_{x,y+1,z}^n - 2T_{x,y,z}^n + T_{x,y-1,z}^n}{(\Delta_L)^2} + \frac{T_{x,y,z+1}^n - 2T_{x,y,z}^n + T_{x,y,z-1}^n}{(\Delta_L)^2} \right) + \rho SAR - b(T_{x,y,z}^n - T_b)$$

And for $A_1 = \frac{\Delta_t}{C_p}$ $and\ A_2 = \frac{\Delta_t b}{\rho C_p}$ $and\ A_3 = \frac{K \Delta_t}{\rho C_p \Delta_L^2}$

The Update equation becomes:

$$T_{x,y,z}^{n+1} = T_{x,y,z}^{n} + A_3\left(T_{x+1,y,z}^{n} + T_{x-1,y,z}^{n} + T_{x,y+1,z}^{n} + T_{x,y-1,z}^{n} + T_{x,y,z+1}^{n} + T_{x,y,z-1}^{n} - 6T_{x,y,z}^{n}\right) + A_1 SAR - A_2\left(T_{x,y,z}^{n} - T_b\right) \quad \{5\}$$

For the Boundary condition, substituting { 3 } into { 2 } at time n+1 where the unit normal is along the z axis where Δ_L is the discretised distance step:

$$\frac{K(T_{x,y,z}^{n+1} - T_{x,y,z+1}^{n+1})}{\Delta_L} = = -h(T_{x,y,z}^{n+1} - T_a)$$

Rearranging, the boundary update equation becomes:

$$T_{x,y,z}^{n+1} = \frac{KT_{x,y,z+1}^{n+1} + T_a\Delta_L}{K + h\Delta_L} \quad \{6\}$$

Which means that the temperature at point z = 1 (surface) can be found from the temperature at z = 2 (1st cell inside the surface) at time n+1.

Update equations { 5 } and { 6 } are thus similar to the update equations from [4] from which the stability condition is used:

$$\Delta_t \leq \frac{2\rho C_p \Delta_L^2}{12K + b\Delta_L^2} \quad \{7\}$$

Thermal Model Physical representation

The thermal simulation is based on a cuboid volume as described below in Figure 5. This cuboid volume represents the tissue in front of the aperture of the MSA. The incident power is a plane wave i.e. constant in the X and Y directions for the central section (18 x 24mm) and zero outside of this central area and dependant on the permittivity and conductivity in the Z direction. The incident power profile in the X and Y directions is idealised, but the main simulation comparisons are taken from a midpoint in the central area. The range of modelled thicknesses are shown in Figure 5 and are the same as those described in the EM model below (Figure 6) .

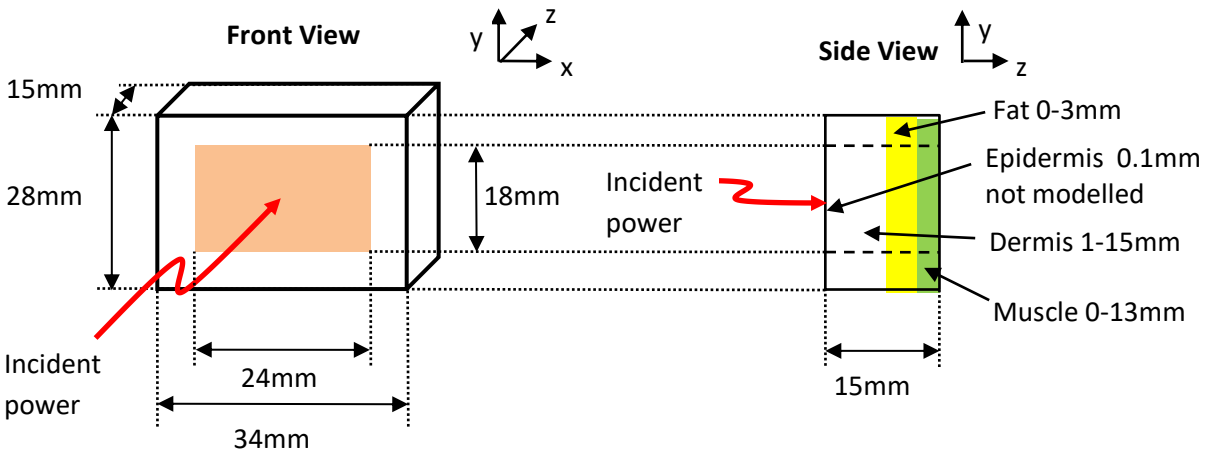


Figure 5 Thermal Model

Thermal Parameters

Several studies were reviewed [4] [2] [6], the parameters used were from [4] except for epidermis which was from [6] and forced convection [7] as follows in Table 2 and Table 3.

Table 2 Thermal Parameters for Epidermis, Dermis, Fat and Muscle

	Density ρ (Kg/m^3)	**Specific Heat C_p (J/Kg °C)**	**Conductivity K (W/m °C)**	**Perfusion b (W/m^3 °C)**
Epidermis	1200	3589	0.235	0
Dermis	1010	3500	0.5	8652
Fat	920	2300	0.22	816
Muscle	1040	3500	0.6	3488

Table 3 Convection heat loss parameters

Convection loss at skin surface	**Wind speed V = 0 m/s** [4]	**Wind Speed V >0 m/s** [7]
h_a (W/m^2 °C)	10.5**	$10.3 * V^{0.6}$

NOTE** On re-evaluation of thermal model the convective surface boundary heat loss was found to be entered as 10 W/m^2 °C instead of 10.5. This will make no difference to forced or convective simulations (based on wind speed > 0 m/s) but may make a slight difference to other simulations. It has been subsequently been corrected in 'Heat5.m' A comparison of the reference simulation of Figure 9 and the corrected simulation is given in Appendix 1D and shows no discernible differences.

Boundary and Initial Conditions

The entire volume is set to 37 °C at the start. All surfaces of the model are set to 37 °C except the front face (Orange section in Figure 5) which is modelled as either convective as described above (with variable wind speed and ambient temperature), or set to a specific temperature as follows later. The front face area surrounding the Orange section is set to 20 °C.

Determination of Damaged Tissue

The CEM43°C thermal dose method is used by the thermal model as described in [1] and [8] as follows:

$$\mathrm{CEM}_{43} = \mathrm{CEM43°C} = \sum_{i=1}^{n} t_i \ . R^{(43-T_i)} \ minutes$$

Where t_i is the time period i (in minutes) and T_i is the average temperature for that period. Using the model values given by [1], R = 0.25 for T<43°C and R = 0.5 for T>43°C. For example, tissue exposed to a single constant temperature of 50°C for 2 minutes would have a CEM_{43} value of $2 \times 0.5^{43-50} = 256$ minutes which is equivalent to 49°C for 4 minutes or 48°C for 8 minutes.

As described in [8] section 3.2 there may be extrapolation inaccuracies outside the 39-57°C range and consequently for the purpose of determining whether the tissue is damaged, the simulation assumes any temperature over the slightly more conservative 55°C to be damaged irrespective of the amount of time. To estimate limiting CEM_{43} values [1] pg.9 reports skin damage between 41-112 CEM_{43} minutes, chronic muscle damage above 80 CEM_{43} minutes and fat death at 15-60 CEM_{43} minutes, consequently as an approximation CEM_{43} values of ≥ 80minutes for skin or muscle and CEM_{43} values of ≥ 40 minutes for fat are determined to be damaged. Note that the CEM_{43} minutes are only accumulated during the simulation run so will be underestimated.

Implementation of the Thermal Model

The Thermal model is implemented as a Matlab function 'Heat5.m' and is given in Appendix 1G [3] page 143. The thermal model uses the above update equations as per [4] and is denoted in the code as 'Numerical_ Method = 0' and includes the update equations as per [9] for 'Numerical_Method = 1' although this has not been fully tested. The various controls and returned values (unless otherwise stated) are related to a line in the Z direction originating at the centre of the heated area. The main input variables for the function are as follows:

Program Label	**Meaning**
Delta_T	Time Step in seconds
Delta_Len	Distance Step in metres
Type_F	Frequency selection 0=14.5, 1=5.8, 2=24.125GHz
Input_Power	Total input power incident on heated area in Watts
Run_Time	Total run time in seconds, CEM43 accumulated for entire period
Epi_Thickness	Epidermis thickness in metres
Dermis_Thickness	Dermis thickness in metres
Fat_Thickness	Fat thickness in metres
Muscle_Thickness	Muscle thickness in metres
Use_Power_Profile	Boolean, set true to use EM power profile
Power_Loss_1Cyclep10	Vector of EM power profile in 0.1mm resolution
Power_Loss_1Cyclep25	Vector of EM power profile in 0.25mm resolution
Force_Aperture_Temp	Boolean, set true to force surface of heated area to Force_T
Force_T	Temperature in °C that aperture is forced to if enabled
Ambient_T	Ambient temperature of heated area surface in °C
Wind_Speed	Wind speed across heated area surface in m/s
Dial_A_Temp	Boolean, set true to turn off heating when following goal is met
Dial_A_Temp_Temp	Temperature goal
Dial_A_Temp_Depth	Depth from surface for temperature goal in metres
Auto_Regulation	Boolean, set true to regulate power by temperature else profiled
Auto_Runtime	Time in seconds to apply heating for auto regulation
Temp_Depth	Depth from surface to measure temperature for auto-regulation
Auto_Temp	Temperature to regulate to in °C
Heat_Profile	Vector of 1 second periods of power profile values 0-1

[3] **Disclaimer:** This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

The function returns the following main results:

Program Label	Meaning
Dial_A_Temp_Result	3D matrix of entire volume when Dial_A_Temp criteria met
Dial_A_Temp_Time	Time in seconds for Dial_A_temp criteria to be met
Dial_A_Temp_Tracking	Vector of maximum temperatures reached along centre line
SAR_Basic	3D matrix of SAR value for each volume element in W/Kg
Power_Basic	3D matrix of power for each element in W/m^3
Heat_Profile_S	Vector of On/Off power values for each Delta_T period when auto regulation used.
Time_Line	Matrix of centre temperatures for each Delta_T period
CEM43	Vector of accumulated CEM_{43} values along centre line
Found_Damage	Boolean, true if damaged tissue found along centre line
OK_Damage_Depth	Greatest depth in mm at which point $CEM_{43} \geq$ damage limit. Past this point tissue is undamaged according to CEM_{43} criteria.
Found_Ablation	Boolean set true if temperature ≥60 °C
Ablation_Depth	Greatest depth for which temperature ≥60 °C

Power input to the Thermal Model

As described below in the EM section, the power loss/dissipation profile is returned from the EM solver as a Matlab vector i.e. a 1 dimensional matrix. This provides a normalised power dissipation profile corresponding to the depth into the tissue. The ρSAR value has the units $Watts/m^3$ and each cell may take a different value although the power is idealised as constant in the X and Y directions. Consequently each 'slice' in the XY plane is apportioned power dissipation based on the corresponding value in the 1D power profile vector:

ρSAR in any cell in slice i = (Input Power/Slice Volume) * Power_Vector(i)/Sum(Power_Vector) W/m^3

Hence the loss/dissipation in any slice is determined by the total power and the 1D dissipation loss profile.

Limitations of the thermal model

Although the modeller is capable of simulating down to 0.1mm, the cell size has been set to 0.25mm due to the long simulation times required. Consequently the Epidermis has not been modelled. Further, all skin parameters (K, C_p ρ, ε, σ , b) have been assumed constant during the heating process, which is unlikely to be the case in practice. However, for comparison purposes of estimating dosage it is hoped that the simulations presented here will at least provide a first approximation. CEM_{43} minutes are only accumulated during the simulation run which (for relevant simulations) is 60 seconds. This means that there is likely to be an underestimate of the damage margin.

Modelling EM power dissipation in biological tissue

In order to estimate the heating effect of incident microwave energy on the skin and compare various frequencies and possible methods of achieving ablative and non-ablative temperatures, modelling of the profile of incident power and the consequent thermal response is required. FDTD update equations are given in [5] pg. 89 and the derivation is worked through here to demonstrate the process. The incident power profile is modelled from Maxwell's equations for a 1D plane wave as follows:

The EM field is modelled according to the FDTD method [10]. Starting with Maxwell's equations as described in [11] pg. 3:

$$\nabla \times E = -\frac{\partial B}{\partial t}$$

$$\nabla \times H = J + \frac{\partial D}{\partial t}$$

$$D = \varepsilon E$$

$$B = \mu H$$

$$J = \sigma E$$

For a 1 Dimensional plane wave travelling in the X direction with only E_y and H_z components, the above equations reduce to:

$$\frac{\partial E_y}{\partial x} = -\mu \frac{\partial H_z}{\partial t} \quad \{8\}$$

$$\frac{\partial H_z}{\partial x} = -\sigma E_y - \epsilon \frac{\partial E_y}{\partial t} \quad \{9\}$$

Using [5], pgs. 50 and 89 to find a value at a midpoint for function f(*i,n*) at *n+1/2* by averaging points *n+1* and *n*; and 2nd Order central difference approximation around point *n+1/2* by differencing points at *n* and *n+1* :

$$f_i^{n+1/2} \simeq \frac{f_i^{n+1} + f_i^n}{2} \quad \{10\}$$

$$\frac{\partial f_i^{n+1/2}}{\partial t} \simeq \frac{f_i^{n+1} - f_i^n}{\Delta_t} \quad \{11\}$$

Substituting Δ_x for Δ_t ,∂x for ∂t,and E_y for *f* and evaluating at time *n* around spacial point *i+½* in { 11 } :

$$\frac{\partial {E_y}^n_{i+1/2}}{\partial x} = \frac{{E_y}^n_{i+1} - {E_y}^n_i}{\Delta_x}$$

Substituting H_z for f and evaluating at spacial point *i+½* around time point *n* in { 11 }:

$$\frac{\partial {H_z}^n_{i+\frac{1}{2}}}{\partial t} = \frac{{H_z}^{n+\frac{1}{2}}_{i+\frac{1}{2}} - {H_z}^{n-\frac{1}{2}}_{i+\frac{1}{2}}}{\Delta_t}$$

and using results in { 8 } for interleaved leapfrog method detailed in [5]:

$$\frac{\partial {E_y}^n_{i+1/2}}{\partial x} = \frac{{E_y}^n_{i+1} - {E_y}^n_i}{\Delta_x} \quad = \quad -\mu \frac{\partial {H_z}^n_{i+\frac{1}{2}}}{\partial t} = -\mu \frac{{H_z}^{n+\frac{1}{2}}_{i+\frac{1}{2}} - {H_z}^{n-\frac{1}{2}}_{i+\frac{1}{2}}}{\Delta_t}$$

Yielding:
$$\frac{{E_y}^n_{i+1} - {E_y}^n_i}{\Delta_x} = -\mu \frac{{H_z}^{n+\frac{1}{2}}_{i+\frac{1}{2}} - {H_z}^{n-\frac{1}{2}}_{i+\frac{1}{2}}}{\Delta_t}$$

{ 12 }

Hence $\boldsymbol{{H_z}^{n+1/2}_{i+1/2} = {H_z}^{n-1/2}_{i+1/2} - \frac{\Delta_t}{\mu\Delta_x}({E_y}^n_{i+1} - {E_y}^n_i)}$

Similarly using { 10 }

$$\frac{\partial {H_z}^n_i}{\partial x} = \frac{{H_z}^n_{i+1/2} - {H_z}^n_{i-1/2}}{\Delta_x} = -\sigma E_y - \epsilon \frac{\partial E_y}{\partial t} = -\sigma\left(\frac{{E_y}^{n+1}_i + {E_y}^n_i}{2}\right) - \varepsilon\left(\frac{{E_y}^{n+1}_i - {E_y}^n_i}{\Delta_t}\right)$$

And hence $\boldsymbol{{E_y}^{n+1}_i = \left(\frac{2\varepsilon - \sigma\Delta_t}{2\varepsilon + \sigma\Delta_t}\right){E_y}^n_i - \frac{2\Delta_t}{\Delta_x(2\varepsilon + \sigma\Delta_t)}({H_z}^n_{i+1/2} - {H_z}^n_{i-1/2})}$ { 13 }

Solving for ${H_z}^{n+1/2}_{i+1/2}$ and ${E_y}^{n+1}_i$ Produces update equations given in [5] pg. 89, the corresponding Matlab [12] equations:

```
E_Y(2:end)   =    (SC1(2:end).* E_Y_P(2:end))  -  ( (SC2(2:end)./EM_Delta_Len) .* (H_Z(2:end) - H_Z(1:end-1)))
H_Z(1:end-1) =     H_Z_P(1:end-1)   -    ( (EM_Delta_T/(mu_0*EM_Delta_Len))  .* (E_Y(2:end) -  E_Y(1:end-1)))
```

Where:

```
SC1 =  ((2.*Space_Eps) -  (EM_Delta_T.*Space_Cond))./ ((2.*Space_Eps) + (EM_Delta_T.*Space_Cond))
SC2 =   (2*EM_Delta_T) ./ ((2.*Space_Eps) + (EM_Delta_T.*Space_Cond))
```

And where Space_Eps = absolute permittivity vector, Space_Cond = Conductivity vector, EM_Delta_T = time step, EM_Delta_Len = distance step, mu_0 = μ_0, E_Y_P and H_Z_P are values from previous iteration for H_Z and E_Y which are the current H_z and E_y vectors respectively.

EM Model

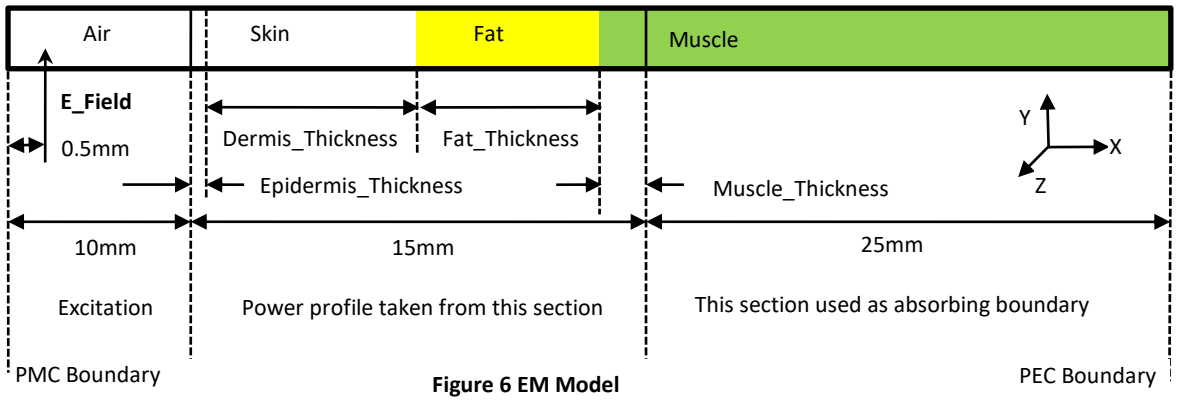


Figure 6 EM Model

Boundary value for x = 0 (i.e. Matlab $E_y(1)$ and $H_z(1)$) is Hz(1) = 0 which is a Perfect Magnetic Conductor which allows the Excitation (modelled as simple soft source [5] pg. 153) to be placed close by without cancellation from reflection from the x=0 boundary. Boundary at x=50mm is a Perfect Electrical Conductor which inverts and reflects the incoming wave. The absorbing boundary is 25mm of 'Muscle' which is approximately 6 skin depths at 5.8GHz and more at the higher frequencies.

The model is run for 38 cycles of sine wave excitation at 'E_Field' to achieve steady state, the next cycle is used to collect the E Field and Power dissipation profile which are returned to the calling script at normalised to maximum values. During the final cycle these and are collected as follows in Matlab and Based on power $P = \sigma E^2$ W/m^3 from [11] pg. 153. Note that here σ includes all losses such as ohmic losses and losses due to the electrical field producing ionic movement (imaginary part of complex permittivity ε''); The combining these losses is discussed in [13] pg. 47-48.

Matlab code to collect energy usage and calculate Steady state E field:

E_Loss2_1Cycle = E_Loss2_1Cycle+sum(((E_Y.^2) .* Space_Cond)*EM_Delta_T) Energy
RMS = RMS + E_Y.^2 = $\sum E^2$ Field Strength

And using [14] $\int Sin^2(ax)dx = \frac{x}{2} - \frac{1}{4a}\sin(2ax) + C$

And substituting $a = 2\pi f$ $x = n\Delta t$ and evaluating over one period we get the Matlab expression

EE2=(RMS.*(2*f*EM_Delta_T)).^0.5

Which gives the steady state E Field vector which is used to generate the E field and Power profile.

Electrical Tissue parameters

The IFAC website [15] was used to obtain the tissue parameters where the IFAC 'SkinDry' was used for the Dermis. These parameters are shown below in Table 4 .

Table 4 Electrical Tissue Parameters

Frequency GHz	Relative Permittivity dimensionless			Conductivity S/m			Wavelength m		
	Dermis	Fat	Muscle	Dermis	Fat	Muscle	Dermis	Fat	Muscle
5.8	35.114	4.95	48.485	3.717	0.293	4.9615	0.0086	0.0231	0.0073
14.5	26.88	4.29	36.954	13.27	0.9	17.194	0.00216	0.0099	0.0033
24.125	18.909	3.83	27.294	22.944	1.496	29.577	0.00264	0.0062	0.0022

Note that other frequencies are available in the simulation program, but only those used for the results presented here are given above. The wavelength parameter is used to check that there are at least 10 cells per wavelength.

Implementation of the EM Model

The EM model is implemented as a Matlab function 'Planewave1dv3.m' and is given in Appendix 1F[4] 139. The main input variables for the function are as follows:

Type_F	Frequency selection 0=14.5, 1=5.8, 2=24.125GHz
Epi_Thickness	Epidermis thickness in metres
Dermis_Thickness	Dermis thickness in metres
Fat_Thickness	Fat thickness in metres
Muscle_Thickness	Muscle thickness in metres

With the main returned results:

Power_Loss_1Cyclep10	Vector of EM power profile in 0.1mm resolution
Power_Loss_1Cyclep25	Vector of EM power profile in 0.25mm resolution

[4] **Disclaimer:** This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

Running the solvers

A typical simulation consists of initially setting the variables such as 'Dermis_Thickness' and the frequency 'F_Type', then running 'Planewave1dv3' and using the resultant power loss profile which is passed to 'Heat5'. The Matlab script 'Heat5run.m' is given in Appendix 1E[5] which contains the code for the various simulations that follow. All simulations were run on Matlab 2012A student edition [12] on an Intel I7 6700 3.4GHz 8Gb Windows 10 [16] desktop computer.

Comparison of simulated to analytical results

A sample simulation was performed at 14.5GHz with Skin thickness = 2mm, Fat Thickness = 2mm and Muscle thickness = 11mm. The analytic results were obtained using the phasor formulae for incident and reflected EM waves given in [13] pgs. 141-144 for normal plane wave propagation in 3 media and hence two boundaries (Skin-Fat and Fat-Muscle) together with the given reflection and transmission coefficients. The analytic value of $|E|$ is derived from the magnitude of the phasors that describe the forward and reflected waves and the normalised power dissipation derived from $\sigma|E|^2$. The Skin-Air boundary is not specifically taken into account as reflected waves at the Skin-Air boundary are added to the forward wave, the amplitude of which is normalised. No reflected waves are taken into account in the muscle as this is several skin depths. The results are shown below in Figure 7 showing a close correlation between the simulated and analytic results.

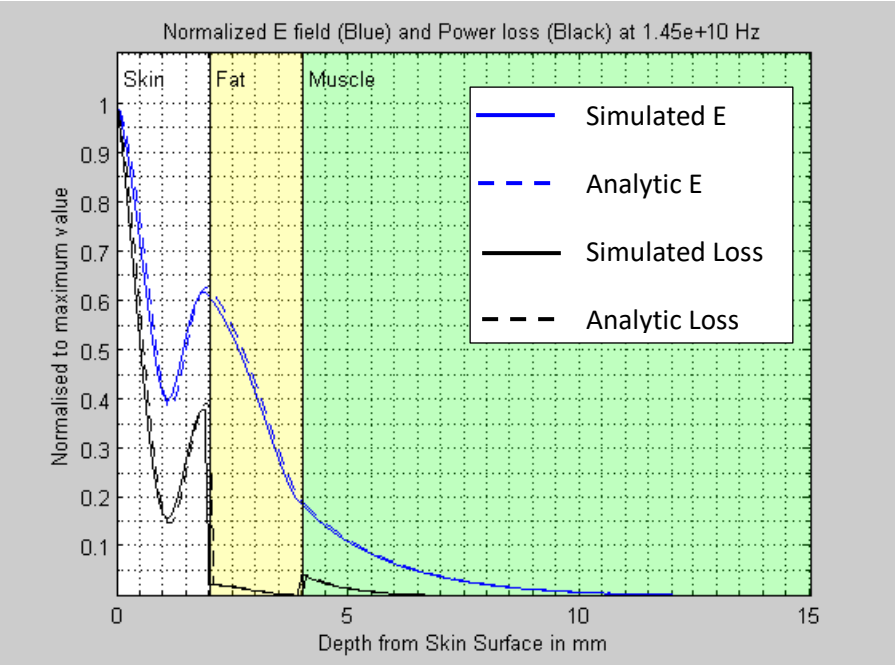


Figure 7 Comparison on Simulated and Analytic results

[5] **Disclaimer:** This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

Simulation of Ablation and Damage for 30 second heating cycle with Temperature Regulation

As an introduction into the various heat modelling runs and to provide a base for future comparisons, the simulation was run with the settings: 30 second heating cycle, 60 second run time, 40W power, Power regulation by temperature control which turns the power off and on to keep a specific point or all points beneath the skin surface below a predefined temperature – in this instance 75°C at 0mm depth and Dermis to 14.75mm depth. Although the Dermis is usually 3-5mm, a 14.75mm thick homogeneous Dermis is used for simple comparisons between frequencies and methods of increasing Ablation depth. Graphs are shown for 14.5GHz only.

Results

Figure 8 shows the normalised 1D E field and EM power profile output by the EM modeller which is used to create a 3D power profile constant in X and Y directions for the aperture dimensions and varies in the Z direction (depth beneath the skin surface). The nearest point to the E field skin depth is shown as 2.2mm at 36% which agrees closely to the IFAC [15] 'SkinDry' value of 2.16mm at 36.79%.

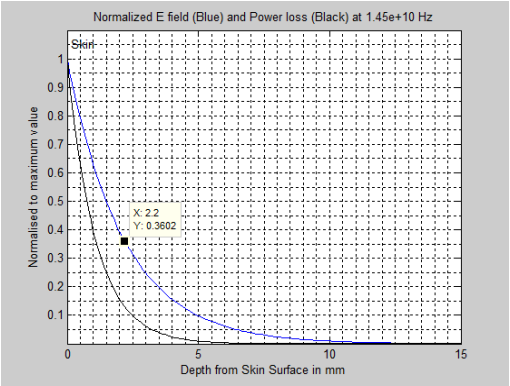


Figure 8 E-Field and Power Loss profile at 14.5GHz

Figure 9 shows the heating profile at tissue X and Y centre, the top Blue trace demonstrates the effect of temperature regulation where the line corresponding to '0mm depth' i.e. the skin surface rapidly reaches 75°C and then stays roughly constant as the power is cycled. After the heating cycle of 30 seconds is finished, the temperatures slowly decay for the remainder of the run.

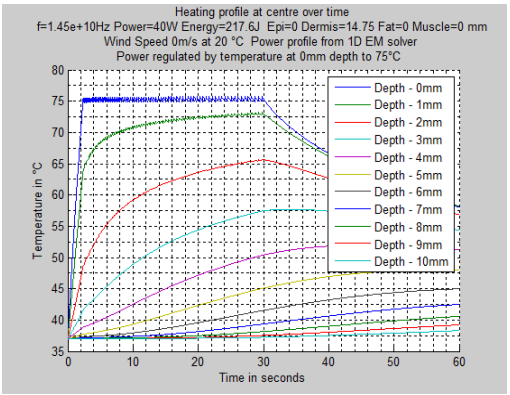


Figure 9 Heating Profile at various depths vs time

Figure 10 shows the Maximum temperatures reached (Blue) and the CEM_{43} minutes (Black) at the X and Y centre. Any point that has reached 60°C for any length of time is assumed to be ablated which means that the ablation will occur from the surface to a depth of 2.5mm. For skin, the maximum CEM_{43} value is 80 minutes and the nearest point below this value is shown as 43.76 minutes at 4.5mm depth – to the left of this all points exceed 80 CEM_{43} minutes and so are assumed damaged (or ablated if over 60°C) .

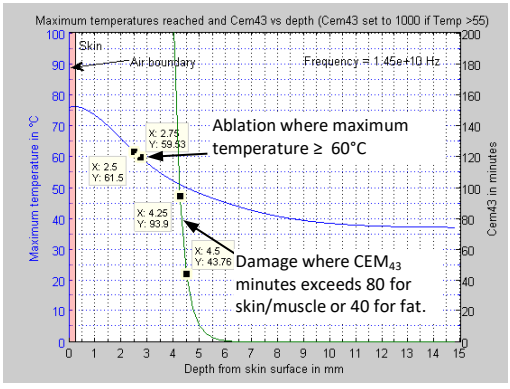


Figure 10 Maximum temperatures reached and CEM_{43}

Summary

At 14.5GHz a temperature regulated heating cycle produces 2.5mm Ablation with damaged margin up to 4.5mm. Similarly 5.8GHz results in 4.75mm Ablation and 7.75mm damaged margin; and at 24.125GHZ 2mm Ablation and 3.75mm damaged margin.

Effect of Frequency on Ablation Time and Damage depth for ablations up to 3mm in homogeneous tissue

The aim of these simulations is to assess how long it takes to reach a particular Ablation depth. The simulations were performed by using the heat modeller facility 'Dial_A_Temp' that turns off power for the remainder of the simulation after a preset temperature of 60 °C at a particular depth is reached. For the remainder of the run, the simulation continues without power and the CEM_{43} minutes accumulated. In addition, the total run time of the simulation is 60 seconds, Temperature regulation at the surface to 75°C, IFAC 'SkinDry' tissue type and maximum heating time of 30 seconds.

Results

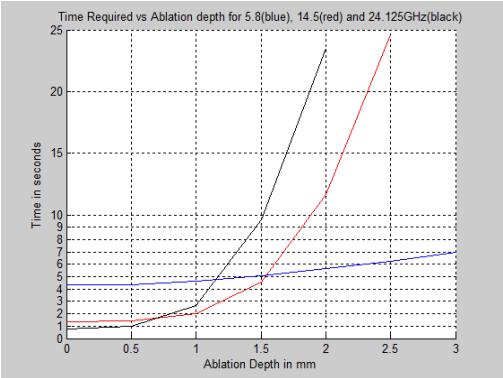


Figure 11 Heating time vs Ablation depth

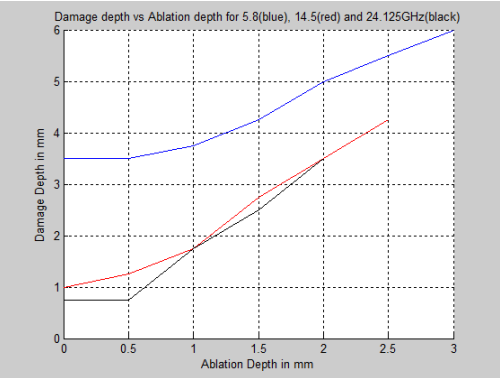


Figure 12 Damage Depth vs Ablation Depth

The Ablation Depths for 5.8, 14.5 and 24.124GHz are shown in Figure 11 for a maximum heating time of 30 seconds in homogenous 'skin' tissue. As would be expected 5.8GHz has a greater penetration depth while 24.125GHz and 14.5GHz reach 2mm and 2.5mm respectively. In addition, 24.125 and 14.5GHz reach the ablation temperature faster than 5.8GHz for ablation depths of up to 1 and 1.5mm respectively but take appreciably longer for greater depths.

Damage depth is defined as the depth where the tissue is not hot enough to be ablated but the CEM_{43} minutes exceed 80 minutes for skin/muscle or 40 minutes for fat (note that this simulation is for skin only). The damaged tissue depth is very similar between 24.125 and 14.5GHz but ranges between 2.5-1.5mm deeper for 5.8GHz (Figure 12).

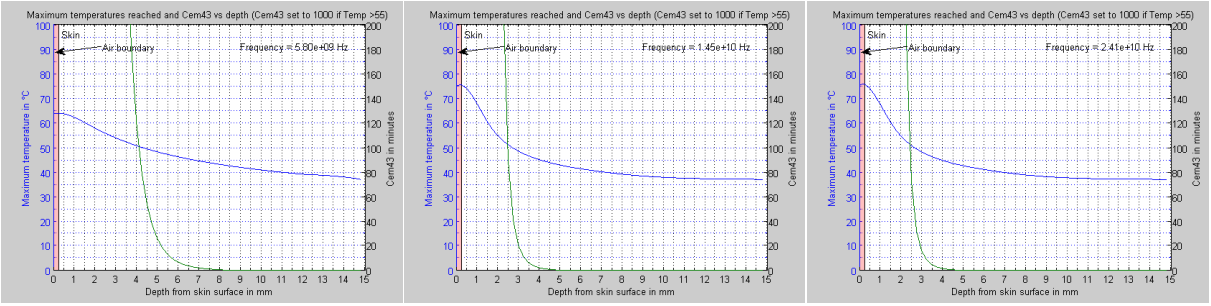


Figure 13 Comparison of Maximum Temperatures reached and CEM43 for 1.5mm Ablation depth for 5.8GHz (left) , 14.5GHz(middle) and 24.125GHz(right)

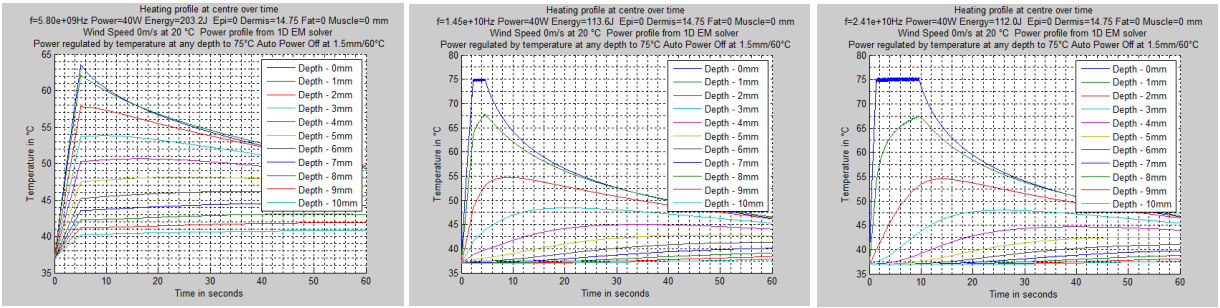


Figure 14 Heating profiles for 1.5mm Ablation Depth for 5.8GHz (left) , 14.5GHz (middle) and 24.125GHz(right)

Figure 13 and Figure 14 show a comparison between frequencies for 1.5mm Ablation where 5.8GHz demonstrates a flatter maximum temperature heating profile and greater damage margin than 14.5GHz and 24.125GHz which appear almost identical at this ablation depth.

The energy used at 5.8GHz (203J) demonstrates approximately a 1.8x increase compared to 14.5GHz (113J) and 24.125GHz (112J). This difference in energy is due to the greater skin depth at 5.8GHz of 8.6mm compared to 2.1mm at 14.5GHz and 1.1mm at 24.125GHz resulting in more power being dissipated at greater depths in the tissue. Consequently by the time the ablation goal of 60°C at 1.5mm is met, tissue deeper than 1.5mm at 5.8GHz has absorbed more energy and has higher temperatures compared to 14.5GHz and 24.125GHz. In addition, note that the available power is in effect reduced by temperature regulation for 14.5GHz and 24.125GHz but not for 5.8GHz which does not reach the surface temperature limit of 75°C.

Further, the role of heat conduction is greatest for 24.125GHz which is evidenced by the requirement of 9 seconds to reach the ablation depth goal compared to 14.5GHz (4.4 seconds) and 5.8GHz (5 seconds).

Effect of Incident power variation on Ablation depth for Regulated Surface Temperature and Fixed Power profile at 14.5GHz

Real-time feedback of surface temperatures has many advantages such as the ability to ensure that the maximum allowable temperature is not exceeded; the temperature rise matches power dissipation so providing a means of checking that the power source is delivering the expected amount of power; the skin tissue is at normal body temperature at the beginning of treatment (and providing a small amount of power to bring it up to 37°C if necessary) and also that the underlying skin structures are as expected (Later results show that fat under the surface can cause greater power dissipation near the dermis/fat boundary due to reflection at the boundary) . In order to achieve this temperature regulation, the MSA may be modified to include a sensor such as a thermal camera as shown in Figure 15. This camera would be located so that it can view the entire irradiated skin surface.

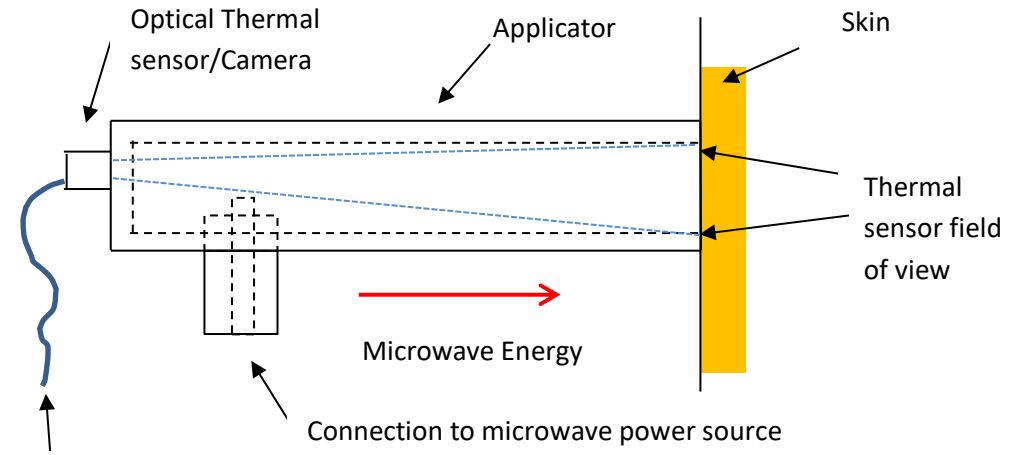


Figure 15 Modified MSA side view with Thermal Camera

For this investigation the EM modeller was run to provide an EM power profile, which was subsequently used by the Heat Modeller which was set to attain a 75°C temperature for 30 seconds followed by a 30 second cool down period with homogenous 'skin' to 15mm deep. The Thermal modeller returns the on/off power vector recorded during the run which was then used to calculate a fixed input power profile for the Thermal modeller based on 1 second time windows and with variable power levels (Figure 16, right). The Thermal Modeller was then run for -0,-10,-20 and -30% power levels for both regulated and fixed power profiles.

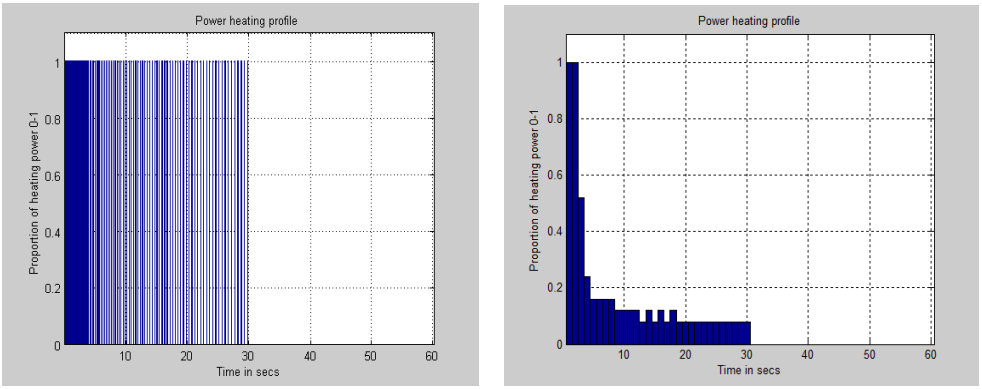


Figure 16 Power Profile for regulated temperature (left) and resultant Fixed Power Profile (right)

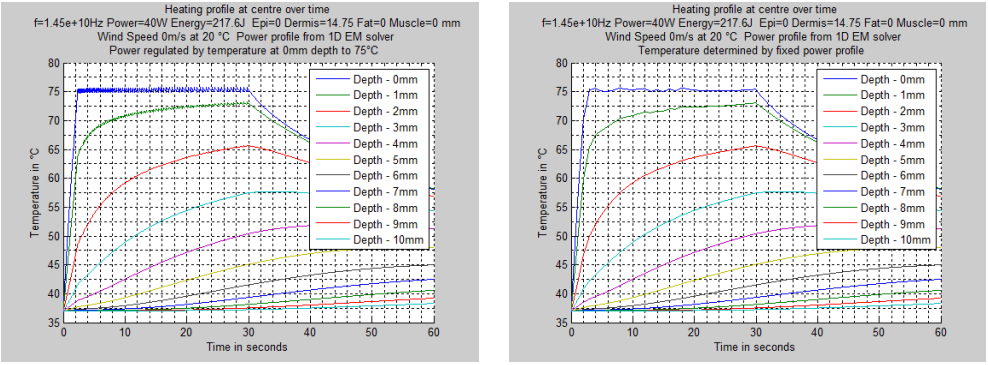


Figure 17 Temperature vs Time and Depth for Regulated Temperature (left) and Fixed Power Profile (right) at 40W

Figure 17 Shows the Temperature Regulated heating profile (left) and the heating profile generated from the calculated Fixed Power Profile (right). This shows that the Fixed Power Profile produces the same results as the automatic Temperature Regulated simulation when the input power levels are the same at 40W. A Fixed Power Profile might be used to obtain a maximum temperature of 75°C at the skin surface where no thermal feedback is available, but this approach is sensitive to reductions in power levels as shown below.

Results

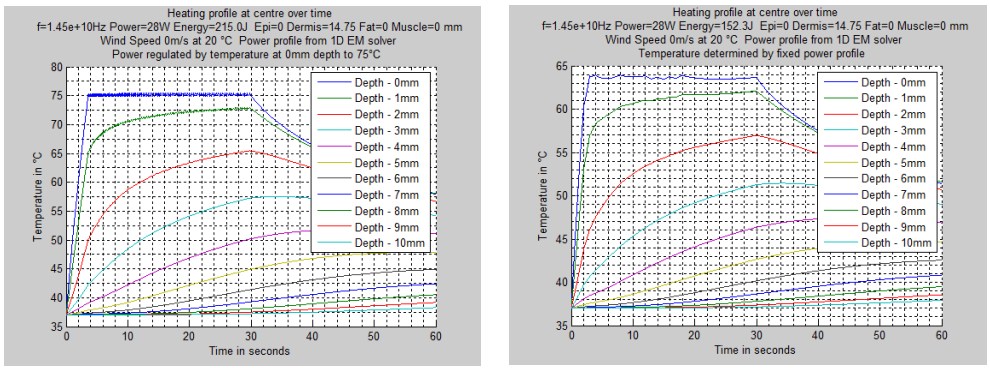


Figure 18 Temperature vs Time and Depth for Regulated Temperature (left) and Fixed Profile (right) at 28W

The results shown by Figure 18 compare the temperature profiles for 28W (i.e. -30% power drop) between Temperature Regulated (left) and Fixed Power Profile (right). The 28W Temperature Regulated heating profile is very similar to that at 40W (Figure 17, left) and the energy usage is almost the same at 215J vs. 217.6J. The fixed power profile demonstrates a reduction in temperature at all depths due to the 30% drop in energy dissipation 152.3J vs 217.6J – the heating has essentially been scaled down by 30% over the same heating period.

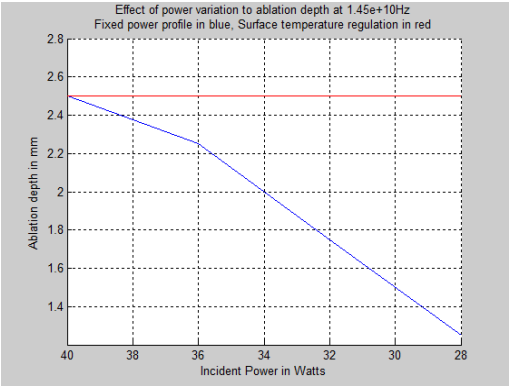


Figure 19 Comparison of Regulated Temperature (red) with Fixed Power Profile (blue) for 0-30% power drop

The Fixed Power Profile (Figure 19, blue) shows a reduction in ablation depth from 2.5mm to 1.25mm while the Temperature Regulated Profile demonstrates unchanged ablation depth at 2.5mm over 40-28W incident power with 2.3 seconds and 3.54 seconds taken to reach the 75°C surface temperature limit respectively. By maintaining a fixed surface temperature, the Temperature Regulated Profile compensates for a drop in available power by increasing the ratio of power-on to power-off time after the preset surface temperature goal of 75°C is reached while the Fixed Power Profile has no means of compensation and so results is a drop in temperatures.

Thus Temperature regulation provides compensation for power variations in this instance of homogeneous and known skin parameters.

Effect of difference in Dermis Thermal parameters on Ablation depth for Regulated Surface Temperature and Fixed Power profile at 14.5GHz

There are differences in Dermal thermal parameters between various studies. This simulation intends to estimate the effect of a fixed power profile that was generated with one set of Dermis thermal skin parameters when it is used on a different set of thermal parameters and compare the result with a temperature regulated simulation.

Figure 20 shows the heating profile from which the reference power profile in Figure 21 is derived. The Power Profile is basically the coefficient of the total power input (40Watts) required to produce the heating profile of Figure 20. As may be seen on the graph, the total energy utilised is 217.6J

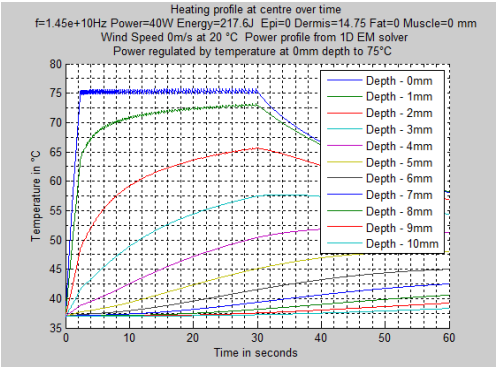


Figure 20 Reference Heating simulation

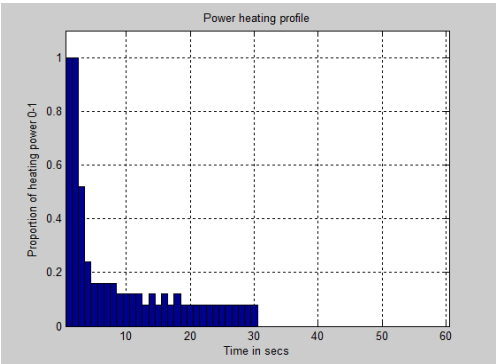


Figure 21 Reference Power Profile

Figure 22 shows the heating profile that results from a variation in Skin Thermal parameters which are now based on [6] i.e. K = 0.445 W/m K, C_p= 3300 J/Kg K, ρ = 1200 Kg/M^3 instead of [4] K = 0.5 W/m K, C_p= 3500 J/Kg K, ρ = 1010 Kg/M^3. As before, the total energy required is 217.6J.

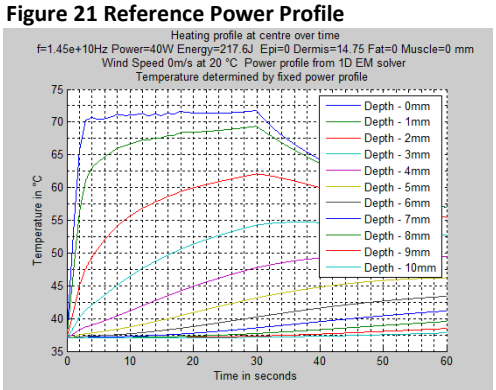


Figure 22 Fixed power heating profile with varied Skin thermal parameters

Figure 23 shows the heating profile with the alternate Skin thermal parameters but with temperature regulation. Note that the total energy has increased to 240.8J

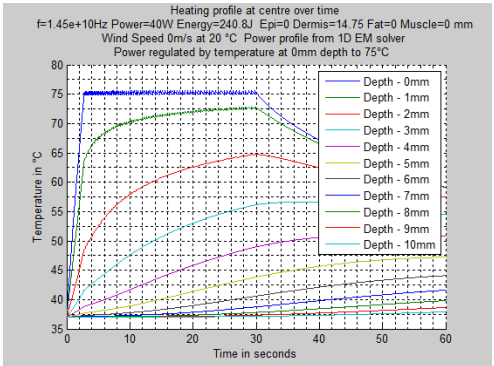


Figure 23 Temperature regulated heating profile with varied Skin thermal parameters

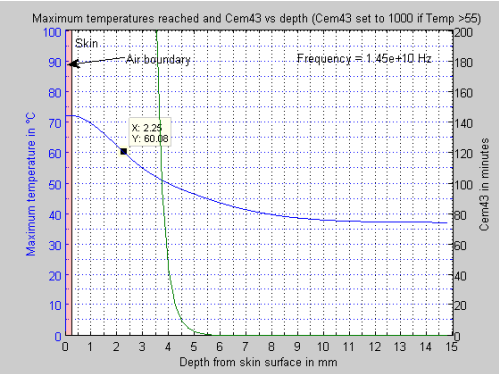


Figure 24 Maximum temperatures and CEM43 for Fixed Power Profile

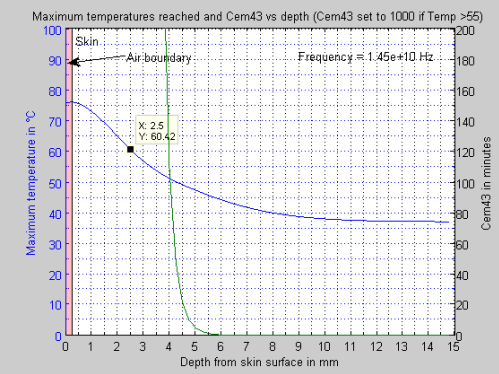


Figure 25 Maximum temperatures and CEM43 for Temperature Regulation

Figure 24 and Figure 25 show the resultant maximum temperatures and CEM_{43} for fixed power and temperature regulation respectively. Compared to the reference simulation of Figure 10 Page 40, the Fixed Power Profile demonstrates a reduced ablation depth from 2.5mm to 2.25mm and reduced damaged margin from 4.5mm to 4.25mm while the Temperature Regulated system shows unchanged ablation depth at 2.5mm and reduced damage margin from 4.5mm from 4.25mm .

The Temperature Regulation appears to compensate for these changes in thermal parameters.

Effects of Surface Convective Cooling at 5.8GHz, 14.5GHz and 24.125GHz

The purpose of this work is to investigate the effect of surface cooling on the maximum ablation depth and the depth of potentially damaged tissue. Ablation depth is assumed to be the depth at which 60°C or above is reached and the damaged depth is that at which the CEM_{43} is more than 40 minutes for fat and 80 minutes for skin/muscle with any tissue reaching 55°C assumed to be above the CEM_{43} minute limit regardless of time spent at that temperature. The tissue type is selected as Dermis on the basis that this part of the evaluation is for homogenous lesion of similar properties to the Dermis of full depth of the thermal simulation space.

The cooling effect is achieved by adjusting the surface wind speed which essentially removes energy from the skin surface so that the maximum temperatures achieved are below rather than at the surface. Although unrealistic wind speeds of 100 m/s and above are used in this simulation, the net effect is to remove incident power at the skin surface, a proportion of which is related to the difference in skin and wind (i.e. Ambient) temperature. Incident power is applied for 30 seconds which is assumed to be the top limit that could be envisaged in any potential clinical setting. Apart from the actual temperature simulation, the modeller applies full power until the temperature limit is reached at any depth in the tissue and then re-applies full power when the temperature falls – in practice this would be difficult if not impossible to achieve but is useful here for comparison purposes. For each frequency 'wind speeds' of 1 – 10000 m/s are simulated and the resultant ablation and damage depth is plotted.

As has previously been described [17] , the MSA was tested with the facility to include a coolant spray. This method could be used for forced cooling as shown in Figure 26.

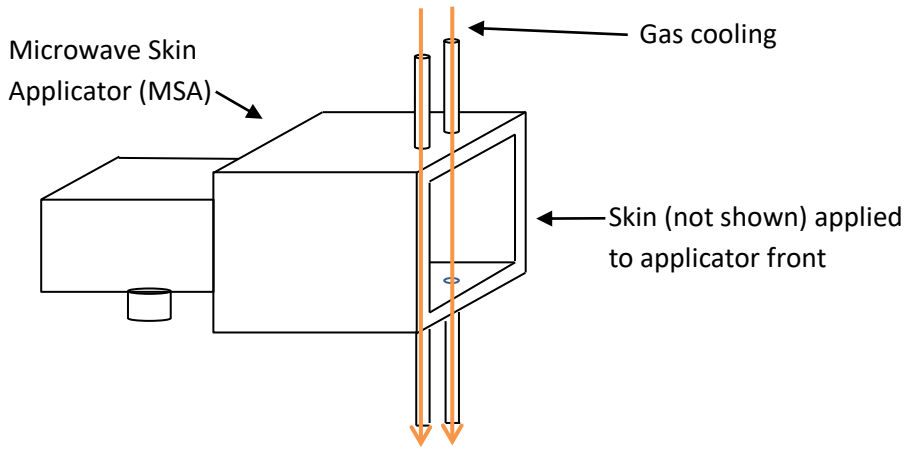


Figure 26 Applicator with forced gas cooling

Results

As shown in Figure 27, at 5.8GHz 1 m/s wind speed 4.75mm ablation can be achieved. This increases to 7.0 mm with sufficient cooling. Note that high wind speeds are obviously unattainable, but the equivalent cooling effect might be achieved with liquid and/or lower gas/liquid temperatures. At 1m/s wind speed tissue at 7.75mm and deeper is within a CEM_{43} limit of 80minutes (for skin/muscle) meaning that for 4.75mm ablation depth there is 3mm of damaged or potentially damaged tissue. As the wind speed increases the ablation depth increases at approximately the same rate as the damage depth so that at 500m/s a 6.6mm ablation might be achieved with damage to 9.6mm i.e. 3mm damaged margin.

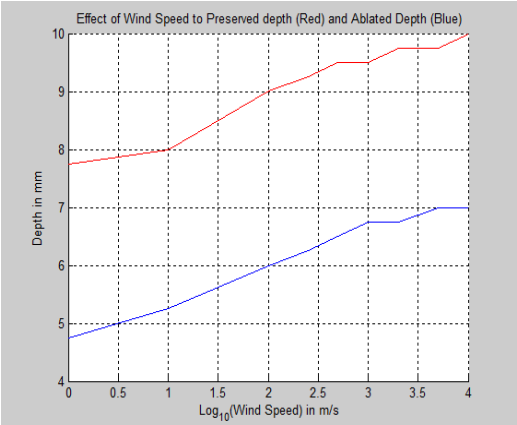


Figure 27 Wind Speed vs Ablated and Preserved Depth at 5.8GHz

As shown in Figure 28, at 14.5GHz 1 m/s wind speed 2.5mm ablation can be achieved. This increases to 3.75mm with sufficient cooling. Note that high wind speeds are obviously unattainable, but the equivalent cooling effect might be achieved with liquid and/or lower gas/liquid temperatures. At 1m/s wind speed tissue at 4.5mm and below is within a CEM_{43} limit of 80minutes (for skin/muscle) meaning that for 2.5mm ablation depth there is 2mm of damaged or potentially damaged tissue. As the wind speed increases the ablation depth increases at an apparently faster rate as the damage depth so that at 500m/s 3.5mm ablation might be achieved with damage to 5mm i.e. produces a 1.5mm damaged margin.

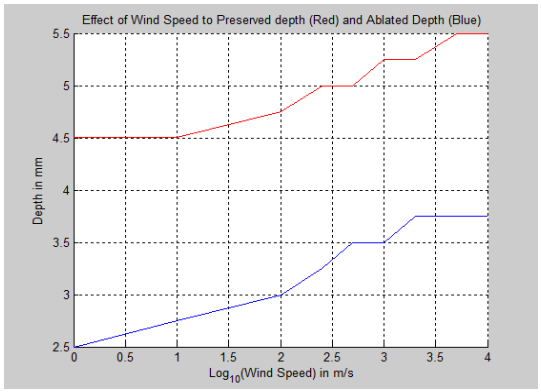


Figure 28 Wind Speed vs Ablated and Preserved Depth at 14.5GHz

As shown in Figure 29, at 24.125GHz 1 m/s wind speed 2.0mm ablation can be achieved. This increases to 2.75mm with sufficient cooling. At 1m/s wind speed tissue at 3.75mm and below is within a CEM_{43} limit of 80minutes (for skin/muscle) meaning that for 2.0mm ablation depth there is 1.75mm of damaged or potentially damaged tissue. As the wind speed increases the ablation depth increases at approximately the same rate as the damage depth so that at 500m/s 2.5mm ablation might be achieved with damage to 4mm i.e. produces a 1.5mm damaged margin.

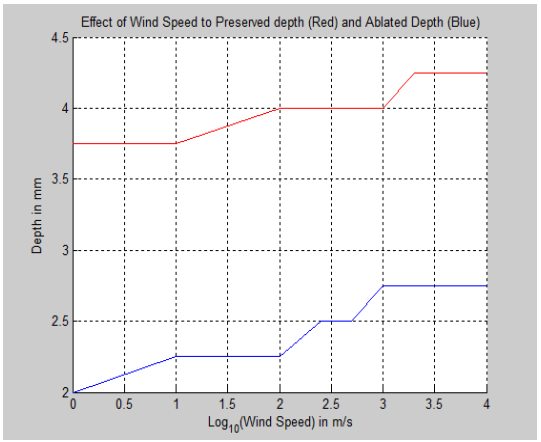


Figure 29 Wind Speed vs Ablated and Preserved Depth 24.125GHz

Effect of Forced Temperature Bias at 5.8, 14.5 and 24.125 GHz

The following simulations are similar to those presented in the section on Surface Cooling above except that the skin surface temperature is set to a specific temperature in the range of 0-70°C during a 30 second heating period after which the simulation continues to run and accumulate CEM_{43} minutes. The skin surface temperature is set from the start to the end of the heating cycle after which normal convection boundary conditions apply (a further 30 seconds) . In practice it may be difficult to achieve a constant skin surface temperature while heating but possibly the MSA could be filled with a fluid which could be constantly recycled although this approach would still have the problem of ensuring that temperature limits below the skin surface were not exceeded. The following simulations were run such that the applied power was limited by a temperature ceiling of 75°C at any depth beneath the skin surface.

Results

As can be seen from Figure 30 all 3 frequencies demonstrate an increase in Ablation depth and corresponding Damage depth inversely related to the forced temperature.

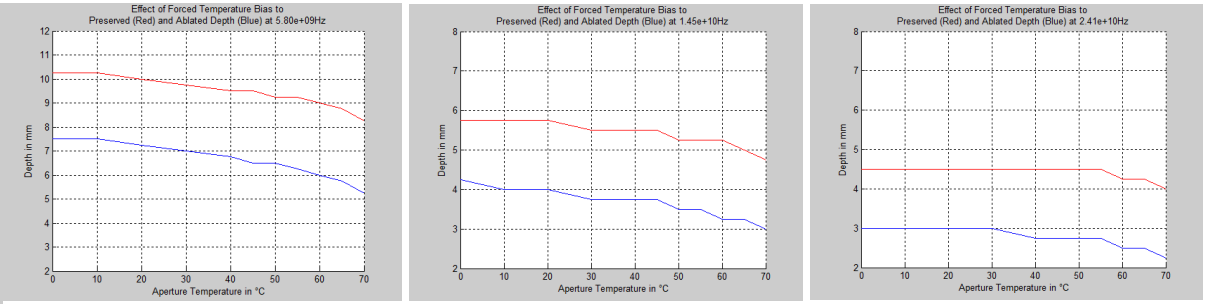


Figure 30 Effect of Forced Temperature bias of 0-70 °C at 5.8GHz (left), 14.5GHz (middle) and 24.125GHz (right)

The results for 0°C forced temperature are compared with the Base case in Table 5.

Table 5 Comparison of Base Case and Forced Temperature bias

	Base Case			Forced to 0 °C		
Frequency GHz	**Ablation mm**	**Damage mm**	**Damaged Margin mm**	**Ablation mm**	**Damage mm**	**Damaged Margin mm**
5.8	4.75	7.75	3.0	7.5	10.2	2.7
14.5	2.5	4.5	2.0	4.25	5.8	1.55
24.125	2.0	3.75	1.75	3.0	4.5	1.5

All frequencies demonstrate an increase in Ablation Depth and a reduction in Damaged Margin implying that greater ablations depths can be achieved with less damage to viable tissue. However, a method of ensuring that temperatures below the skin surface are limited to a predefined temperature (in this case 75°C) would need to be developed. Possibly this could take the form of suspending the ablation period and measuring the surface temperature rise, but again this may be difficult in the presence of a coolant and this approach may be more suited to the convective cooling described above. Heating profile for forced aperture temperature of 20°C is given in Appendix 1B where it may be seen that the surface (0mm depth) is forced to a specific temperature during the heating cycle.

Effect of Subcutaneous fat on Ablation depth

This section compares the Ablation depth and Damage depth at various frequencies in the presence of various depths of subcutaneous fat. The EM and power loss profile is very different from homogeneous tissue as the fat layer reflects a proportion of the EM wave, and as the fat layer has a lower permittivity than skin, the E field is not inverted and consequently produces constructive interference at the boundary. This effect is shown on the EM profiles below. In all simulation runs of 60 seconds, the power is turned off when the required temperature/depth is reached – if it is reached. The maximum temperature is regulated to 75°C at 0mm with 40W incident power.

EM Results

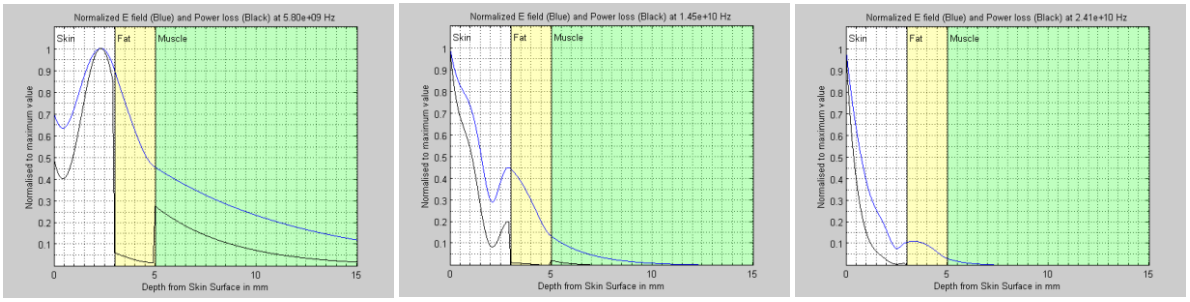


Figure 31 EM Profile for 2mm of fat at 3mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

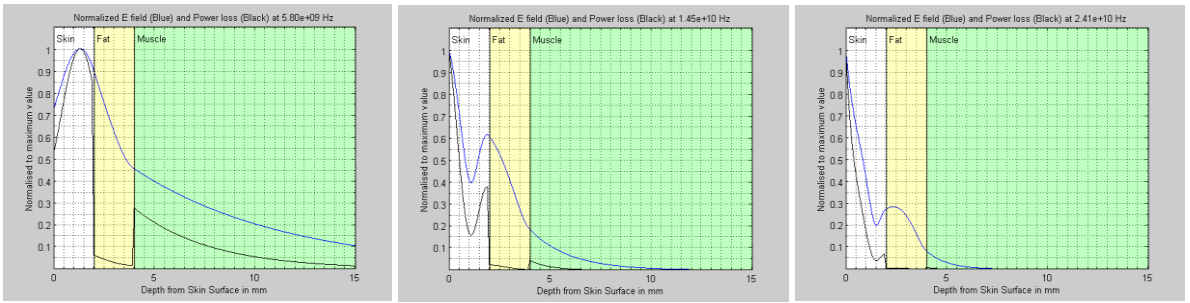


Figure 32 EM Profile for 2mm of fat at 2mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

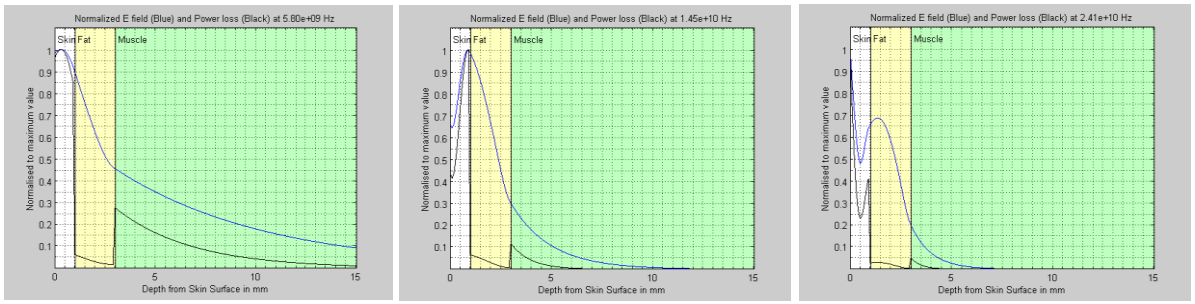


Figure 33 EM Profile for 2mm of fat at 1mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

The main effect of the subcutaneous fat is shown in Figure 31 left, Figure 32 left and Figure 33 middle where more power is dissipated near the fat boundary than the skin surface. Results for 1mm fat thickness are shown in appendix 1C.

Heating Results

The time to Ablation for various depths is shown below in Figure 34 and Table 6. In many cases for 3mm fat depth, the auto-regulation temperature is not reached – i.e. the ablation depth and temperature is reached before the maximum surface temperature is reached.

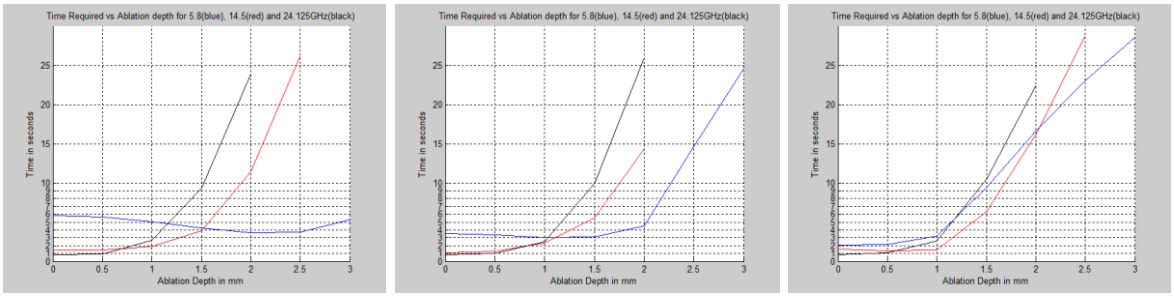


Figure 34 Time vs Ablation depth for 2mm of fat at depths 3mm (left), 2mm (middle) and 1mm (right)

Table 6 Ablation time vs Ablation depth for 1-3mm Fat Depth

	5.8GHz Time to Ablate seconds & Auto reached Y/N			14.5GHz Time to Ablate seconds & Auto reached Y/N			24.125GHz Time to Ablate seconds & Auto reached Y/N		
	Fat Depth mm			Fat Depth mm			Fat Depth mm		
Ablation depth mm	**3**	**2**	**1**	**3**	**2**	**1**	**3**	**2**	**1**
0.0	5.84 N	3.58 N	2.1 N	1.42 N	1.12 N	1.54 N	0.8 N	0.8 N	0.8 N
0.5	5.64 N	3.38 N	2.14 N	1.48 N	1.3 N	1.38 N	0.96 N	0.98 N	1.06 N
1.0	5.06 N	3.02 N	3.16 N	1.9 N	2.32 Y	1.42 N	2.66 Y	2.54 Y	2.58 Y
1.5	4.24 N	3.12 N	9.4 Y	3.92 Y	5.54 Y	6.36 Y	9.36 Y	9.96 Y	10.4 Y
2.0	3.64 N	4.48 N	16.6 Y	11.5 Y	14.4 Y	16.2 Y	23.9 Y	25.9 Y	22.4 Y
2.5	3.72 N	14.6 Y	23.0 Y	26.2 Y	/ Y	28.7 Y	/ Y	/ Y	/ Y
3.0	5.3 N	24.6 Y	28.5 Y	/ Y	/ Y	/	/ Y	/ Y	/ Y

(the symbol '/' indicates that the ablation temperature was not reached)

The time to ablation results show a greater sensitivity to the presence of Fat with lower frequencies. For 24.125GHz 0-1mm Ablation depth, approximate times are within 0.1 seconds, 1 second for 1.5mm and 3.5seconds for 2mm; For 14.5GHz there is a wider variation: 0mm: 0.42s, 0.5mm: 0.18s, 1.0mm: 0.9s, 1.5mm: 2.44s, and 2.0mm: 4.7s; For 5.8GHz the variation is greater - for example at 2.0mm depth the ablation time varies from 3.64 to 16.6 seconds. This variation implies that 5.8GHz might not be as precise as the higher frequencies in the presence of subcutaneous fat where the depth is not known before the Ablation process, 14.5GHz appears more precise and 24.125GHz relatively insensitive to the Fat layer. However, the higher frequencies have a lower depth of ablation. This has implications for estimating the dose (i.e. Ablation time) required in order to attain a particular Ablation depth – this is explored later in this chapter. In addition, for many of the greater depths and lower frequencies, the Temperature Regulation has not come into operation, which might mean that any possible compensatory effect has not been achieved.

Damaged Tissue

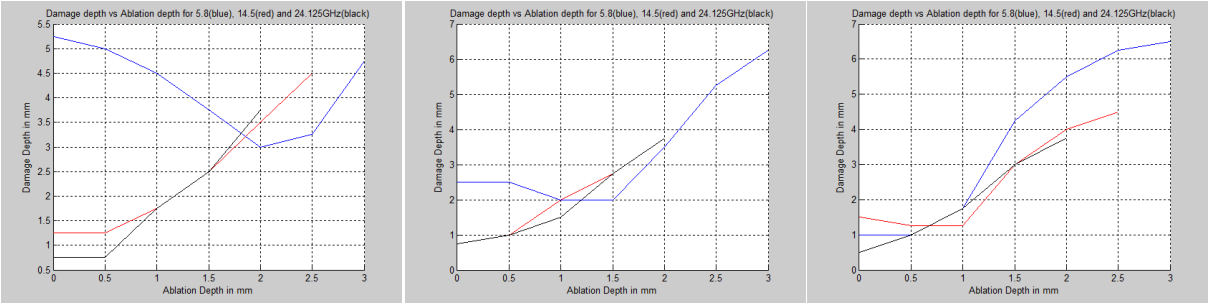


Figure 35 Damage depth vs Ablation Depth for Fat Depth of 3mm (Left) , 2mm (Middle) and 1mm (Right)

Figure 35 above shows the relationship between the Damage depth and the Ablation depth. Ideally it would be a 1:1 relationship where damage does not exceed ablation. However there is considerable departure from this ideal especially at 5.8GHz and 3mm fat depth where 1mm ablation produces a 4.5mm damaged margin. 14.5GHz and 24.125GHz are very similar above 1mm Ablation depth with 24.125GHz producing generally less damage below 1mm ablation depth however 24.125GHz does not reach more than 2mm.

Examination of higher temperatures below surface

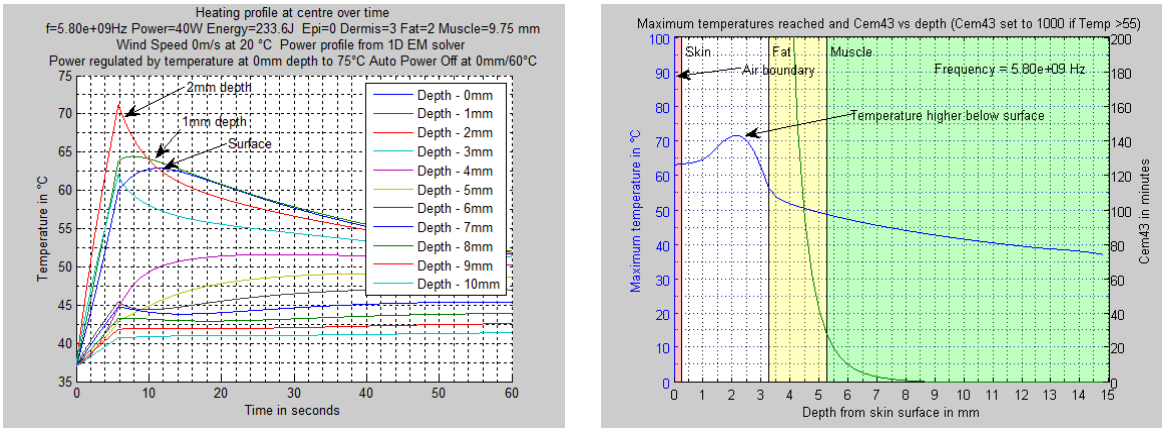


Figure 36 Heating Profile vs Time (Left) and Maximum Temperatures and CEM43 (Right)

The heating profile and maximum temperatures for 5.8GHz, 3mm Fat depth and 0mm ablation depth is shown in Figure 36 above. The surface temperature of 60°C is reached after 5.84 seconds, but as can be seen on the plot on the left, the surface temperature continues to rise for approximately 6 seconds after the power is switched off. To check whether this effect might happen (and so produce higher than expected temperatures) the presence of fat close to the surface during a real ablation procedure might be checked by applying a short, low power heating pulse.

For example referring to Figure 37 which uses 10W, 1 second pulse at 14.5GHz:

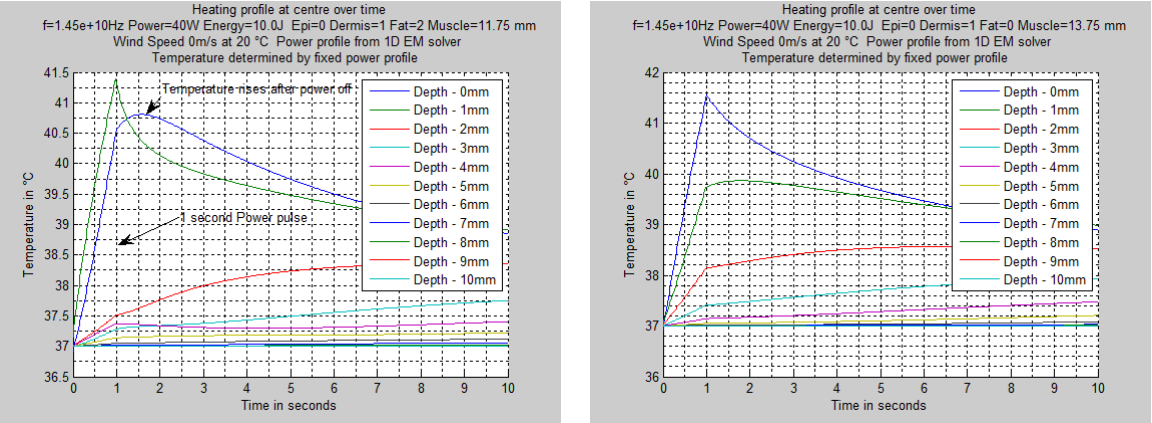


Figure 37 Temperature heating profile for 1 second 10W pulse for 2mm Fat (left) and no Fat (right) at 1mm depth

The measurement of this temperature rise or the slope of the fall after the power is turned off may be useful to modify the heating profile to tighten control of the ablation process although this is not explored further here. Note that the 40W power labelled on the plots is the total power before modification by the input heating profile vector which is 0.25 for 1 second followed by 0 for the remainder of the run, the energy of 10J takes the heating profile vector into account.

Estimation of Dose for specific Ablation depths for various Fat and Dermis thicknesses

The purpose here is to get a fixed time dosage estimate designed to achieve a particular ablation depth when the Fat depth and thickness and Dermis thickness is not known. The simulation was run individually for 5.8GHz, 14.5GHz and 24.125GHz for all combinations of 0, 1 and 3mm Fat thickness and 1, 2 and 3mm of Dermis thickness, 40Watts and 1-15 second heating time, 60 second run time and power regulated by surface temperature to a maximum of 75°C. The result plots are overlaid for each frequency and shown in Figure 38.

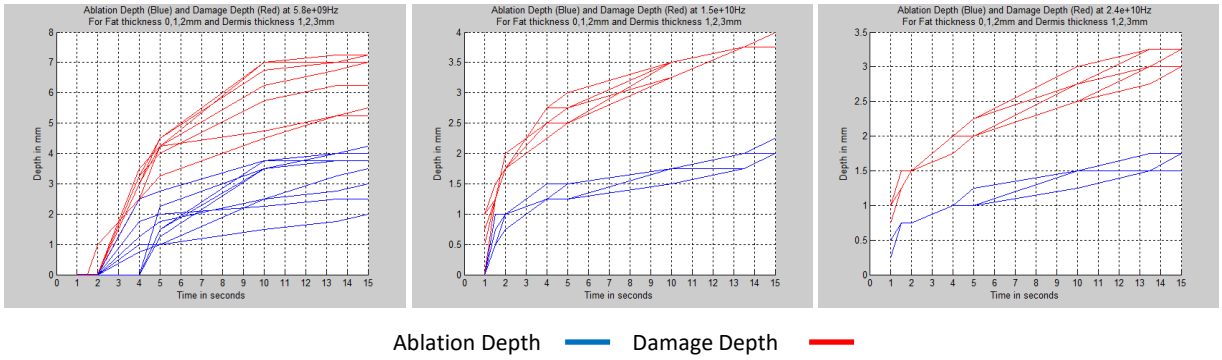


Figure 38 Ablation and Damage depths vs Time for combinations of Fat (0,1,2 mm) and Dermis thicknesses (1,2,3 mm) for 5.8GHz (left) , 14.5GHz (middle) and 24.125GHz (right).

For the Fat and Dermis conditions simulated here, there is a clear difference between 5.8GHz and the higher frequencies. For example, to ablate to a minimum of 1mm at 5.8GHz a 5 second heating time would be required - but this could ablate to 2.8 mm with a damaged margin up to 4.5mm; At 14.5GHz 3 seconds would be required with possible ablation to 1.25mm with 2.5mm margin; and at 24.125GHz 4 seconds would be required with no variation in ablation depth and with damage up to 2mm.

Conclusion

The Temperature Regulation system would appear to provide advantages over a system where the power is pre-determined, giving a high degree of compensation for varying power levels and uncertainty for some skin thermal parameters. As described in later chapters, surface temperature regulation is readily achievable with the use of a thermal camera which may also have advantages for self-calibration given that a suitable standard tissue phantom was available. In addition, the use of a short heating pulse and subsequent monitoring the surface temperature at the end of the pulse provides some information about the structure of the skin and whether some modification to the heating process is needed.

The higher frequencies of 14.5GHz and 24.125GHz would appear to provide better Ablation depth and damage margin control over 5.8GHz in the presence of unknown thickness and depth of subcutaneous fat. 14.5GHz may provide a slightly deeper Ablation compared to 24.125GHz. All three frequencies attain greater Ablation depth from convective or forced temperature bias at the Skin surface, but these would interfere with the surface temperature regulation.

A starting point for dosage as shown in Figure 38 may prove to be useful in later experiments.

Further Work

Monitoring of the surface temperature rise during ablation or during a short heating pulse could be examined further with a view to correlating this to skin heat conductivity and the presence of fat and so providing a better estimate of required power and time dosage.

The thermal simulations most sensitive to the epidermis may be the forced cooling and convection instances where there is substantial energy transfer through the epidermis. The above simulations did not include the epidermis due to the very long simulation times required – the simulation cell being a cube 0.25mm on each side results in an approximate 40x increase in simulation time after stability criteria are taken into account if a 0.1mm epidermis is to be included. Sub-gridding would be helpful here but substantial testing might be needed to gain confidence compared to the existing simple model.

In Appendix 1C page 130 Figure 111, Figure 113 and Figure 115 (middle and right plots) there appears to be a close relationship between the Damage margin and the Total Power used. This relationship could be investigated further with the aim of predicting damage margin during the ablation process.

Thermal and electrical skin parameters have been assumed constant for the various simulations which is unlikely to be the case in practice. Again, compensation for changes to these will need to be taken into account.

Works Cited

[1] G. C. Van Rhoon, T. Samaras, P. S. Yarmolenko, M. W. Dewhirst, E. Neufeld and N. Kuster, "CEM43°C thermal dose thresholds: a potential guide for magnetic resonance radiofrequency exposure levels?," *European Radiology,* vol. 23, no. 8, pp. 2215-2227, 2013.

[2] C. Buccella, V. De Santis and M. Feliziani, "Numerical Prediction of SAR and Thermal Elevation in a 0.25mm 3-D model of the Human Eye Exposed to Handheld Transmitters," in *2007 IEEE International Symposium on Electromagnetic Compatibility pages 1-6*, 2007.

[3] H. H. Pennes, "Analysis of Tissue and Arterial Blood Temperatures in the Resting Human Forearm," *Journal of Applied Physiology,* vol. 1, no. 2, pp. 93-122, 1948.

[4] J. Wang and O. Fujiwara, "FDTD Computation of Temperature Rise in the Human Head for Portable Telephones," *IEEE Transactions on Microwave Theory and Techniques,* vol. 47, no. 8, pp. 1528-1534, 1999.

[5] U. S. Inan and R. A. Marshall, Numerical Electromagnetics The FDTD Method, Cambridge University Press, 2011.

[6] T.-Y. Cheng and C. Herman, "Analysis of skin cooling for quantitative dynamic infrared imaging of near-surface lesions," *International Journal of Thermal Sciences ,* vol. 86, pp. 175-188, 2014.

[7] R. J. de Dear, E. Arens, Z. Hui and M. Oguro, "Convective and radiative heat transfer coefficients for individual human body segments," *International Journal of Biometeorology,* vol. 40, no. 3, pp. 141-156, 1997.

[8] M. W. Dewhirst, B. L. Viglianti, M. Lora-Michiels, M. Hanson and P. J. Hoopes, "Basic principles of thermal dosimetry and thermal thresholds for tissue damage from hyperthermia," *International Journal of Hyperthermia, DOI:10.1080/0265673031000119006,* vol. 19, no. 3, pp. 267-294, 2003.

[9] P. Bernardi, M. Cavagnaro, S. Pisa and E. Piuzzi, "Specific Absorption Rate and Temperature Elevation in a Subject Exposed in the Far-Field of Radio-Frequency Sources Operating in the 10-900-MHz Range," *IEEE Transactions on Biomedical Engineering,* vol. 50, no. 3, pp. 295-304, 2003.

[10] K. S. Yee, "Numerical Solution of Initial Boundary Problems involving Maxwell's Equations in Isotropic Media," *IEEE Antennas and Propagation Society,* vol. 14 , no. 3, 1966.

[11] B. S. Guru and H. R. Hiziroglu, Electromagnetic Field Theory Fundamentals Second Edition, Cambridge: Cambridge University Press, 2004.

[12] *The Mathsworks Inc, Matlab R2012A Student Edition,* Natick, MA, USA, 2012.

[13] U. S. Inan and A. S. Inan, Electromagnetic Waves, Upper Saddle River: Prentice Hall, 1999.

[14] “List of integrals of trigonometric functions,” Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/List_of_integrals_of_trigonometric_functions. [Accessed 22 10 2016].

[15] D. Andreuccetti, R. Fossi and C. Petrucci, “An Internet resource for the calculation of the dielectric properties of body tissues in the frequency range 10 Hz - 100 GHz, Based on data published by C.Gabriel et al. in 1996.,” IFAC-CNR, Florence (Italy), 1997. [Online]. Available: Website at http://niremf.ifac.cnr.it/tissprop/. [Accessed 08 9 2016].

[16] Microsoft Inc., *Windows 10 Operating system,* Redmond, Washington, USA.

[17] *Discussion about the Microwave Skin Applicator with Prof Chris Hancock, 2012 , Bangor.*

Chapter 3 EM modelling of the Microwave Skin Applicator (MSA)

Contents

Introduction

The goal of the MSA project is to produce uniform microwave energy deposition into human skin in order to ablate small (say 5mm diameter) skin lesions. Substantial work [1] had been done previously by Professor Hancock and his associates which resulted in a design comprised of two adjoining waveguides cavities, with a monopole antenna to launch EM energy in the first cavity and an aperture bounding the other (output) cavity (see figure 1 below).

In order to evaluate and improve the existing MSA and provide some insight into the electromagnetic wave propagation, a Matlab [2] simulation program was written. This simulation was subject to several iterations to include different methods for setting boundary conditions, and varying a range of parameters such as simulation time and materials within the MSA. The main aim of these simulations was to evaluate methods of enhancing the design of the existing MSA's resonant output cavity with internal side blocks to produce an electric (E) field intensity that is as flat as possible across the opening of the aperture - the characteristic E field intensity of a waveguide excited with a vertical (Y axis) monopole is already constant along the vertical (Y) axis but has a cosine shape along the horizontal (X) axis with the maximum intensity in the middle of the aperture. The heating profile is related to power dissipation which is related to the square of the E field.

Various methods were evaluated to flatten the cosine shape: 1) PTFE side blocks, 2) Ferrite side blocks, and 3) Metamaterial strips, all located on both sides of the resonant chamber were considered.

In effect, the purpose of the blocks on the side of the chamber is to approximate a Perfect Magnetic Conductor to contain the wave with the result that the vertical E-field does not get zeroed by the sides of the chamber, so enabling a more even intensity in the horizontal direction as described in [3]. As the E-field is perpendicular to the top/bottom of the chamber, it is already constant in the vertical direction.

Brief Theoretical basis of the 3D simulation.

The EM modeller in Chapter 2 outlines the theory and demonstrates the process of discretising Maxwell's equations in 1 Dimension. In [4] , Yee describes a method of solving Maxwell's equations by dividing the problem volume into finite size cells. Each cell is described as a cube (or similar shape) with electrical field vectors along the centres of its edges and magnetic field vectors central and perpendicular to its faces. The magnetic field spatial definition is essentially the same as the electrical field spatial definition except perpendicular and staggered by half a cell, see [4] Fig 1 and [5] Figure 4.1 page 73, this provides a means of calculating an E field component from the Magnetic components that circulate around it or vice versa and by splitting time into discrete steps and half steps. The relationship between these electric and magnetic circulating fields and the corresponding magnetic and electric components are described by Maxwell's equations which are based on Faraday's and Ampere's laws respectively as noted in [6] pg. 3.

The examples below show the equations relating to the calculation of the X components of E, D, H and B; the Y and Z components are calculated similarly.

Numerical solution for free space

The following Matlab [2] update equations are for free space i.e. lossless medium. Refer to chapter 2 for Maxwell's equations.

Calculating D from H

Refer to [5] for the full derivations, but an example for E(t,x,y,z) from pg. 85 and using $D = \varepsilon E$ where H has been evaluated ½ step in time previously.

$$D_x(n+1,i+½,j,k) = D_x(n,i+½,j,k)$$

$$+ \left(\Delta t \left(\left[H_z(n+½,i+½,j+½,k) - H_z(n+½,i+½,j-½,k) \right) / \Delta y \right] \right.$$

$$\left. - \left[H_z(n+½,i+½,j,k+½) - H_z(n+½,i+½,j,k-½) \right) / \Delta z \right]))$$

This translates into the Matlab equivalent statement (see Appendix 2 for program code[6]) for the current time instant n:

```
D_X(Xs:Xe,Ys:Ye,Zs:Ze) = D_X(Xs:Xe,Ys:Ye,Zs:Ze)
                        + ((Delta_T)* ((diff(H_Z(Xs:Xe, Ys-1:Ye,  Zs:Ze    ),1,2)/Delta_Y)
                                      - diff(H_Y(Xs:Xe,     Ys:Ye,  Zs-1:Ze),1,3)/Delta_Z));
```

Where Xs, Ys and Zs = correspond to the start of the problem space and Xe, Ye and Ze correspond to the end of the problem space.

[6] Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

The Y and Z components of D are calculated similarly.

Calculating E from D

$D = \varepsilon E$ produces $E = D/\varepsilon$

Producing the Matlab equivalent:

```
E_X(Xs:Xe,Ys:Ye,Zs:Ze) = R_e_X(Xs:Xe,Ys:Ye,Zs:Ze) .* D_X(Xs:Xe,Ys:Ye,Zs:Ze) ;
```

Where R_e_X is $1/\varepsilon_0$ and ε_0 is the permittivity of free space. Note that the matrix R_e_X (and similar Y and Z matrices) allow for a different permittivity on a cell by cell basis (and X,Y and Z basis within a cell), so allowing a simple means of simulating different materials in different parts of the overall volume.

Calculating B from E

And similarly [5] page 86 with $B = \mu H$

$$B_x(n+½,i,j+½,k+½) = B_x(n-½,i,j+½,k+½)$$
$$+ (\Delta t \ (\ [E_Y(n,i+1,j,k+½) - E_Y(n,i,j,k+½) \ / \Delta z]$$
$$- \ [E_z(n,i,j+1,k+½) - E_z(n,i,j,k+½) \ / \Delta y]))$$

Producing the Matlab equivalent:

```
B_X(Xs:Xe,Ys:Ye,Zs:Ze) = B_X(Xs:Xe,Ys:Ye,Zs:Ze)
                        + ((Delta_T) * (  (diff(E_Y(Xs:Xe  ,Ys:Ye  ,  Zs:Ze+1),1,3)/Delta_Z)
                                        - (diff(E_Z(Xs:Xe  ,Ys:Ye+1,Zs:Ze    ),1,2)/Delta_Y)));
```

Calculating H from B

$B = \mu H$ producing $H = B/\mu$

Producing the equivalent Matlab code:

```
H_X(Xs:Xe,Ys:Ye,Zs:Ze) = (1/Fmu_0)*B_X(Xs:Xe,Ys:Ye,Zs:Ze);
```

Where Fmu_0 is the permeability of free space.

A single iteration of 1 time step consists of :

H → D

D → E

E → B

B → H

The sequence is repeated for the required number of time steps.

Boundary conditions

The boundary condition for a PEC surface is that the tangential component of electrical field is zero at the surface Yee [4]and [5] pg. 22. This is achieved in the simulation by enforcing the appropriate E field values to zero shortly after all the E fields have been computed from H or B – i.e. overriding the computed values at the boundary. As the problem space comprises a 3D volume of cubes of dimensions Δx, Δy, Δz, this does lead to a stepped profile for curved surfaces - the MSA however, is largely orthogonal. For PMC boundaries, the appropriate magnetic components are zeroed shortly after default calculation from E or D.

Ferrite

Inan and Marshall [5] pg 281-287 provide a derivation for the calculation of H from B & H in ferrite materials. Their example is given for a background magnetic field parallel with the Z axis whereas the MSA's magnetic field would be in the X axis as per [3], consequently the update equations to produce H_z in Matlab code (Appsim8.m) are, for example:

```
H_Z(Ferrite_Loc) =  (C1P*B_Z(Ferrite_Loc))    + (C2P*B_Z_PP(Ferrite_Loc)) +
                    (C3P*B_Z_P(Ferrite_Loc))  + (C4P*H_Z_P(Ferrite_Loc))  +
                    (C5P*H_Z_PP(Ferrite_Loc)) + (C6P*H_Y_PP(Ferrite_Loc)) -
                    (C7P*B_Y(Ferrite_Loc))    - (C8P*B_Y_PP(Ferrite_Loc))  -
                    (C9P*B_Y_P(Ferrite_Loc))  - (C10P*H_Y_P(Ferrite_Loc));
```

Where Ferrite_Loc is a matrix denoting the location of the Ferrite material and B_Z_P corresponds to $B_z(n-1/2)$ and B_Z_PP corresponds to $B_z(n-3/2)$ etc. Refer to Appsim8 for calculation of the coefficients C1P..C10P.

The implementation here substitutes Z for X and vice versa directed magnetic field components from [5] so that:

$$\begin{bmatrix} B_z \\ B_y \\ B_x \end{bmatrix} = \mu_0 \begin{bmatrix} U_{11} & U_{12} & 0 \\ U_{21} & U_{22} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} H_z \\ H_y \\ H_x \end{bmatrix}$$

Formulae for U11 and U12 are given in [5] with similar notation to [7] pg. 369. Refer to Appsim8 for other relevant formulae which have been adapted from [5]. Note that as outlined later useful results were not obtained most likely due to this adaption/implementation or choice of parameters.

PML

Initial numerical experiments were performed with the problem space is surrounded by a Perfectly Matched Layer (PML) that absorbs any waves escaping from the aperture. Again, refer to Inan and Marshall [5] chapter 9 pgs. 222- 225 for derivation of the update equations using auxiliary fields. Refer to Appsim8 lines 996-1006 and 1032-1042 for the update equations and 553-620 for coefficient initialisation. The PML implementation produced reasonable results but is not necessary for the simulations as the MSA aperture is bounded by the lossy skin tissue material i.e. the PML is outside the tissue and MSA volume. Nevertheless the simulations were run with the PML enabled so it has been retained in the program.

Excitation

Various excitation schemes have been described such as a simple voltage source or Thin Wire Approximation as per [5] pg. 154, or more elaborate schemes such as 'An improved FDTD model for the feeding gap of a thin wire Antenna' [8] or other schemes [9] and [10]. The system adopted here was a simple hard excitation of a short monopole (surrounded by PTFE) where the feeding gap (A single cell gap between the monopole and PEC surface) is set to a specific value at each iterative step according to the formula:

$$E = \frac{R * \sin(2fn\Delta t)}{\Delta y}$$

Where n is the current time step, f = 14.5GHz, Δt = time step value (usually 0.5×10^{-12} seconds), Δy = Cell size in direction parallel to the monopole. Note that although the program does provide for different cell dimensions in the x, y and z directions, all dimensions have been set to 'Delta_Len' for simplicity. Also, R ramps up linearly from 0 to 1 Volts over the first 3 cycles and then stays constant at 1 in order to provide a 'soft start'.

The current in the dipole is sampled by the line integral of H around the base of the antenna as per [11] pg. 1580.

The excitation model is shown below in Figure 39

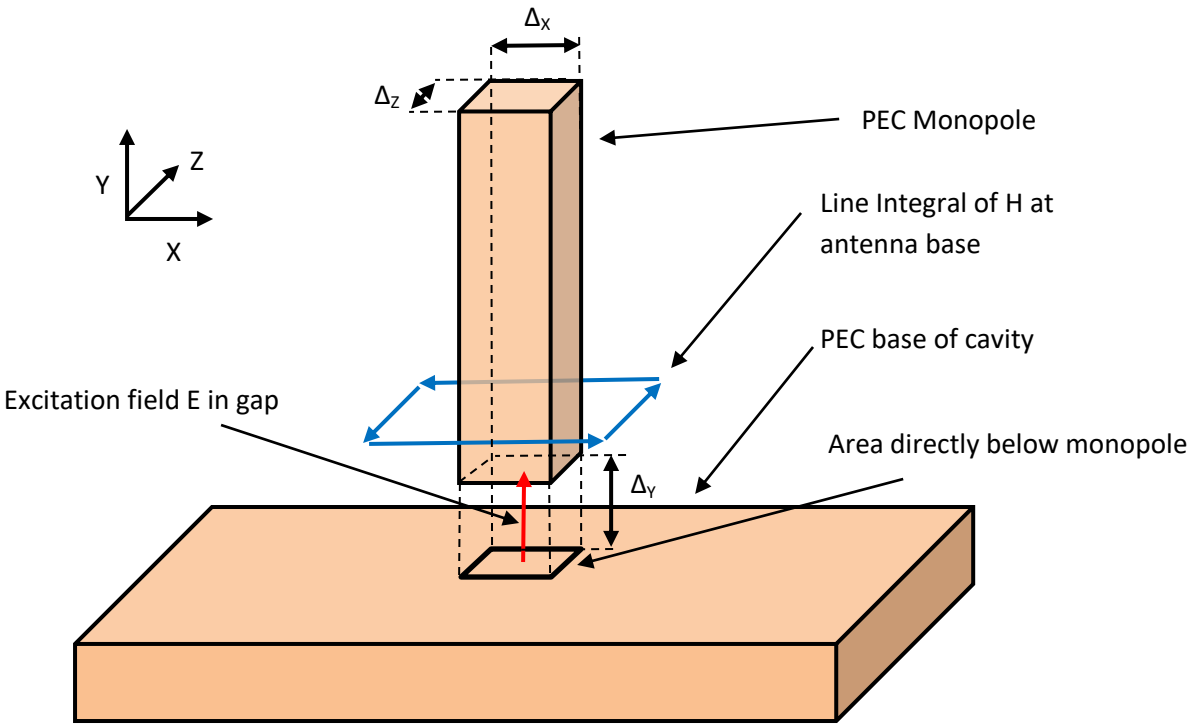


Figure 39 Excitation model

Courant Stability limit

From [7] pg. 154 the criteria for stability of a numerical solution in 3 dimensions is given by:

$$\Delta t \leq \left(\frac{1}{c\sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} + \frac{1}{(\Delta z)^2}}} \right)$$

Where Δx, Δy and Δz are the dimensions of a grid cell – note that in the Matlab program Appsim8, Δx=Δy=Δz=Delta_Len

The above equation is rewritten as:

$$c\Delta t \sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} + \frac{1}{(\Delta z)^2}} \leq 1$$

Which is calculated and displayed in Appsim8.m line 385 and calculated as 0.5193 for Delta_Len = $5x10^{-4}$ metres, Delta_T = $5x10^{-13}$ Seconds, c = $3x10^{8}$ m/s, consequently the numerical solution should be stable.

Subgridding

In order to provide better resolution around finer details of FDTD structures the cell size can simply be decreased. This approach however has the disadvantages of considerably increasing the number of cells to simulate and the CFL limit implies a further decrease in time step for stability, so, for example halving the cell dimensions x, y, z results in 8x as many cell operations (and computer memory storage requirements) for the same problem dimensions and would require half the time step, all resulting in 16x longer computing time.

Subgridding provides the means of using a finer mesh in just some parts of the problem space and so allocates resources only where needed. No subgridding facilities were built into Appsim8 although various methods such as [12] or [13] could be employed.

Modification to Original MSA

Choice of Side Blocks.

At the start of this project [14] Ch. IV pg. 599 was discussed [15] - it outlines a method of flattening the E-field profile of a horn antenna using blocks of higher relative permittivity attached to the insides of a horn antenna – this has been the basis of much of the simulation work presented here. A further possible solution [16], led to the approach of simulating a Meta-Material structure comprised of PEC strips along the insides of the MSA (see description below).

In addition [3] compares simulations of E-Field distribution in a hollow waveguide using Ferrite side-blocks, PMC and no side-blocks – this comparison is useful as the basis of further investigation into different methods of attaining flattened E-Field profile using a PMC on the sides of the waveguide. Various types of artificial magnetic conductor surfaces have been described in the literature: such as a lattice that does not conduct AC currents in a particular frequency band [17], Various frequency selective surfaces (FSS) based on slot or printed dipoles on a dielectric slab [18], square patch arrays [19] , cross type [20], Spiral [21] , tuneable ferrite [3] which shows a detailed electric field profile for a PMC lined waveguide – in particular ferrite tuned by an external magnetic field.

Camera tube

Although the camera tube at the rear of the waveguide (for the purpose of measuring the surface temperature of the skin during ablation), is not included in this simulation code, it is nevertheless helpful to note the basic concept here. There are several manufacturers of Thermopile temperature sensor arrays such as Heimann GmbH who, at the time of writing, produce an HTPA 8x8 series camera/sensor [22] which has an outside diameter of 8.8mm. In order to minimise energy leakage from the camera tube, the tube must be such that the resultant circular waveguide is operated below its dominant cutoff frequency.

From the example given in [23] pg. 360, the lower cutoff frequency of a circular waveguide for a TE_{11} mode is given by:

$f = 1.8412c/2\pi a$ Where c = speed of light in a vacuum (m/s) and a = radius of the waveguide (m)

A tube of radius 4.5mm results in a cutoff frequency $\simeq$ 19.5GHz, which is above the operating frequency of 14.5GHz. Note that this does not mean that a negligible amount of energy will escape via the camera tube and was modelled later with CST Microwave studio [24].

Physical Description of modified MSA

With reference to the diagram below, the microwave energy is injected via the Monopole antenna at the rear and Cavities 1 & 2 resonate in order to deposit the energy into the simulated human skin (Red) at the front. The Side blocks (blue) are separately described and adjusted in order to optimise the energy deposition profile.

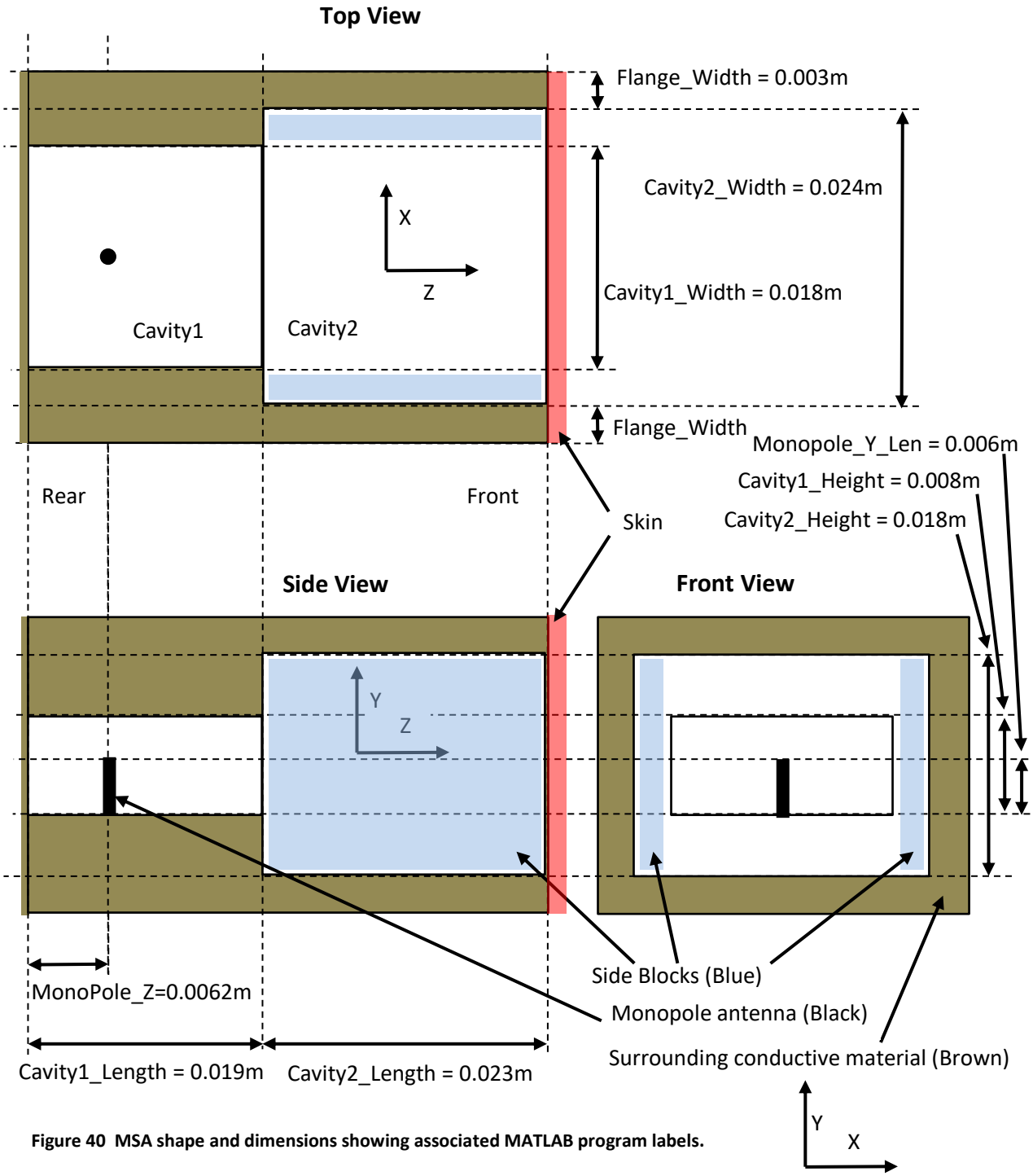


Figure 40 MSA shape and dimensions showing associated MATLAB program labels.

Simulation Program Description

The program consists of two main sections: The calling program (multiapp.8) which contains the main variable assignments that are adjusted for optimisation; and the actual program and support functions (appsim8.m) which contains the assignments related to the description of the physical space, perfectly matched layer (PML) and actual simulation loop and recording of results. Refer to Appendix 2 for program listings [7] (page 154 for Multiapp8.m and page 156 for Appsim.m).

Multiapp8.m

This is the calling program/script which assigns several variables which determine the behaviour of Appsim8.m Several results are returned from Appsim8 to MultiApp8 for reporting.

Most important simulation control variables (also see side block control below)

Program Label	Program setting	Meaning
Delta_Len	0.0005	Cell size in metres
Inject_F	14.5×10^{9}	injected frequency 14.5GHz
Cycles	10	number of full cycles to simulate
Scale_Time	1	simple way to speed up sim. Caution. Delta_T, the time step used is calculated as: Delta_T = 0.50×10^{-12} * Scale_Time Seconds
Block_Steps	1	The number of simulation runs to perform. The parameters that are modified for each run are hard coded and may be seen from line 78 to 85. Currently the code to step the block thickness by 0.5mm for each run is commented out on lines 82 & 83
Skin_Gap	0	Gap between flange and skin
Skin	true	model human skin yes/no
Skin_nr	26.88	Relative Skin Permittivity
Skin_s	13.27	Skin conductivity S/m
Capture_Start_Cycles	5	This many cycles ignored before capturing Data for input current and voltage calculation.

[7] Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

Output control

Program Label	Program setting	Meaning
MovieName	C:\Users\Paul\Desktop\Applicator-Waves0.avi'	name for animation movie
Animate	true	make E-field wave movie
Make_Movie	true	Write movie to disk
DontPlot	true	set true to inhibit plots from Appsim8
DrawPlots	true	set true to allow plots from multiple runs from Multiapp8

Formulae used by Appsim8.m

The function 'Dipole_Params' which returns the impedance of a dipole uses the Induced EMF Method as described by Wikipedia [25] and Eulers constant [26]. This is not used in the simulation but just calculated for comparison purposes. The antenna current and voltage are extracted from the instantaneous values by use of the Discrete Fourier Transform as described in [27]. The value of S_{11} displayed on the plots is calculated as 20Log(abs(S11)) where S11 = (Zin-50)/(Zin+50) as may be seen in [28].

Control of side block simulation

As mentioned previously, the side blocks are objects added to either side of the resonant cavity to flatten the deposited energy profile. Four different types of block have been modelled:

1) PTFE - Simple lossless material with an adjustable permittivity. Refer to Figure 40 for the relevant location.
2) Perfect Magnetic Conductor (PMC) – results in zero magnetic field along its surface. Refer to Figure 40.
3) Ferrite – With adjustable external magnetic field. Refer to Figure 40
4) Meta-Material - Conductive strips running in Z direction. The relevant drawing and program labels are shown in Figure 41 below.

Note that only one type of side block may be enabled (i.e. corresponding program label = true) and no checking is done within the program to avoid conflicts

PTFE side blocks

Program Label	Program setting	Meaning
Side_Blocks_PTFE	false	Side blocks are simple PTFE
Block_Thickness	0.002	Thickness in metres
Block_re	2.1	Relative permittivity

Simple Perfect Magnetic Conductor sides to cavity.

Program Label	Program setting	Meaning
Simple_Side_Blocks_PMC	false	Side blocks are Perfect Magnetic Conductors

Ferrite side blocks

Program Label	Program setting	Meaning
Ferrite_Side_Blocks_PMC	false	Side blocks are Ferrite PMC
B_0	0.45	Applied magnetic field in Tesla
E_r	10	Relative permittivity of ferrite
U_r	2	Relative permeability of ferrite
A_loss	0	Ferrite loss factor

The block thickness is hard-coded to 1 cell in Appsim8.

Meta-Material side blocks - physical description and relevant program variables

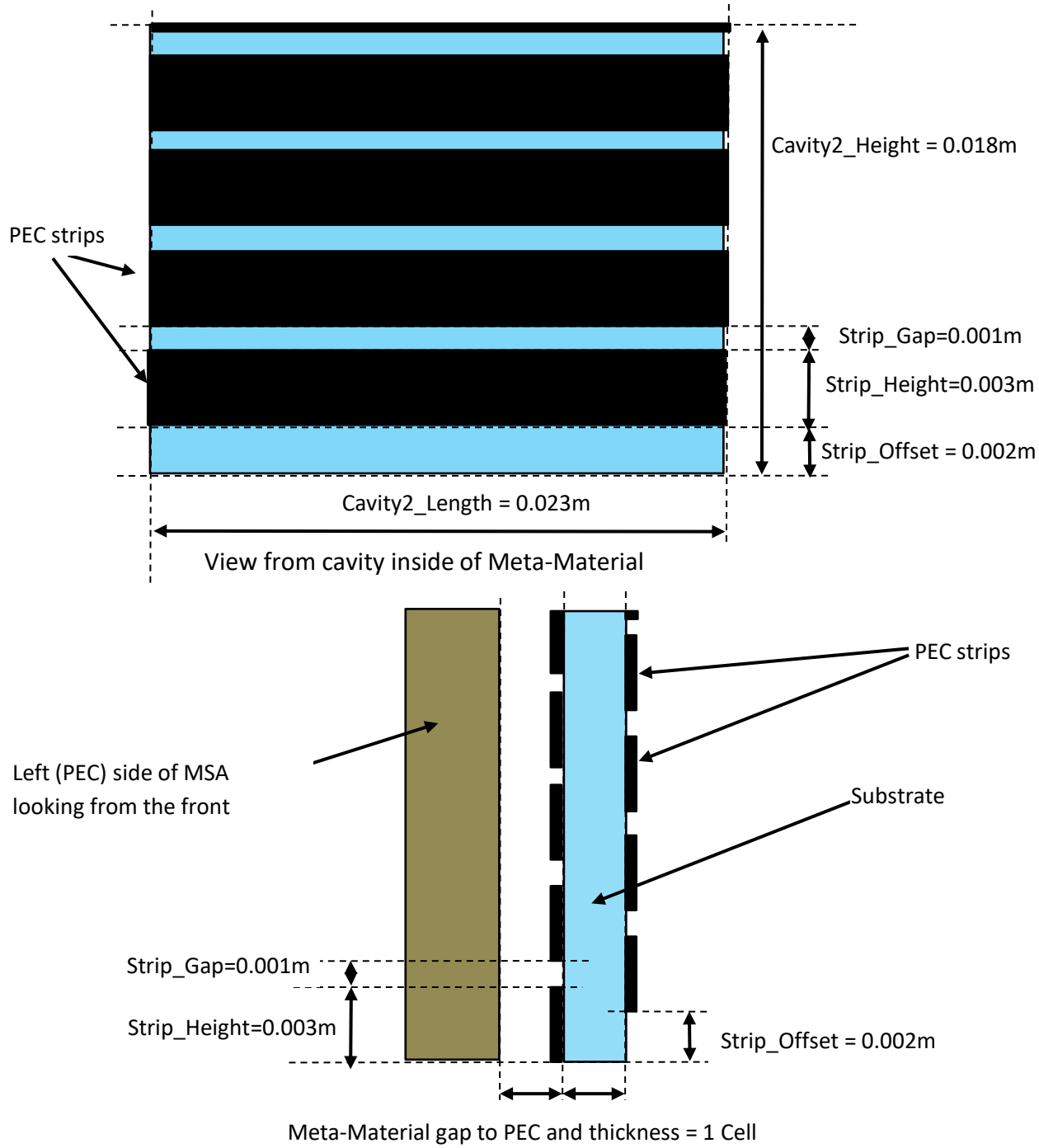


Figure 41 Side (top) and Front (lower) view of Left Meta-Material. Right side (not shown) is mirror image.

Program Label	Program setting	Meaning
Side_Blocks_META	false	Set to true to enable
Strip_Height	0.003	Height in metres
Strip_Offset	0.002	offset outside/inside in metres
Strip_Gap	0.001	gap between strips in metres
Block_re	2.1	Relative permeability of substrate

Description of the code contained in appsim8.m.

This module contains the code to prepare the problem space, then to iteratively solve Maxwell's equations over many time-steps, then (if enabled) draw graphs of the results, and (if enabled) produce an animation of the E-field over the selected time period.

The problem space is simulated in 3 dimensions using separate matrices to describe E, D electric fields and H, B magnetic fields. Further, a history of these fields is kept. The main loop calculates these fields on each iteration, and stores a recent history of a few time-steps.

Preparation for main loop

Variable and constant assignments:

Line	Action
133-140	Perfect Matched Layer (PML) constant assignment. The constant 'PML_T' defines the number of cells that make up the thickness of the PML, currently this value is set to 4. Increasing improves PML performance at the expense of execution time, and due to restrictions on matrix size, a thicker PML also results in decrease of the maximum possible size of the applicator and skin.
163-165	Problem space definition X = 4cm, Y=3.5cm, Z=7cm
169-183	Definition of cell size. CELLNUMX,Y and Z define the number of cells in the X,Y and Z directions and are used later (for example) to initialise the field matrices.
188-223	Definition of cavity dimensions
228-252	Definition of monopole antenna, note that current setting defines the monopole as surrounded by PTFE
255-256	Calculation of equivalent monopole impedance
261-350	Calculation of E fields to zero in order to simulate PEC. In order to speed execution time this is achieved by the creation of Boolean 3 dimensional matrices where a 'true' value corresponds to an E field that is to be zeroed.
352-359	Meta-Material constant calculations
361-377	Skin constant definitions and calculations
379-384	Antenna impedance variable initialisation
386	Calculation of CFL limit to determine if numerical solution to maxwells equations with the current settings (time step, cell size) is stable.
388-439	definition of ferrite location and display of graphs to aid in selecting B_0 in order to choose the appropriate magnetic field strength to approximate the required PMC.

440-461	Initialisation of animation variables.
462-468	Initialisation of 3 dimensional permittivity matrices.
473-526	Initialisation of E,D,H,B matrices. For example B_X holds the X direction B field magnitude over the entire problem space. B_X_P is the previous version, and B_X_PP is the version before that, i.e. a history of 2 iterations.
530-552	Initialisation of ferrite coefficients
553-620	Initialisation of Perfect Matched Layer (PML) coefficients.
628-672	Side block constant calculations
674-684	calculation and initialisation of cells surrounding antenna in order to simulate PTFE.

Main loop:

Calculation of Electric fields

700	**Start of loop**. Loop is executed number of times defined by 'Timesteps'
748-757	Calculation of PML fields. D calculated from H
760-768	Calculation of PML fields. E calculated from D
771-772	Calculation of default fields for entire space excluding PML. D calculated from H and E calculated from D.
775-779	Ferrite slabs: E calculated from D.
782-787	Antenna: driving injected voltage set. Antenna PEC zeroed.
796-800	Skin: E calculated
807-809	PEC enforced on E_X,Y,Z

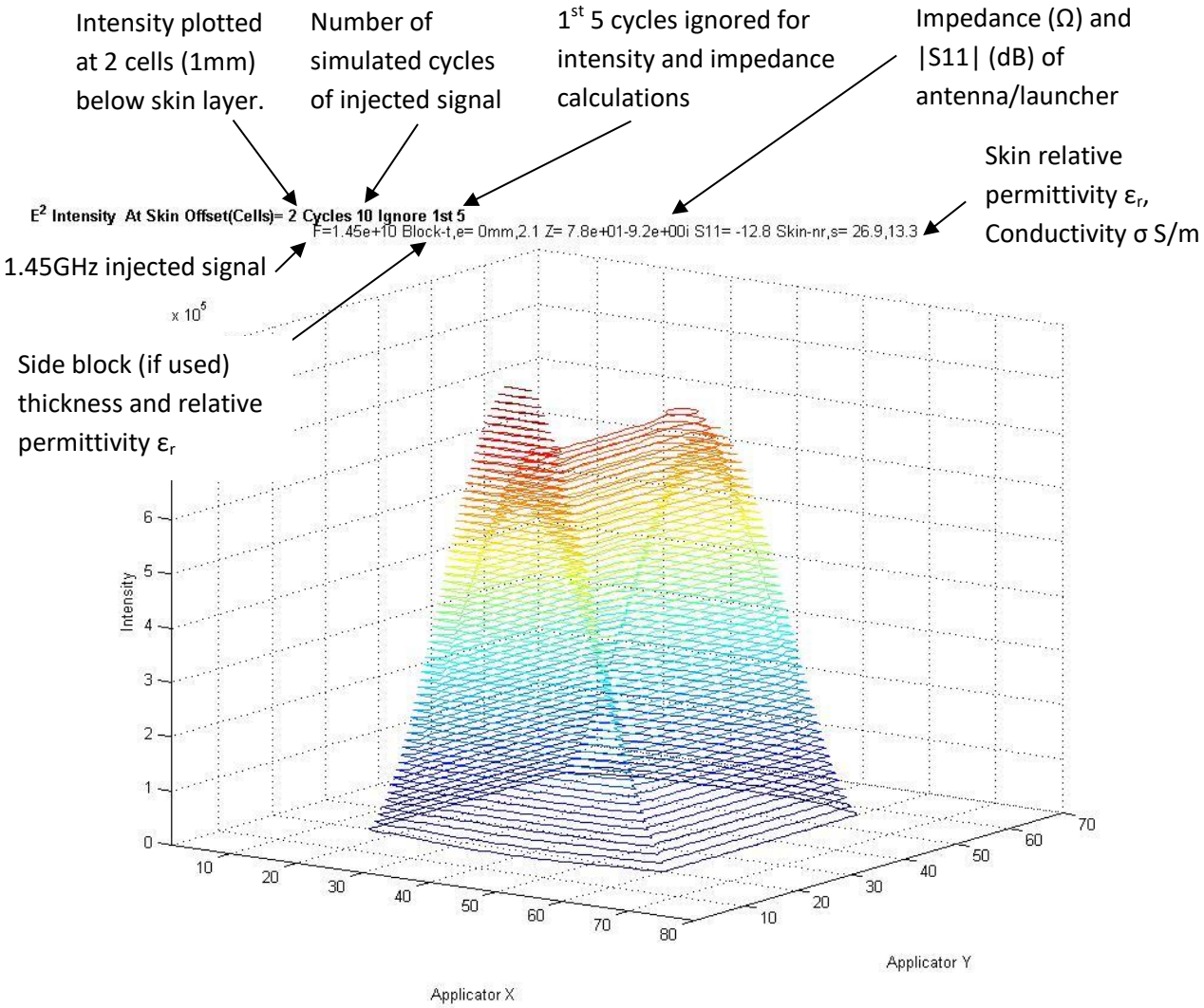
Calculation of Magnetic fields

815-823	PML: B calculated from E
826-834	PML: H calculated from B
837-838	Calculation of default fields for entire space excluding PML. B calculated from E. H calculated from B.
841-860	If enabled, PMC calculation.

862-866	If enabled, Ferrite side block calculation
868-894	Copying of field matrices in order to retain short history.
896-897	Calculation of E field intensity.
906-929	display of progress bar and update of graphs (if enabled)
931	**End of loop**
937-956	Clean up, Graphical display if enabled. Display of antenna parameters.
996-1048	Maxwell equation calculation functions.
1293-1381	Graphical display functions.
1384	Utility to calculate ideal dipole corresponding to monopole.

Results – MSA with no modification

The following plots Figure 42, Figure 43 & Figure 44 were obtained with no additions to the MSA – i.e. the MSA has no side blocks, Ferrite or PMC components. The shape of the intensity distribution appears to be Cosine^2.



X and Y axis are in units of 0.0005m, Intensity $\propto E^2$ V^2/m^2

Figure 42 XY vs Electrical field Intensity at 1mm depth into skin for MSA with no modification

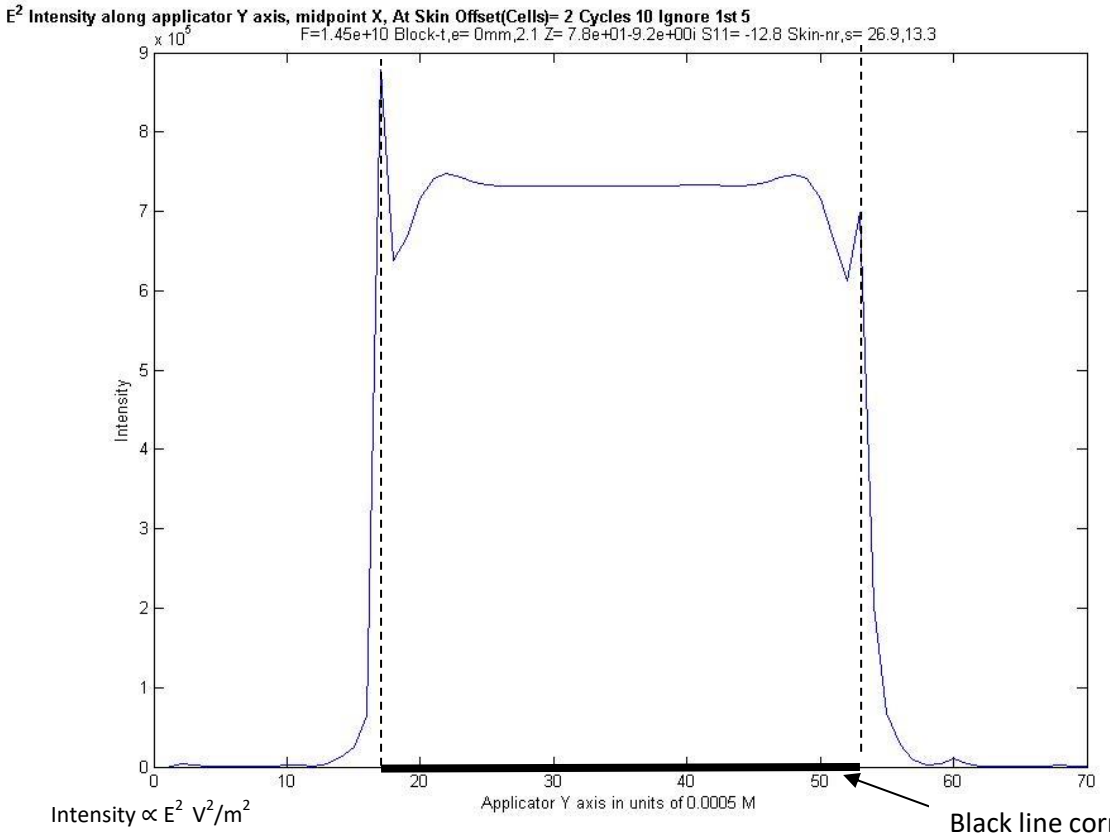


Figure 43 Y vs Electrical field Intensity at 1mm depth into skin for MSA and midpoint X - no modification.

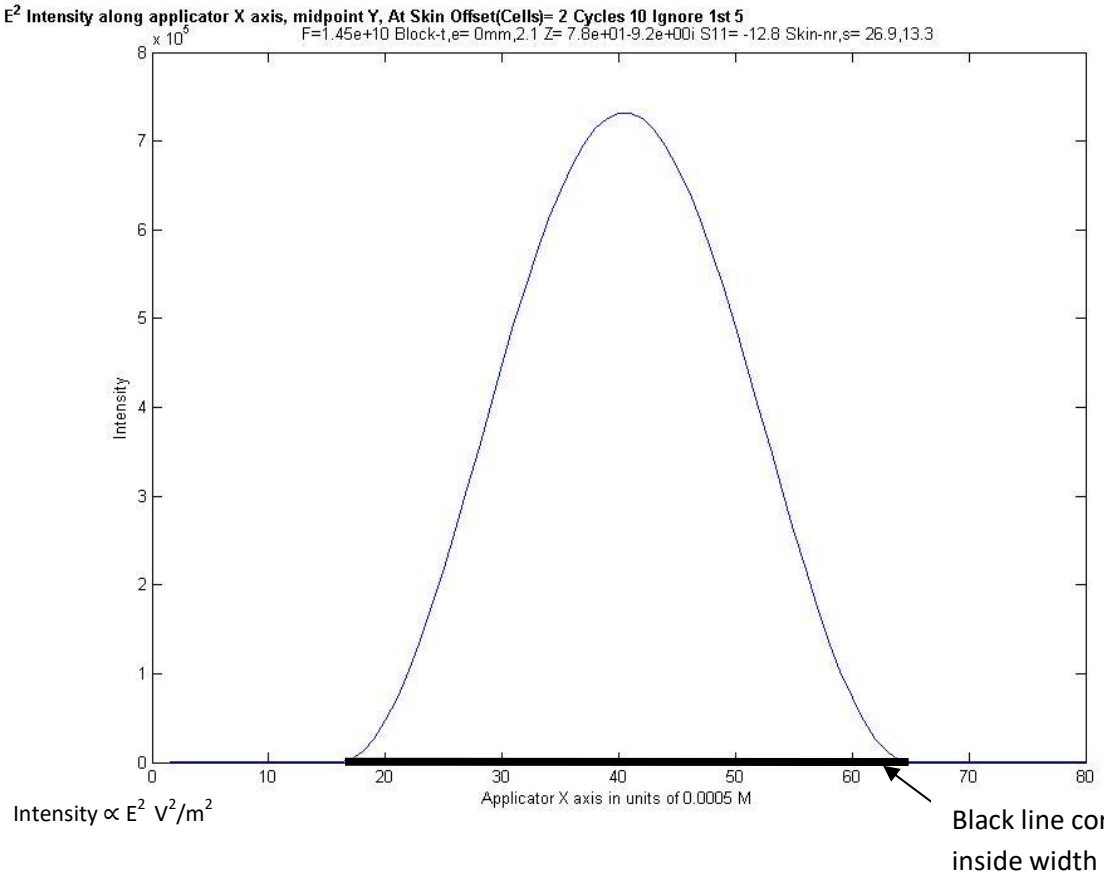


Figure 44 X vs Electrical field Intensity at 1mm depth into skin for MSA and midpoint X - no modification

Results – PTFE side blocks

The following plots were obtained by running the program twice with different parameters. Each run comprises 6 sub-runs each with stepped side block thickness. The first run varies the side block thickness from 0-5mm with 1mm step and the second from 3.5-6mm with a 0.5mm step. The first shows little variation in intensity across the aperture with the flattest profile achieved for 5mm side blocks. The second run provides further exploration of the intensity profile by utilising 0.5mm steps.

The first run was performed with the following parameters:

Block_Min = 0	Minimum side block thickness
Side_Blocks_PTFE = true	Enable side blocks with no loss and same magnetic characteristics as a vacuum.
Block_re = 2.1	Set side block relative permittivity
Block_Steps = 6	Perform 6 sub-runs
Block_Thickness = Block_Min+(0.001 * (ii-1))	Calculate side block Thickness for each run, note that 'ii' varies from 1 to Block_Steps

This code results in 6 separate sub-runs, with the side block thickness incremented by 1mm each time starting from 0mm. The resultant graph was manualy rotated within Matlab to produce the X and Y views. This is shown in Figure 45, Figure 47 and Figure 46 below.

The second run was performed with the following parameters:

Block_Min = 0.0035	Minimum side block thickness in metres
Side_Blocks_PTFE = true	Enable side blocks with no loss and same magnetic characteristics as a vacuum.
Block_re = 2.1	Set side block relative permittivity
Block_Steps = 6	Perform 6 sub-runs
Block_Thickness = Block_Min+(0.0005 * (ii-1))	Calculate side block Thickness for each run, note that 'ii' varies from 1 to Block_Steps

This code results in 6 separate sub-runs, with the side block thickness incremented by 0.5mm each time starting at 3.5mm. The resultant graph was manualy rotated within Matlab to produce the X and Y views. This is shown in Figure 48, Figure 49 and Figure 50 below.

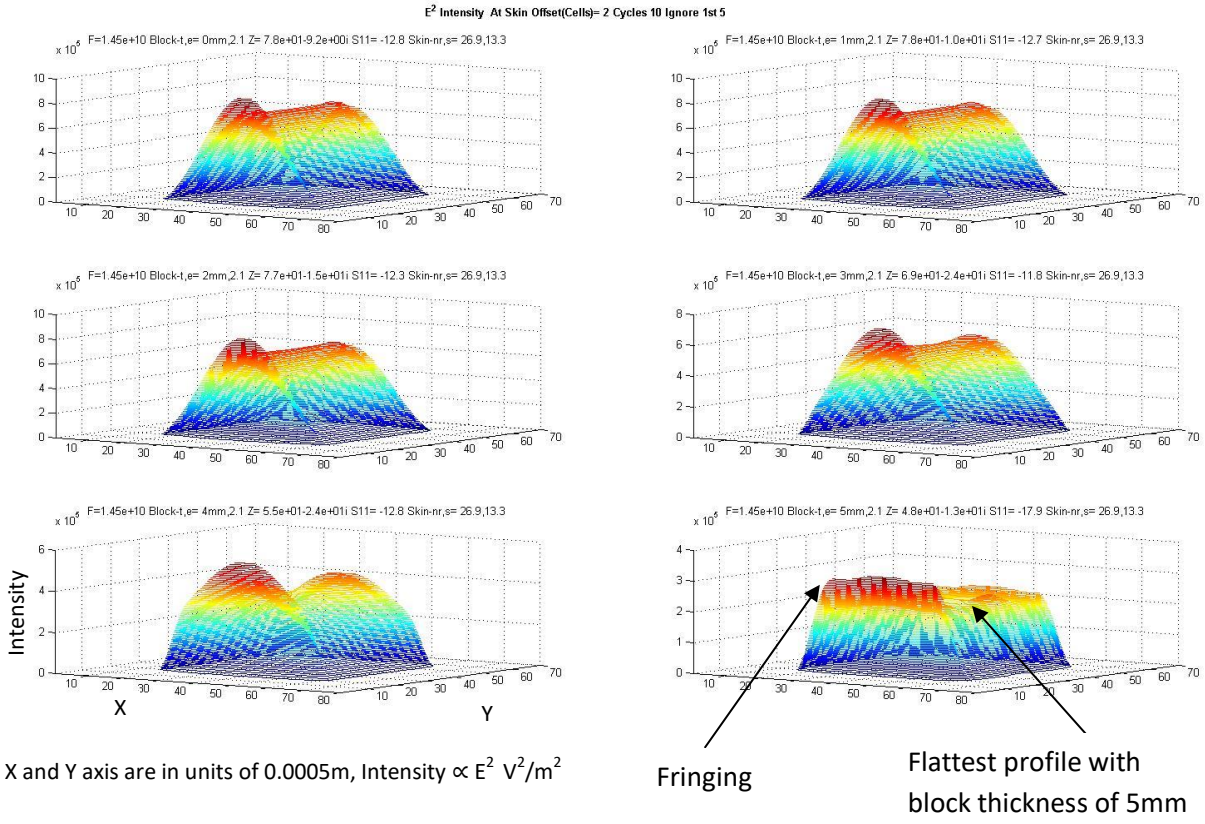


Figure 45 E field intensity at 2 cells (1mm) within human skin for side block thickness of 0-5mm. XY across horizontal axis, Intensity on Y axis. Note fringing effect on top and bottom of the aperture and flattest profile corresponding to block thickness

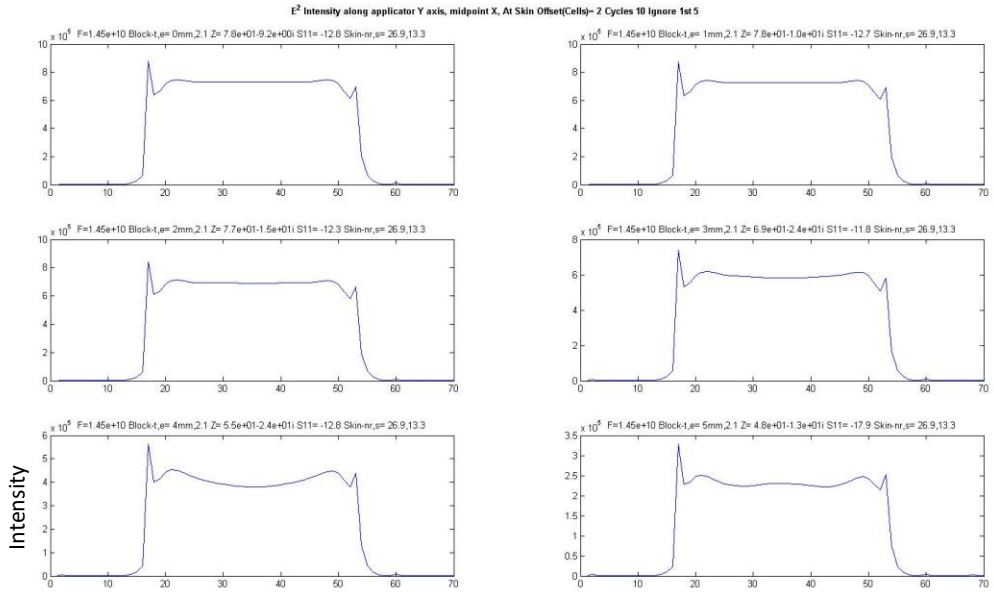


Figure 46 E field intensity at 2 cells within human skin for side block thickness of 0-5mm. Aperture Y across graph horizontal axis, Intensity on vertical axis.

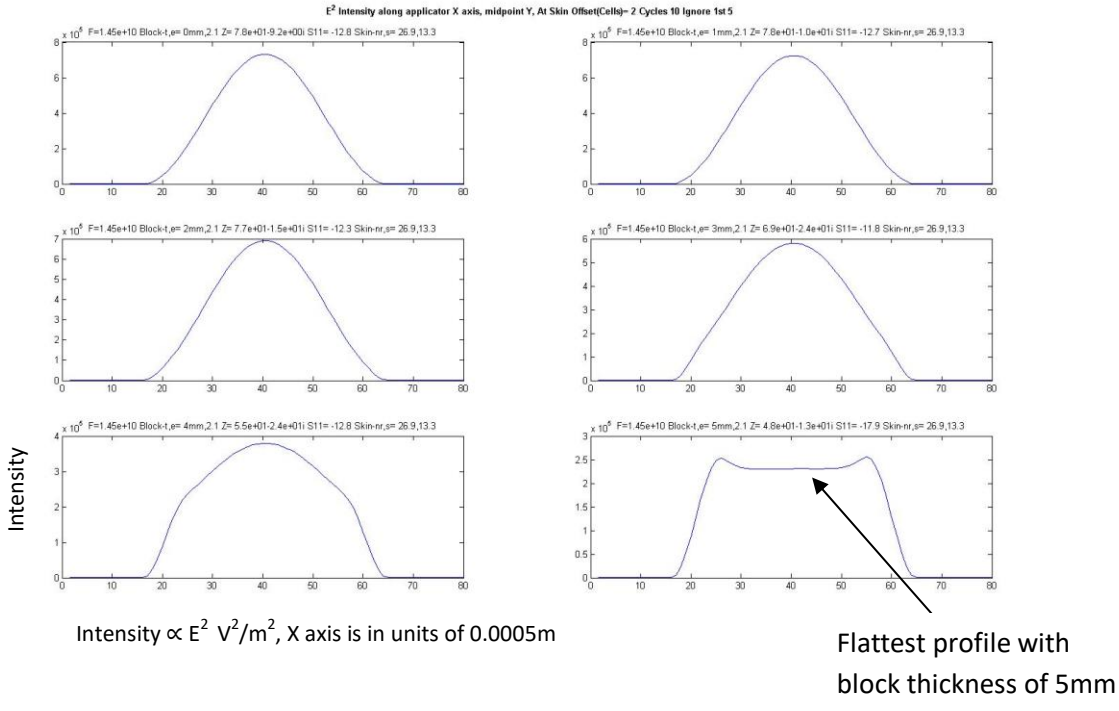


Figure 47 E field intensity at 2 cells within human skin for side block thickness of 0-5mm. Aperture X across graph horizontal axis, Intensity on vertical axis.

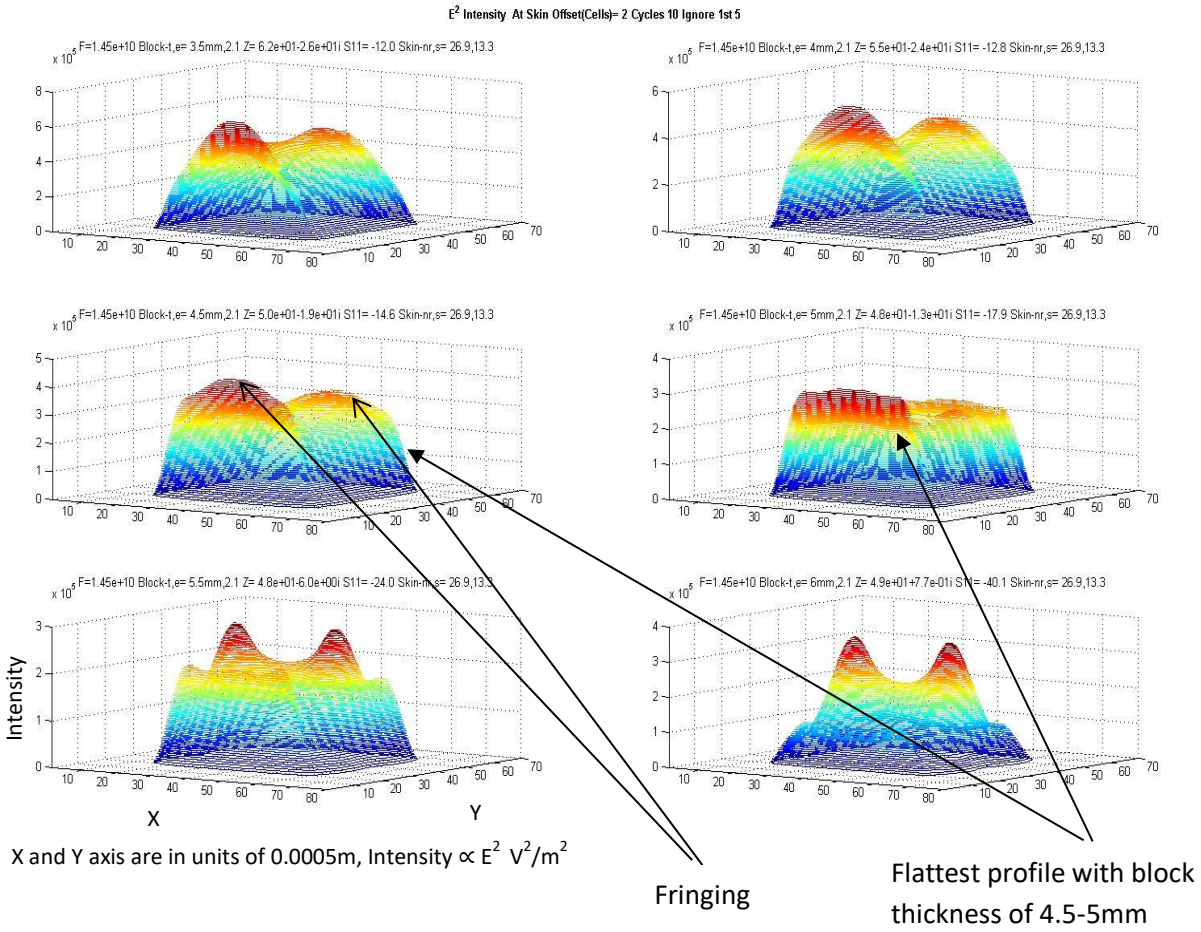
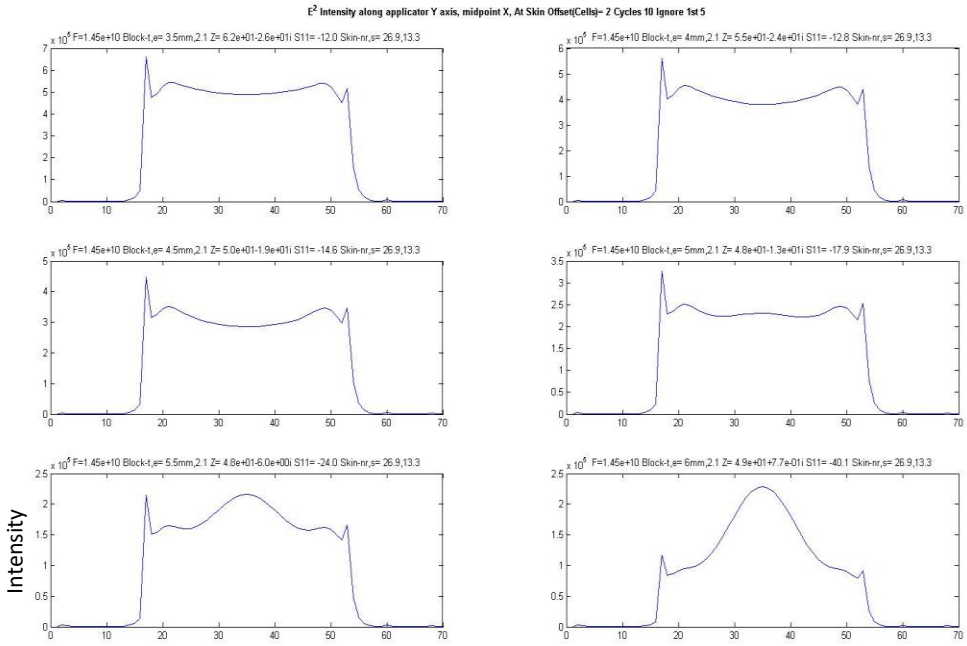
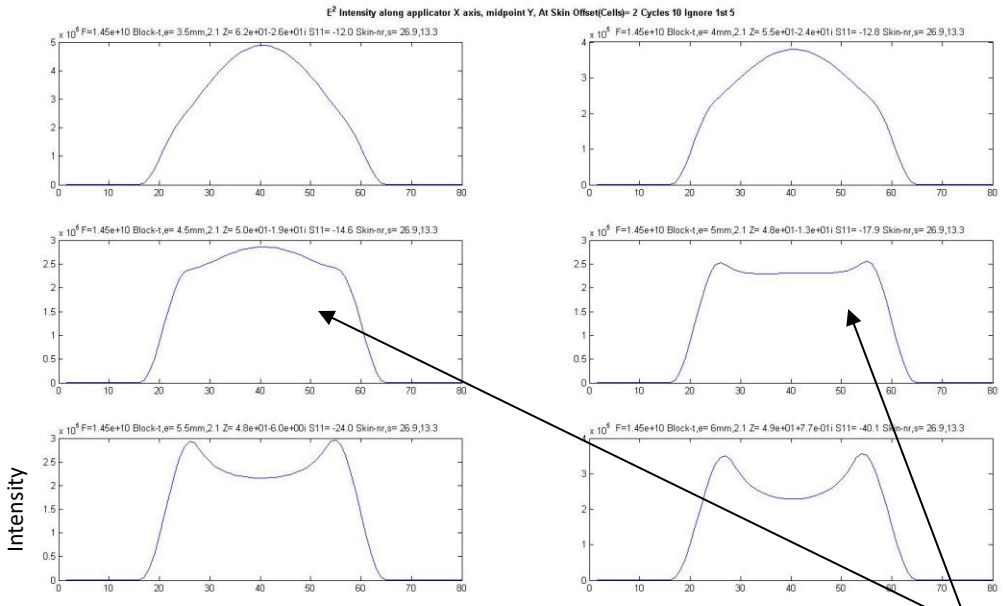


Figure 48 E field intensity at 2 cells (1mm) within human skin for side block thickness of 3.5-6mm. XY across horizontal axis, Intensity on Y axis. Note fringing effect on top and bottom of the aperture.



Intensity $\propto E^2$ V^2/m^2, Y axis is in units of 0.0005m

Figure 49 E field intensity at 2 cells within human skin for side block thickness of 3.5-6mm. aperture Y across graph horizontal axis, Intensity on vertical axis.



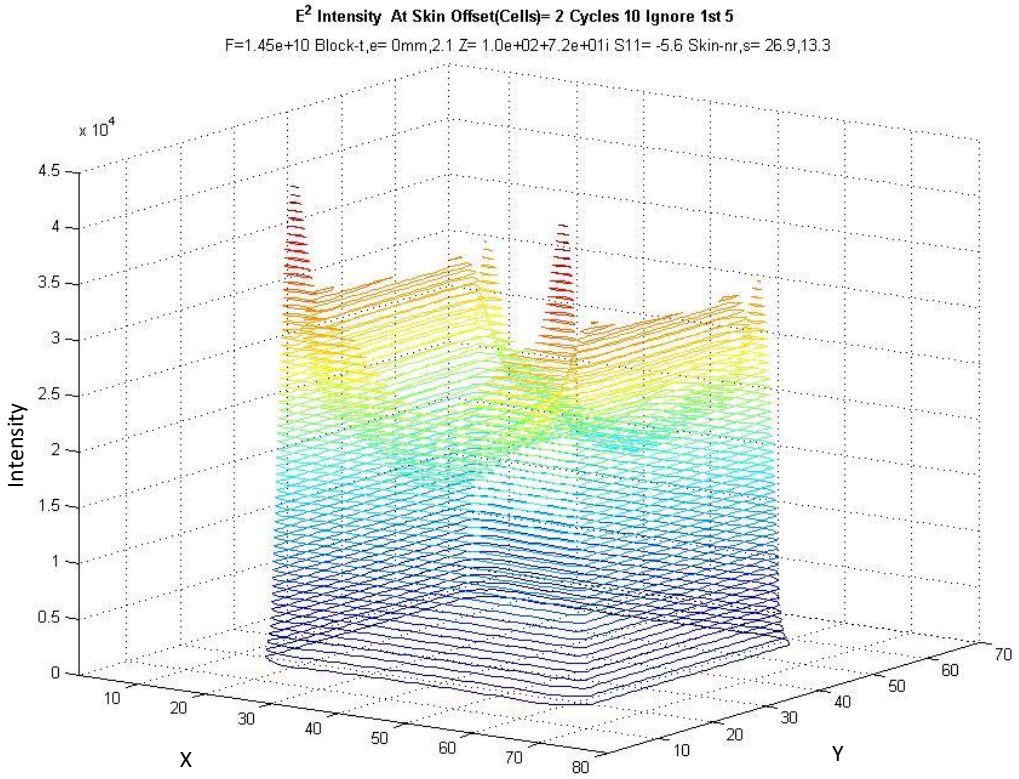
Intensity $\propto E^2$ V^2/m^2, X axis is in units of 0.0005m

Flattest profile with block thickness of 4.5- 5mm

Figure 50 E field intensity at 2 cells (1mm) within human skin for side block thickness of 3.5-5mm. aperture X across graph horizontal axis, Intensity on vertical axis.

Results – PMC side blocks

The following plots Figure 51, Figure 52 and Figure 53, were obtained with Simple_Side_Blocks_PMC = true. The electrical intensity can be seen to reach a maximum at the sides of the aperture, rather than a minimum that would occur with a Perfect Electrical Conductor (PEC). The Intensity is not flat along the X axis however as would be expected from a waveguide – This may be due to length and single monopole excitation in the waveguide. The monopole excitation voltage and current are shown in Figure 54 and Figure 55 respectively.



X and Y axis are in units of 0.0005m, Intensity $\propto E^2$ V^2/m^2

Figure 51 XY vs Electrical field Intensity at 1mm depth into skin for MSA with PMC sides.

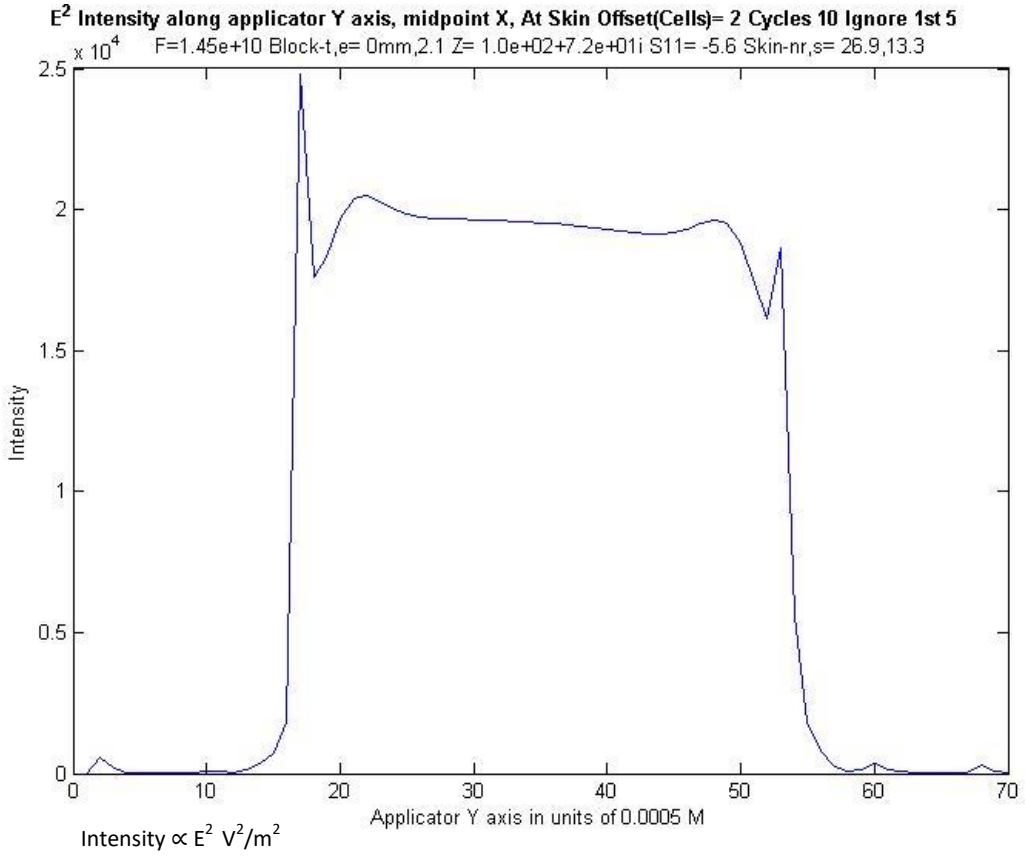


Figure 52 Y vs Electrical field Intensity at 1mm depth into skin for MSA with PMC sides.

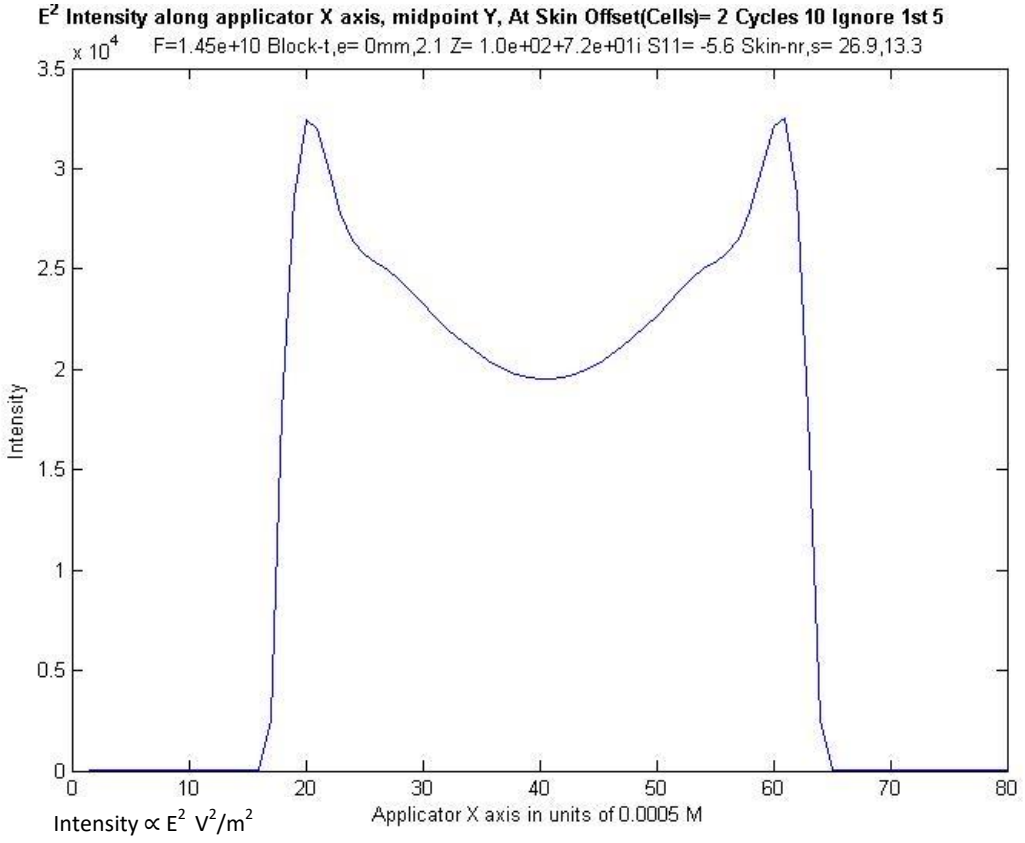


Figure 53 X vs Electrical field Intensity at 1mm depth into skin for MSA with PMC sides.

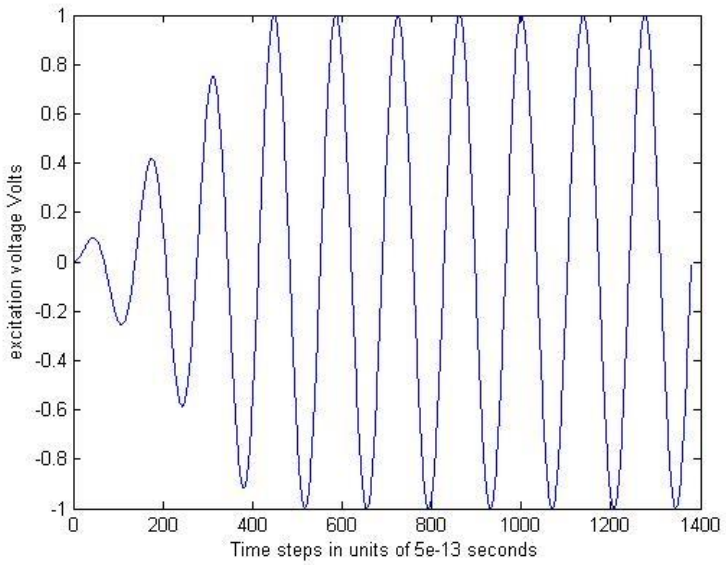


Figure 54 Monopole excitation voltage

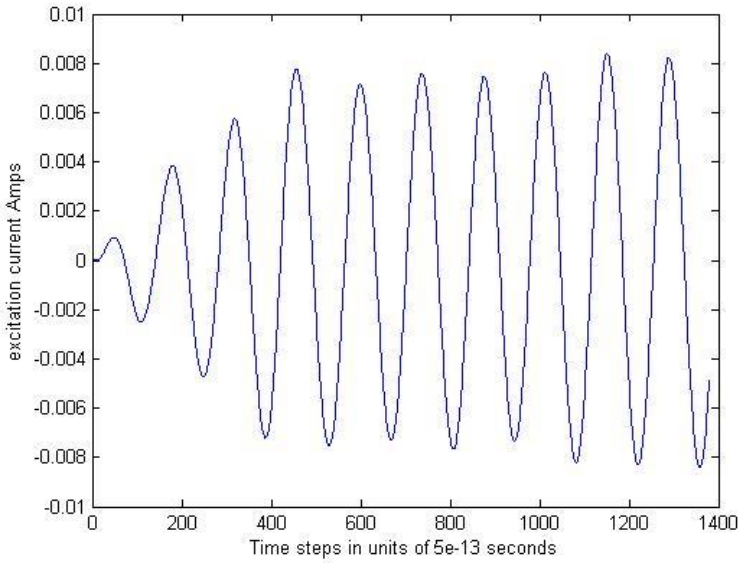
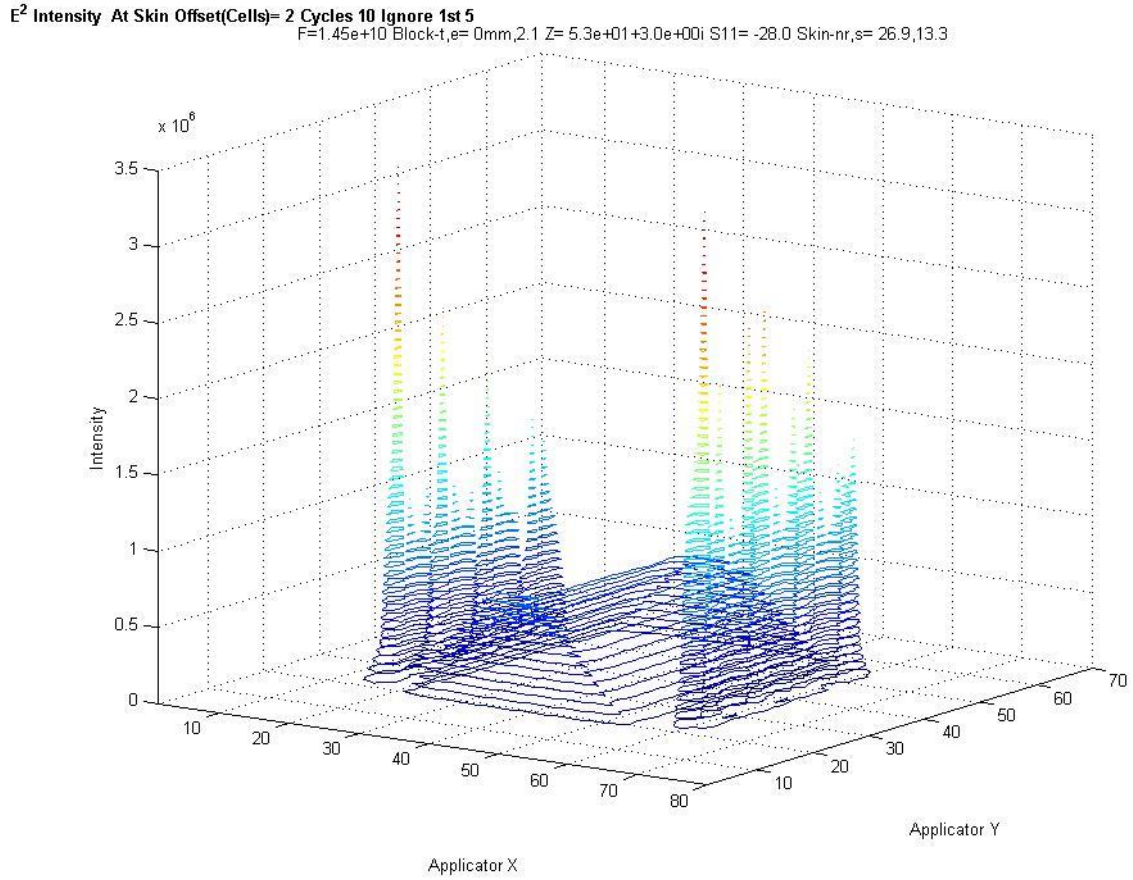


Figure 55 Monopole excitation current

Results – Meta Side Blocks

The following plots Figure 56 - Figure 58 were obtained with Side_Blocks_META = true. The intensity profile for the given configuration does not provide a flat intensity profile with the parameters used, This could also be the result of the coarse gridding or the implementation.



X and Y axis are in units of 0.0005m, Intensity $\propto E^2$ V^2/m^2

Figure 56 XY vs Electrical field Intensity at 1mm depth into skin for MSA with META sides.

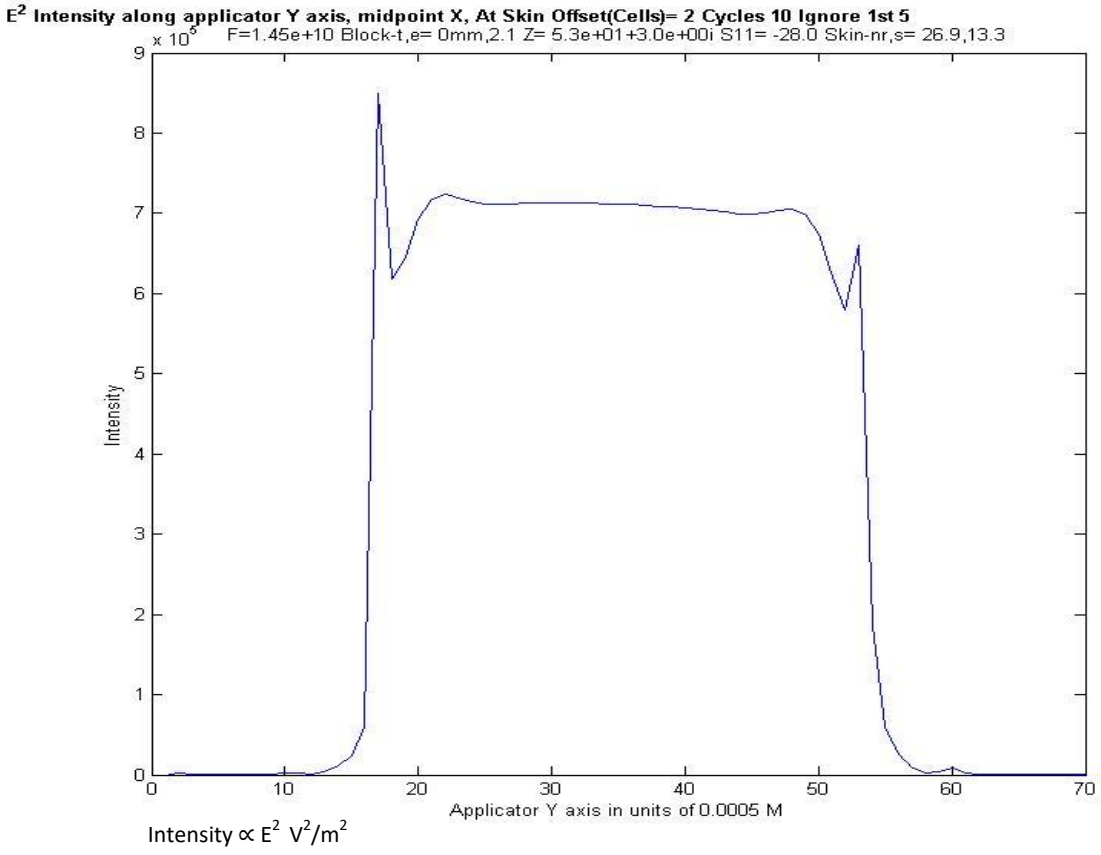


Figure 57 . Y vs Electrical field Intensity at 1mm depth into skin for MSA with META sides.

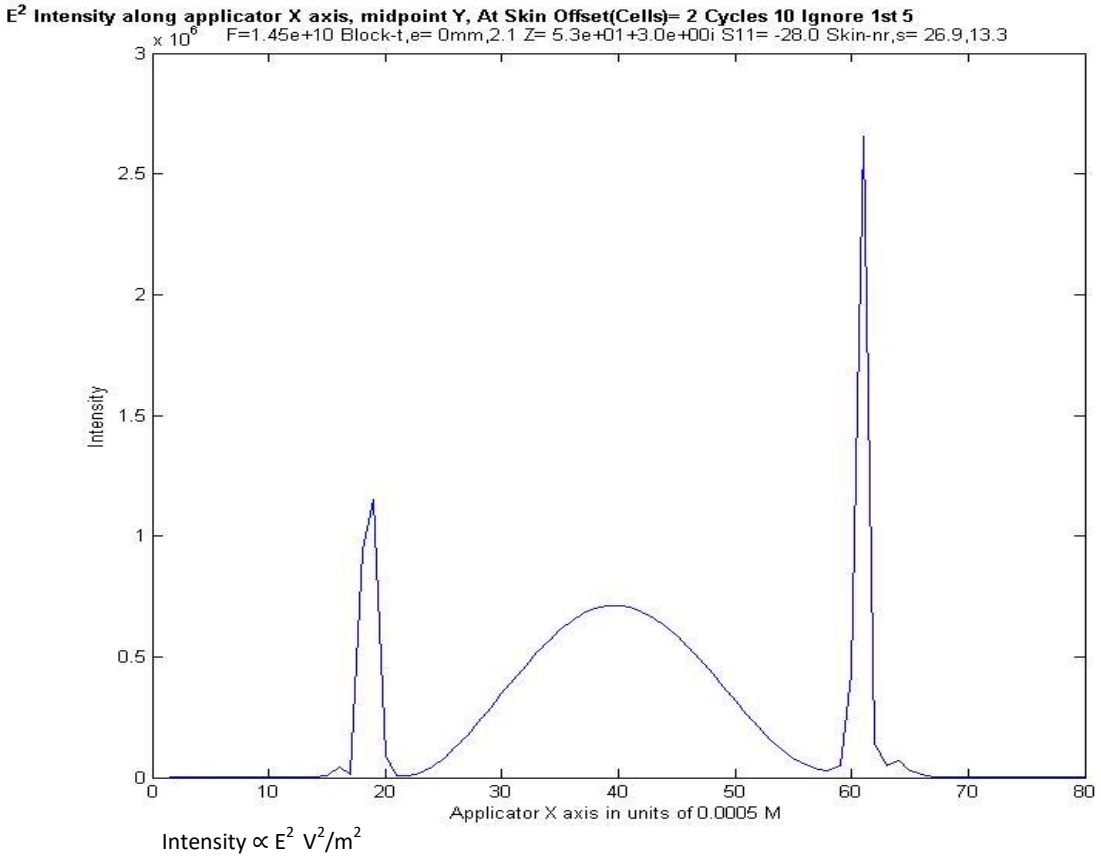


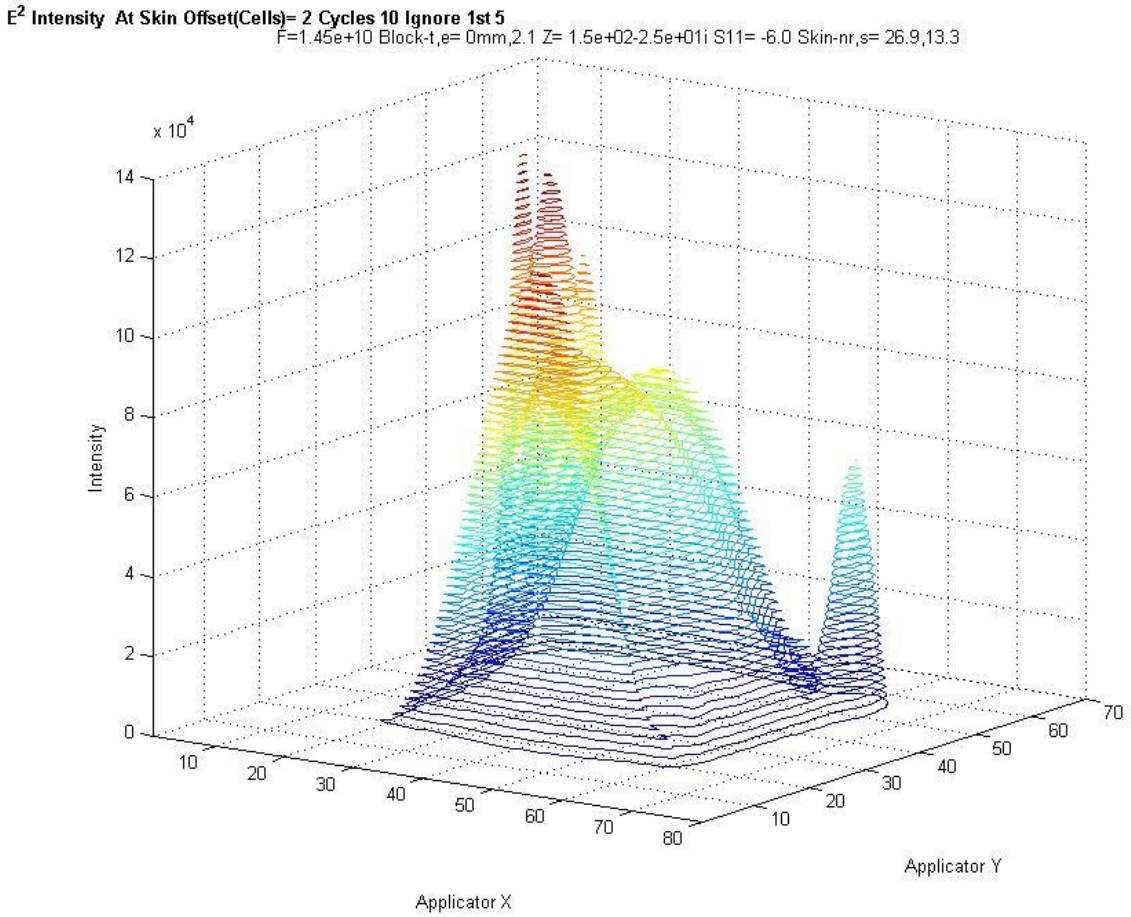
Figure 58 X vs Electrical field Intensity at 1mm depth into skin for MSA with META sides.

Results – Ferrite Side Blocks

The following plots Figure 59, Figure 60 and Figure 61 were obtained with Ferrite_Side_Blocks_PMC=true. Also, B_0 = 0.45T (magnetic field strength), E_r = 10 (relative permittivity) , U_r = 2 (relative permeability), A_Loss = 0 (loss factor). In addition, Figure 62 and Figure 63 show the ferrite relative permeability vs frequency and magnetic field respectively (the appropriate formulae from [5] pg 284. – these graphs are shown as a guide to choosing the field B_0 in order to obtain the appropriate rotation of the wave magnetic field in order to transform the PEC to a PMC at the side.

With the above parameters constant and varying B_0 , below approximately 0.2T produces similar results to the unmodified MSA. Various other simulations were performed above 0.2T and around the peak effects near 0.52T (see Figure 63) but no useful results obtained. The results at B_0 = 0.45T are included as an example.

As is clear from the graphs, the intensity profile does not approximate a flat profile, due either to the actual implementation or the parameters used.



X and Y axis are in units of 0.0005m, Intensity $\propto E^2$ V^2/m^2

Figure 59 XY vs Electrical field Intensity at 1mm depth into skin for MSA with Ferrite side blocks.

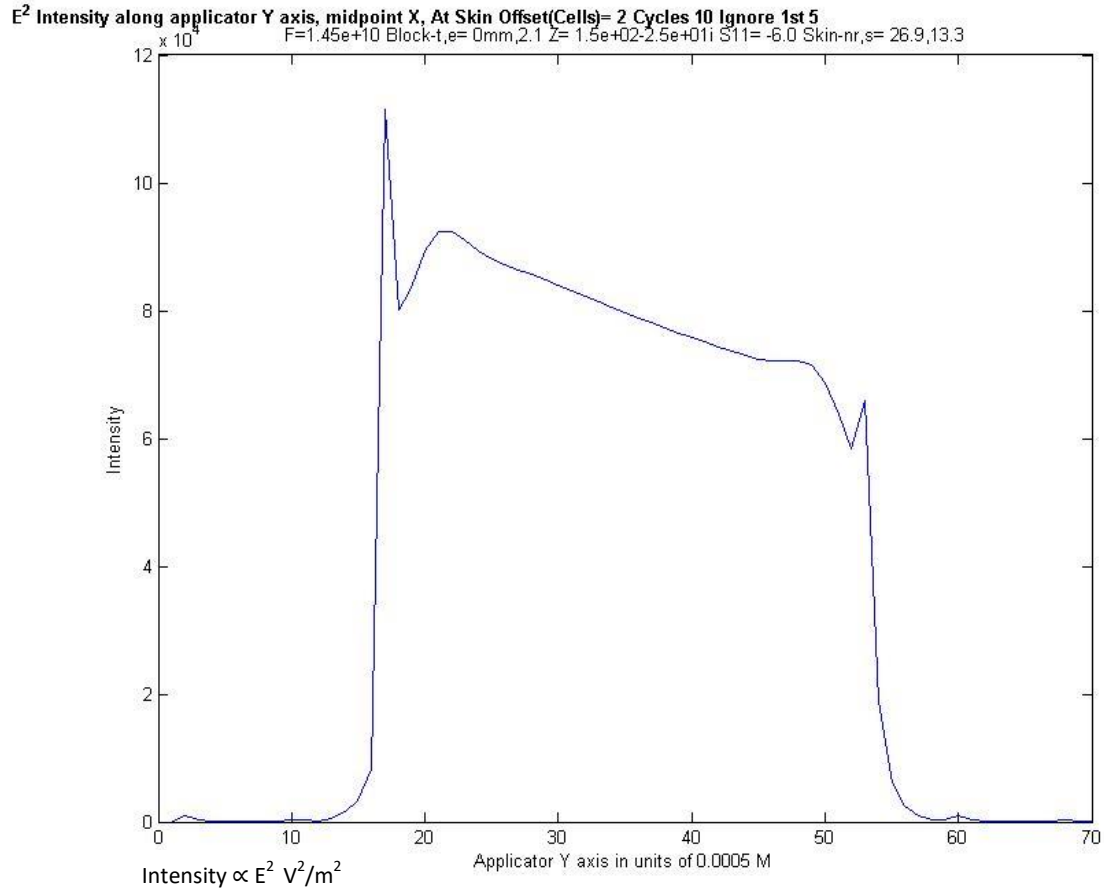


Figure 60 Y vs Electrical field Intensity at 1mm depth into skin for MSA with Ferrite side blocks

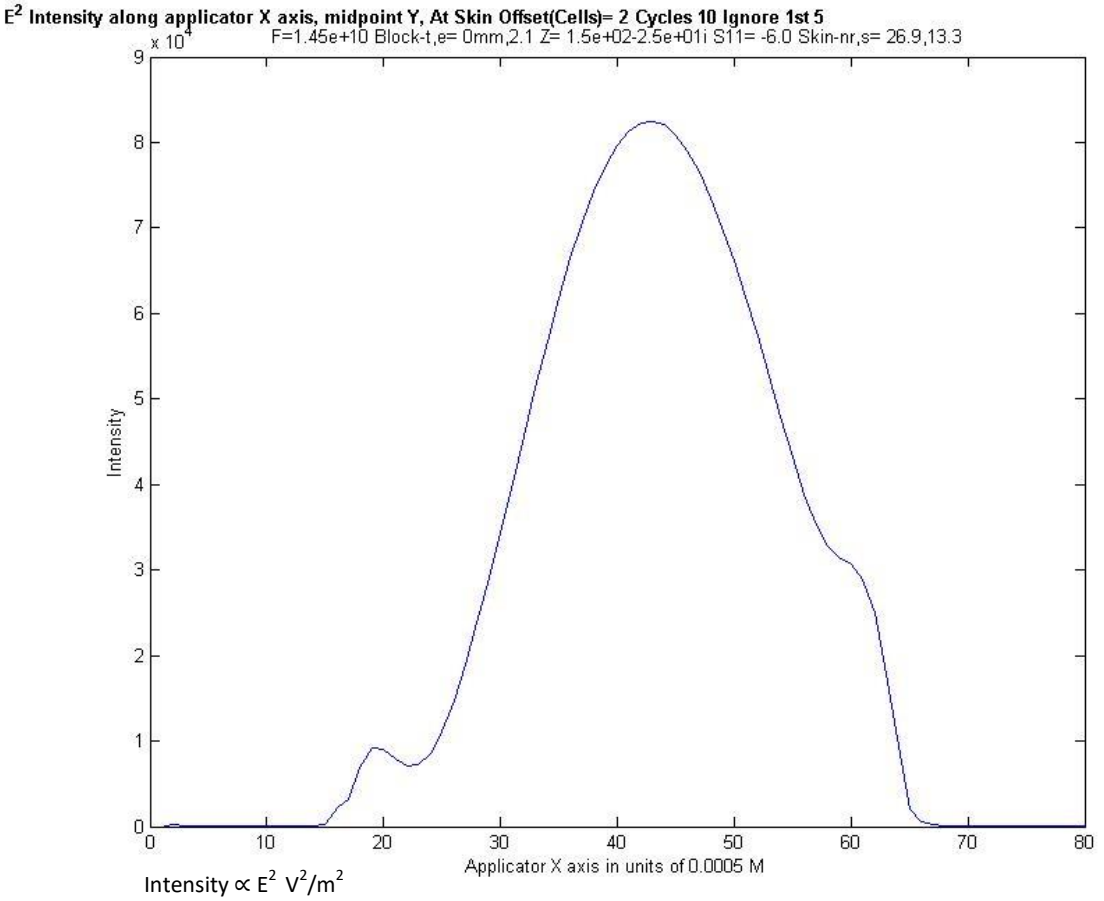


Figure 61 X vs Electrical field Intensity at 1mm depth into skin for MSA with Ferrite side blocks.

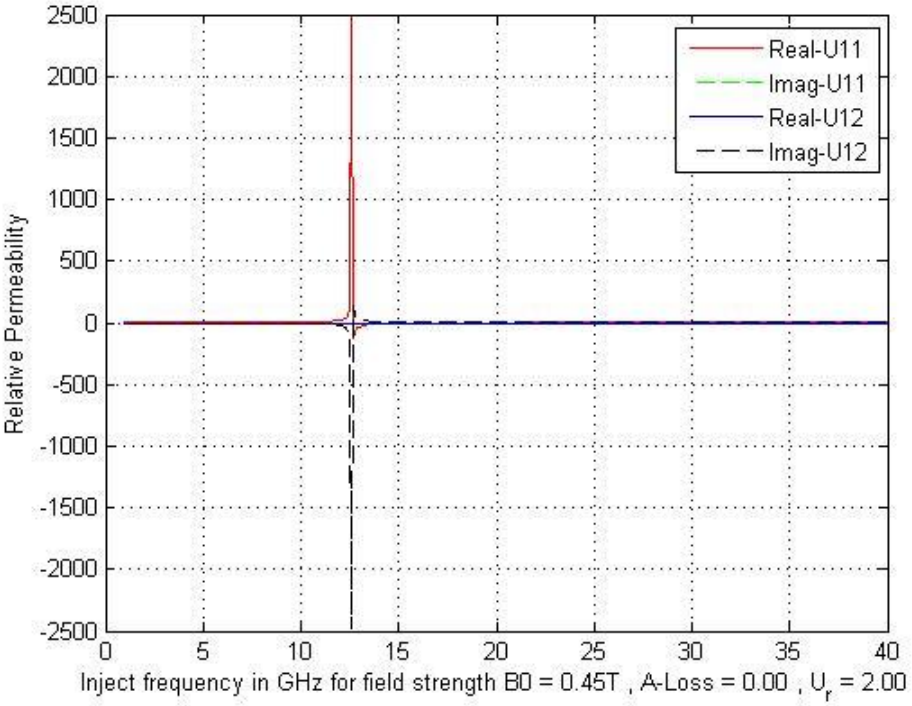


Figure 62 Ferrite Real and Imaginary relative permeability vs frequency

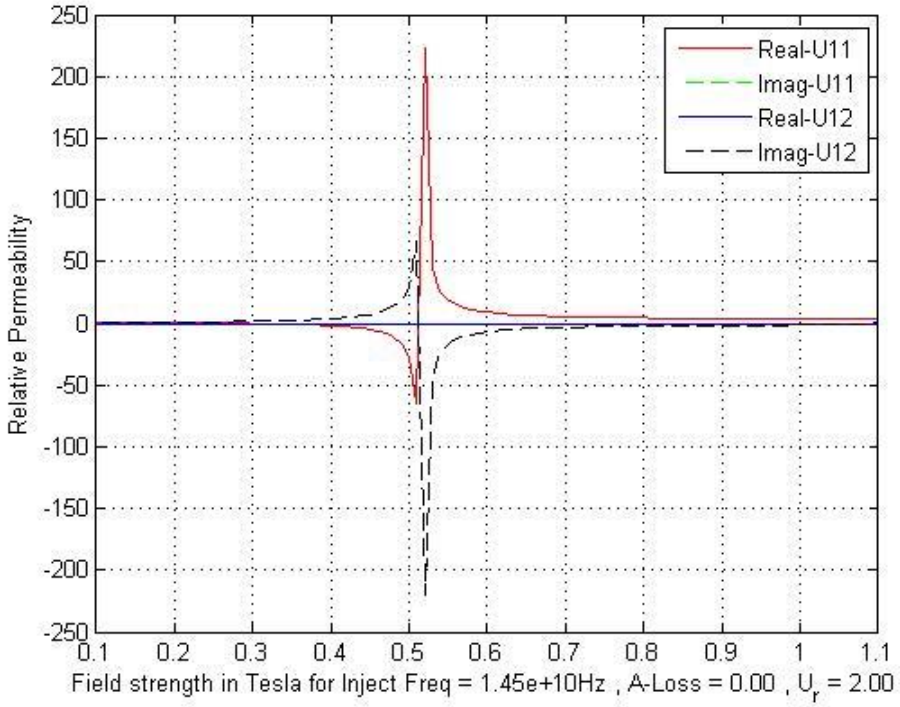


Figure 63 Ferrite Real and Imaginary relative permeability vs B0

Discussion of results

Figure 64 Below shows the simulation of E-field intensity along the X axis at a depth of 1mm into human skin at the mouth of the aperture. The intensity along the Y axis is approximately constant except for the fringing effect along the flange – refer to the results section for the appropriate graphs. Note that the intensity shown is a simple accumulation of total E^2 summed every time step and is useful for demonstrating the profile only.

The E-field intensity profile for META and ferrite types were not investigated further, mainly due to:

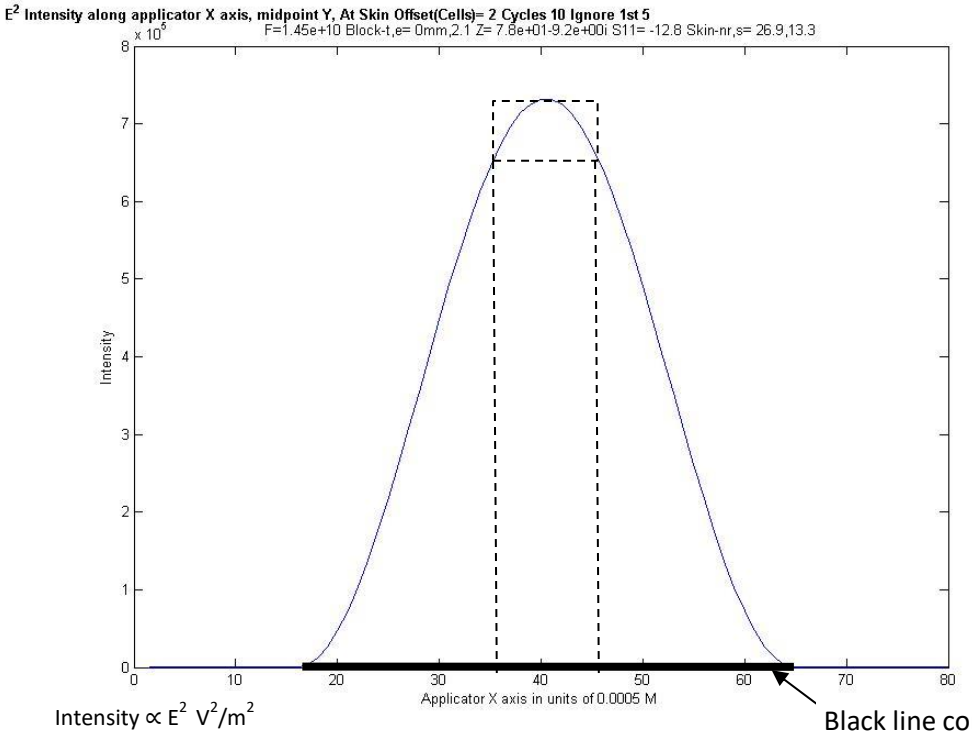
- The simplicity of using PTFE together with the acceptable profile obtained from the simulation.
- The practical difficulty in achieving a constant magnetic field with lightweight construction for use with ferrite. The ferrite simulation is somewhat asymmetrical possibly due to the location of the ferrite cells in relation to the PEC outer waveguide or the parameters used or the program implementation. This approach nevertheless may still be useful, especially as it has the advantage of dynamically tuning the PMC characteristics by varying the applied magnetic field [3].
- The results for META side blocks were not satisfactory with this simulation, possibly partly due to the coarse gridding (0.5mm) used for the simulation or the program implementation. Further, various other types of Meta material designs may be more appropriate.

Usable heating ablation area

With reference to the graphs in Figure 64 below, and setting a target band of +0% -10% at the maximum intensity in order to obtain an approximate usable heating profile width: The unmodified MSA provides a usable width of 10 cells whereas the MSA with 5mm PTFE side blocks provides a usable width of 32 cells, corresponding to 5 and 16mm respectively. Considering that the actual aperture width is 24mm, this is equivalent to 21% and 67% of the actual aperture width respectively. The intensity is approximately constant in the Y direction, so providing an ablation zone of 16W x 18Hmm

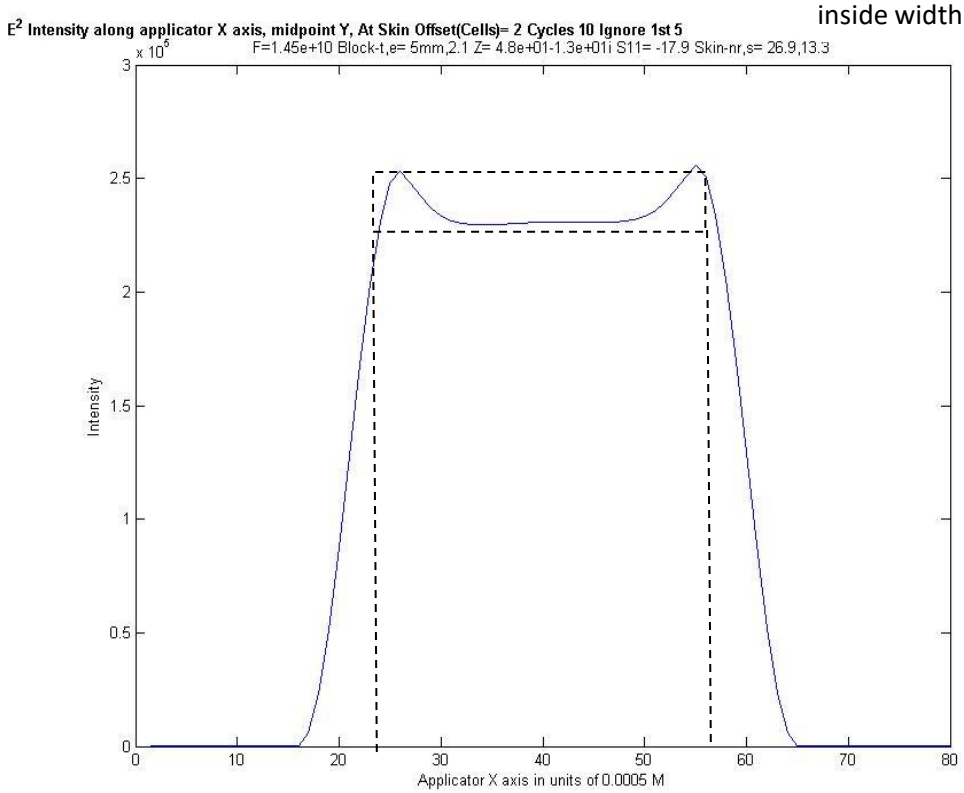
Unwanted heating outside of ablation area

Defining this as the area where the intensity rises from 10-90% of the maximum, with reference to Figure 64; the PTFE loaded MSA provides a zone approximately 3mm wide compared to the unloaded MSA's 7mm. As a very rough approximation of temperature rise which is solely due to microwave heating and not conduction (unrealistic but useful as a guide), with a maximum rise of 30 °C and a body temperature of 37 °C: the ablation zone would have a maximum temperature of 67 °C, while the adjacent zone would rise from 40 to 64 °C. In temperature simulations (as seen in Chapter 2), conduction plays a major role in depth of penetration and may further blur the heating zone. Nevertheless, the steeper the intensity profile rise across the X and Y axis, the more defined the actual heating profile.



Intensity $\propto E^2$ V^2/m^2

Black line corresponds to the inside width of MSA



Intensity $\propto E^2$ V^2/m^2

Figure 64 Comparison of E-field intensity along X axis of MSA without (top) and with (bottom) side blocks of 5mm PTFE.

Conclusion and Further Work

The simulated results indicate that simple 5mm thick side-blocks consisting of PTFE should be suitable to greatly improve the MSA's heating profile (i.e. uniformity) along the horizontal (x) axis of the resonant cavity.

Background research [3] indicates that this profile may be further improved by utilising side-blocks that approximate a perfect magnetic conductor (PMC). The Matlab code presented here would appear to require refinement such as subgridding in order to gain useful simulated results for meta-material structures. Further work on the ferrite simulation, such as increasing the ferrite thickness and proximity to the outer PEC walls in conjunction with further verification and if necessary, correction of the implementation should be sufficient to gain useful results.

In addition, the simulation system could be modified to provide a power profile useful for the Thermal Modeller described in Chapter 2 however, due to the long simulation times, subgridding and the simultaneous use of multiple CPU cores would be advantageous.

And as a comment on programming simulations and their results – these have proved invaluable to gaining an insight into Maxwell's equations in the context of numerical solutions as applied to waveguide structures with simple monopole excitation, lossy material (human skin) and anisotropic magnetic material (ferrite).

Works Cited

[1] C. Hancock, “Skin Cancer Treatment System, Medical Microwaves Systems Research Group,” Bangor University, Bangor, 2012.

[2] *The Mathsworks Inc, Matlab R2012A Student Edition,* Natick, MA, USA, 2012.

[3] A. Shahvarpour and C. Caloz, “Grounded Ferrite Perfect Magnetic Conductor and Application to Waveguide Minaturization,” in *Microwave Symposium Digest, 2009. MTT '09. IEEE MTT-S International, pgs 25-28*, 2009.

[4] K. S. Yee, “Numerical Solution of Initial Boundary Problems involving Maxwell's Equations in Isotropic Media,” *IEEE Antennas and Propagation Society,* vol. 14 , no. 3, 1966.

[5] U. S. Inan and R. A. Marshall, Numerical Electromagnetics The FDTD Method, Cambridge University Press, 2011.

[6] B. S. Guru and H. R. Hiziroglu, Electromagnetic Field Theory Fundamentals Second Edition, Cambridge: Cambridge University Press, 2004.

[7] A. Tavlove and S. C. Hagness, Computational Electrodynamics The Finite-Difference Time-Domain method third edition, Boston: Artech House, 2005.

[8] S.-C. Watanabe and M. Taki, “An Improved FDTD Model for the Feeding Gap of a Thin-Wire Antenna,” *IEE Microwave and Guided Wave Letters,* vol. 8, no. 4, pp. 152-154, 1998.

[9] R. Luebbers and H. Langdon, “A Simple Feed Model that Reduces Time Steps Needed for FDTD Antenna and Microstrip Calculations,” *IEEE Transactions on Antennas and Propagation,* vol. 44, no. 7, pp. 1000-1005, 1996.

[10] T. W. Hertel and G. S. Smith, “On the Convergence of Common FDTD feed models for Antennas,” *IEEE Transactions on Antennas and Propagation,* vol. 51, no. 8, pp. 1771-1779, 2003.

[11] R. Luebbers, L. Chen, T. Uno and S. Adachi, “FDTD Calculation of Radiation Patterns, Impedance and Gain for a Monopole Antenna on a Conducting Box,” *IEEE Transactions on Antennas and Propagation,* vol. 40, no. 12, pp. 1577-1583, 1992.

[12] M. Okoniewski, E. Okoniewska and M. A. Stuchly, “Three-Dimensional Subgridding Algorithm for FDTD,” *IEEE Transactions on Antennas and Propagation,* vol. 45, no. 3, pp. 422-429, 1997.

[13] M. W. Chevalier, R. J. Luebbers and V. P. Cable, “FDTD Local Grid with Material Traverse,” *IEEE Transactions on Antennas and Propagation,* vol. 45, no. 3, pp. 411-421, 1997.

[14] J. Thuery, MICROWAVES: Industrial, Scientific and Medical Applications, Artech House, 1992.

[15] C. Hancock, *Discussion about flattening E field,* Bangor , November, 2011.

[16] M. White, *Conversation,* Peasdown,Bath, 2013.

[17] D. Sievenpiper, L. Zhang, R. F. J. Broas, N. G. Alexopolous and E. Yablonovich, "High-Impedance Electromagnetic Surfaces with a Forbidden Frequency Band," *IEEE Transactions on Microwave Theory and Techniques,* vol. 47, no. 11, pp. 2059-2074, 1999.

[18] M. A. Haranandani, Y. A. B. and A. KishK, "Artificial Magnetic Conductors realised by frequency-selective surfaces on a grounded dielectric slab for antenna applications," *IEE Proceedings - Microwave Antennas Propagation,* vol. 153, no. 5, pp. 487-493, 2006.

[19] G. Goussetis, A. P. Feresidis and J. C. Vardaxoglou, "Tailoring the AMC and EBG Characterisitics of Periodic Metallic Arrays Printed on Grounded Dielectric Substrate," *IEEE Transactions on Antennas and Propagation,* vol. 54, no. 1, pp. 82-89, 2006.

[20] R. Kuse, T. Hori and M. Fujimoto, "PMC and EBG Characteristics of Cross Type Artificial Magnetic Conductor," in *ISAP2012*, Nagoya, 2012.

[21] Y. Kim, F. Yang and A. Elsherbeni, "Compact Artficial Magnetic Conductor Designs Using Planar Square Spiral Geometries," *Progress in Electromagnetics Research ,* vol. 77, pp. 43-54, 2007.

[22] *Heimann Sensor GmbH, Maria-Reiche-Str 1, 01109 Dresden, Germany: HTPA 8x8 L7.0A Thermopile Array.*

[23] U. S. Inan and A. S. Inan, Electromagnetic Waves, Upper Saddle River: Prentice Hall, 1999.

[24] *CST Microwave Studio,* CST GmbH, Bad Nauheimer Str. 19, 64289 Darmstadt, Germany.

[25] Wikipedia, "Dipole Antenna," [Online]. Available: https://en.wikipedia.org/wiki/Dipole_antenna#Impedance_of_dipoles_of_various_lengths. [Accessed 23 October 2016].

[26] Wikipedia, "Euler–Mascheroni constant," [Online]. Available: https://en.wikipedia.org/wiki/Euler%E2%80%93Mascheroni_constant. [Accessed 23 October 2016].

[27] Wikipedia, "Discrete Fourier transform," [Online]. Available: https://en.wikipedia.org/wiki/Discrete_Fourier_transform. [Accessed 23 Oct 2016].

[28] Maxim, "Converting S-Parameters from 50Ω to 75Ω Impedance," [Online]. Available: https://www.maximintegrated.com/en/app-notes/index.mvp/id/2866. [Accessed 24 Oct 2016].

Chapter 4 New Microwave Skin Applicator (NMSA) with Thermal Feedback

Contents

Introduction

The existing Microwave Skin Applicator (MSA) was revised and initial experiment work was performed to check overall operation. The system comprised the New Microwave Skin Applicator (NMSA) fitted with a Heimann Thermal Camera [1], a 14.5GHz energy source and a computer with Matlab [2] installed. The Matlab program was designed to utilise the Thermal Camera's data stream to control the 14.5GHz energy source in order to maintain a constant ablation temperature.

Design of the New Microwave Skin Applicator (NMSA)

Based on the results of Chapter 3 a 14.5GHz resonant applicator was designed using CST Microwave studio [3] with PTFE side blocks. CST was used to obtain the following design goals:

Design goals:

This work was divided into several sections as follows:

1. Single cavity construction for simpler fabrication.
2. Resonance with Thermal camera tube at rear of cavity.
3. Steep rise in power dissipation profile at edges of aperture.
4. Tuning screw to optimise resonance.
5. Smaller aperture so that lower power source could be used.
6. Control of fringing effect on top and bottom of aperture.

NMSA Design

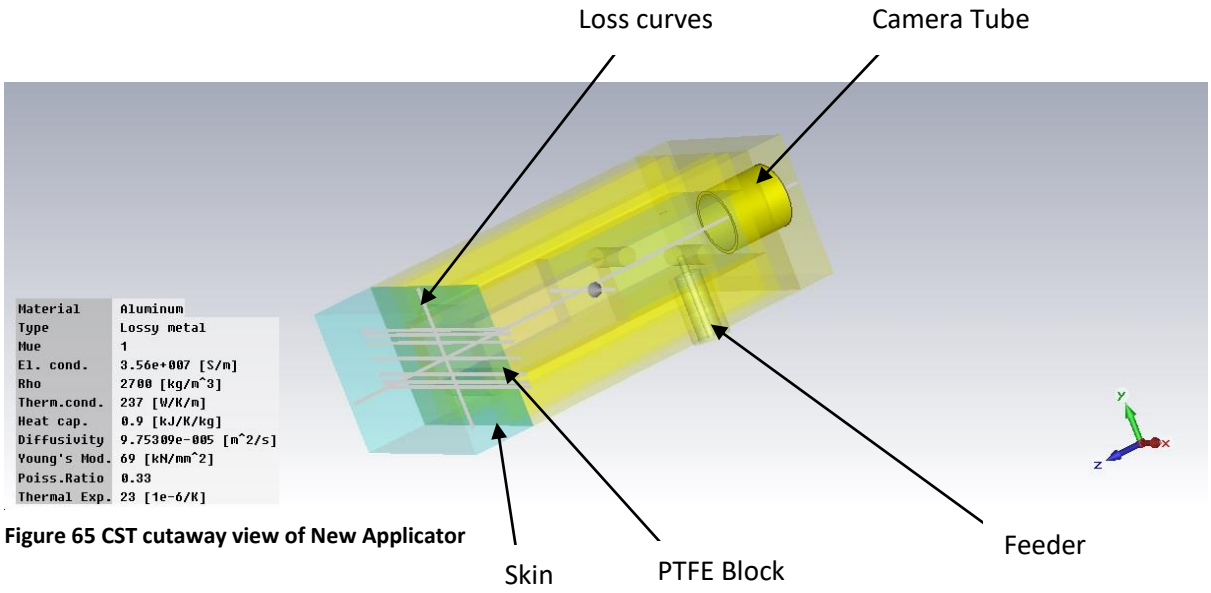


Figure 65 CST cutaway view of New Applicator

A 3D view of the design is shown by Figure 65. The Camera tube is on the right, feeder at the bottom right and Skin on the left. The Vertical and Horizontal lines in the skin are CST curves that are used to collect loss profiles .The component parts are shown in Appendix 3B[8] and the drawings in Figure 66 and Figure 67. Note that the design includes a 0.1mm film of Mylar on the front and grounded 'reflectors' that cover the PTFE side blocks – the 'reflectors' are shown on Figure 74 page 104 but not shown on Figure 65, Figure 66, Figure 67 or other photographs. The reflectors are intended to prevent heating directly in front of the PTFE side blocks where the E field intensity decreases with proximity to the side of the cavity. In addition, note that the thermal camera itself requires a 360° electrical connection to the camera tube which is achieved by several turns of copper tape around the camera body.

[8] **Disclaimer:** The design detailed here is included for documentation purposes only and has not been verified as to its correct basis, operation, results, safety, spurious or other emissions or fitness for any purpose. Any person or third party user of the design does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

Drawing of the NMSA[9]

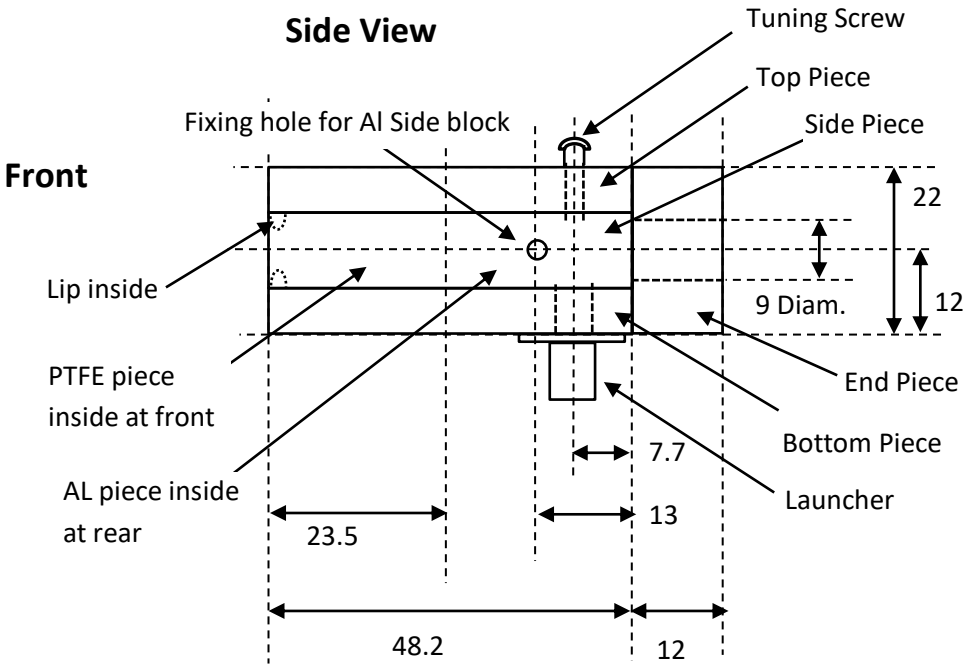
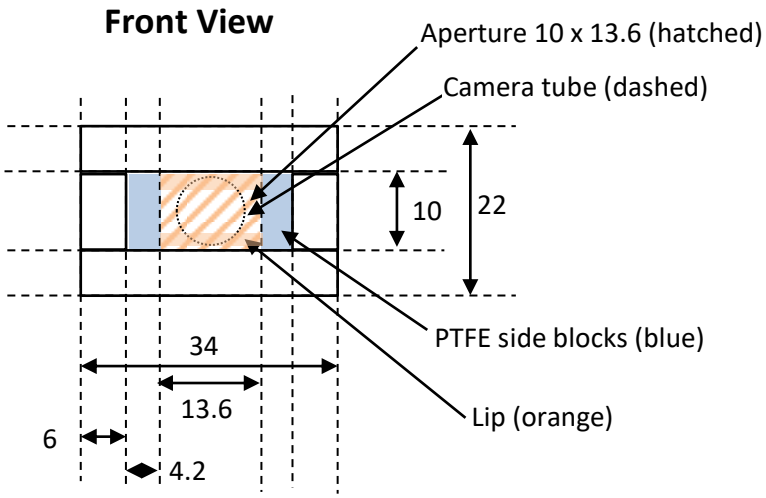


Figure 66 NMSA Side View



Note: Aluminium side pieces located behind PTFE. Reflector foil over PTFE not shown. All dimensions in mm. Fixing screws not shown.

Figure 67 NMSA Front View

[9] **Disclaimer:** The design detailed here is included for documentation purposes only and has not been verified as to its correct basis, operation, results, safety, spurious or other emissions or fitness for any purpose. Any person or third party user of the design does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

Pictures of the NMSA

These are shown below in Figure 68 to Figure 73, Note that the reflectors and Mylar cover are not fitted. The camera body is wrapped with several turns of copper tape to achieve a 360° electrical connection to the camera tube. The copper tape although not part of the tube itself provides, together with the camera's conductive housing can , an EM seal which greatly reduces the escape of microwave energy from the rear of the NMSA.

Rear and Underside showing camera mount

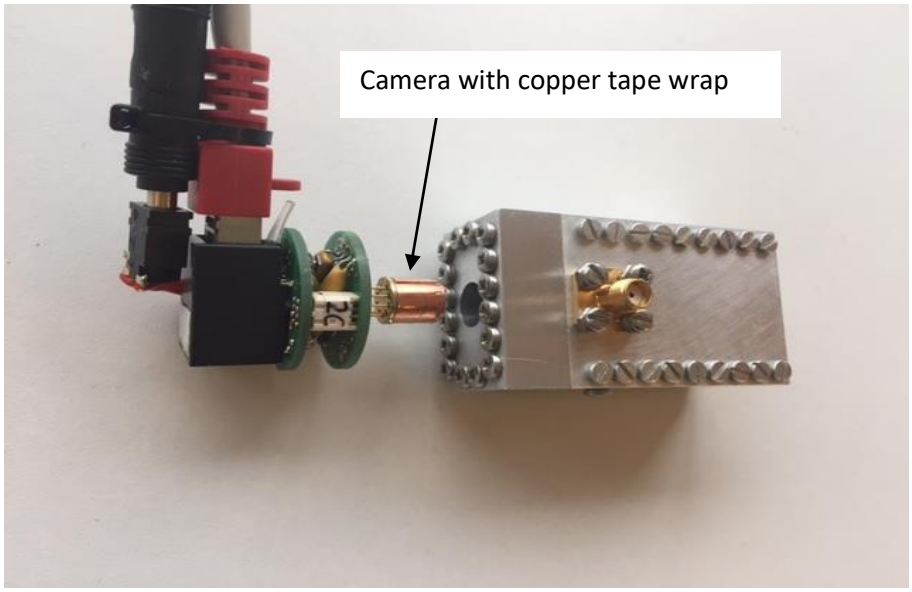


Figure 68 Rear and Underside with Camera demounted

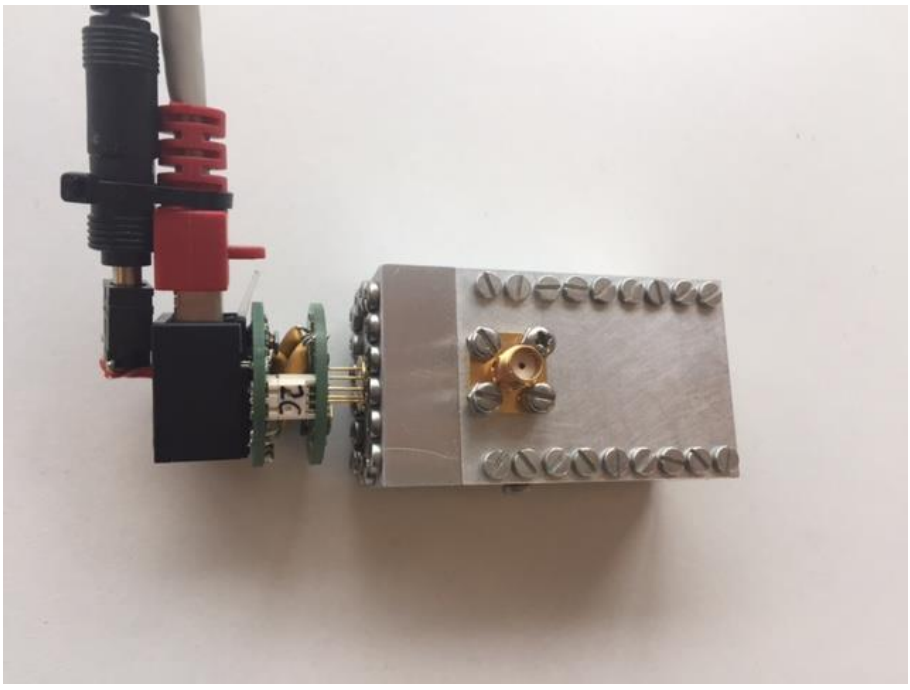


Figure 69 Rear and Underside with Camera mounted

View through Camera tube

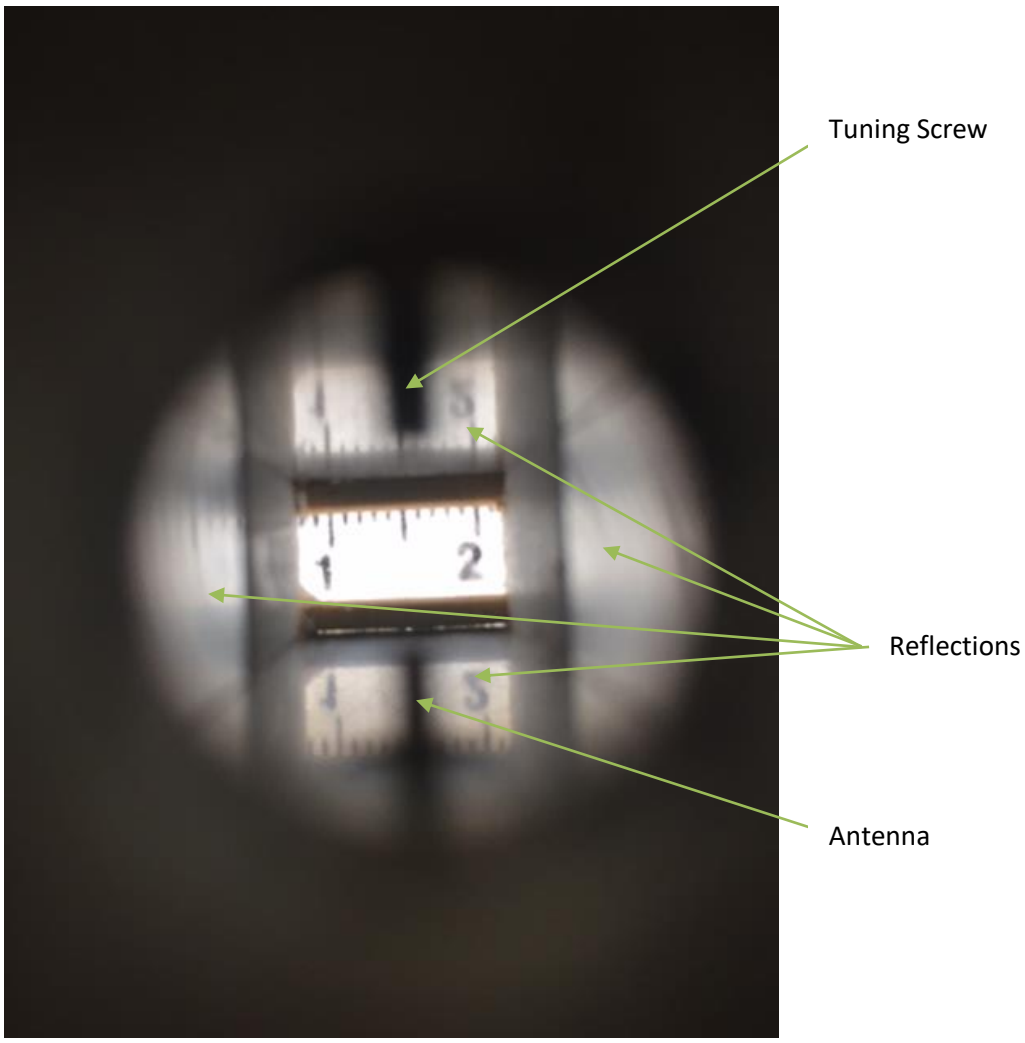


Figure 70 View of target through the Camera Tube

This picture was taken from just outside the Applicator through the camera tube. A photocopy of a millimetre ruler was placed against the aperture. The aperture is visible apparently unobscured by either the antenna (SMA launcher) or tuning screw and reflections on the top, bottom and sides can be seen clearly. The reflections also seem to appear on the thermal image shown later.

Front Top, Front Underside and Side Top views

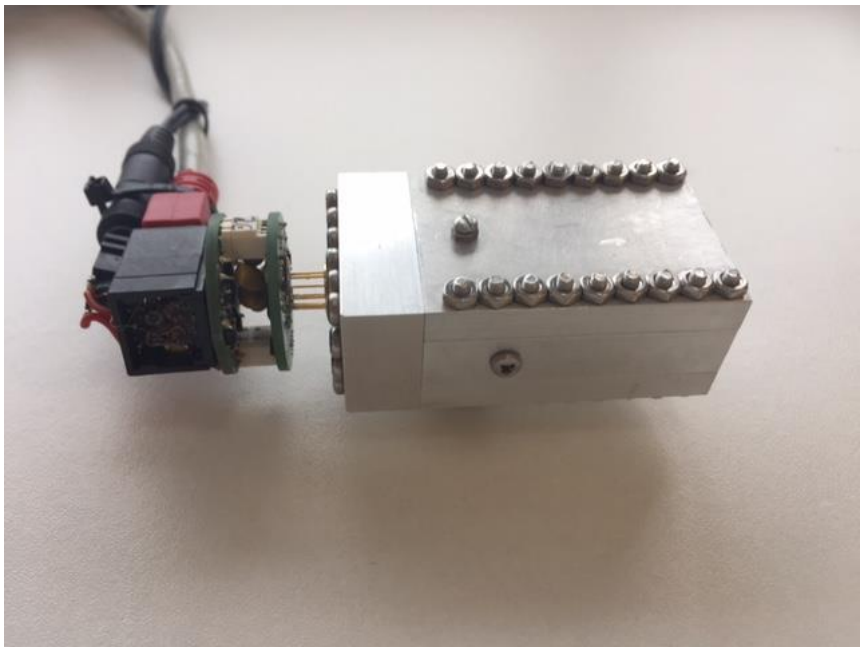


Figure 71 Picture of side showing Al side block mounting screw and Top showing tuning screw.

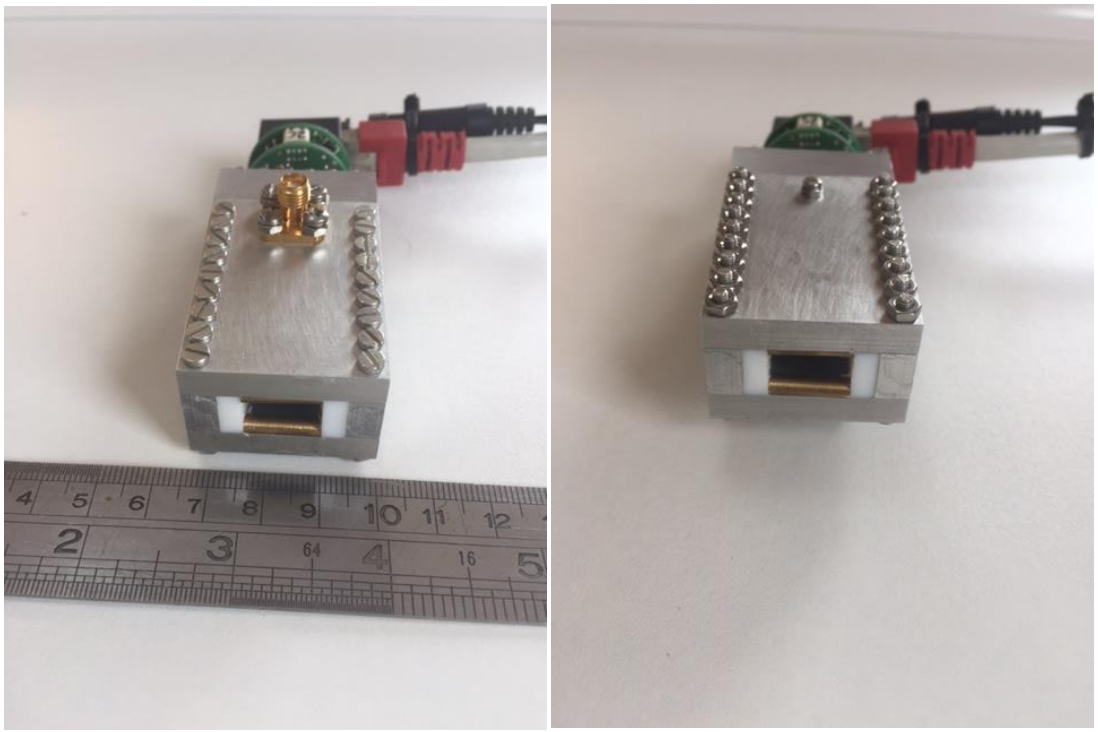


Figure 72 Front and underside (left) showing SMA connection and Front and top (right) showing tuning screw. Note: Camera connection has been rotated, reflector foil over PTFE not fitted.

Rear view showing Camera Tube

This picture was taken from approximately 6cm from the rear and the tuning screw and antenna can be seen through the camera tube.

Figure 73 Rear View of empty Camera Tube

Front and top view showing Aperture with foil reflectors

Figure 74 Front view showing aperture with foil reflectors fitted over PTFE side blocks

Simulated Characteristics of New Applicator

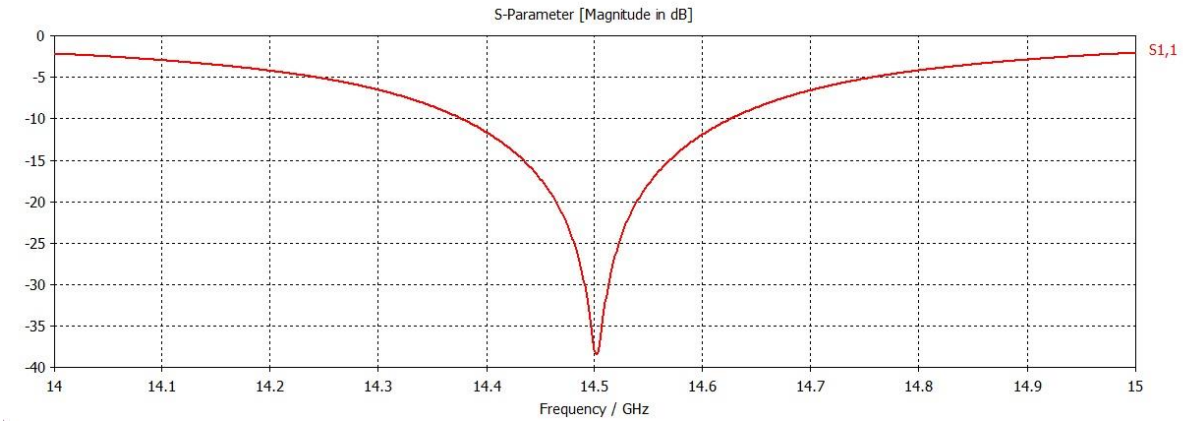


Figure 75 S11 reflection Coefficient

As shown in Figure 75, the new design provides a good match (50 Ω source) of approximately -35dB at 14.5GHz.

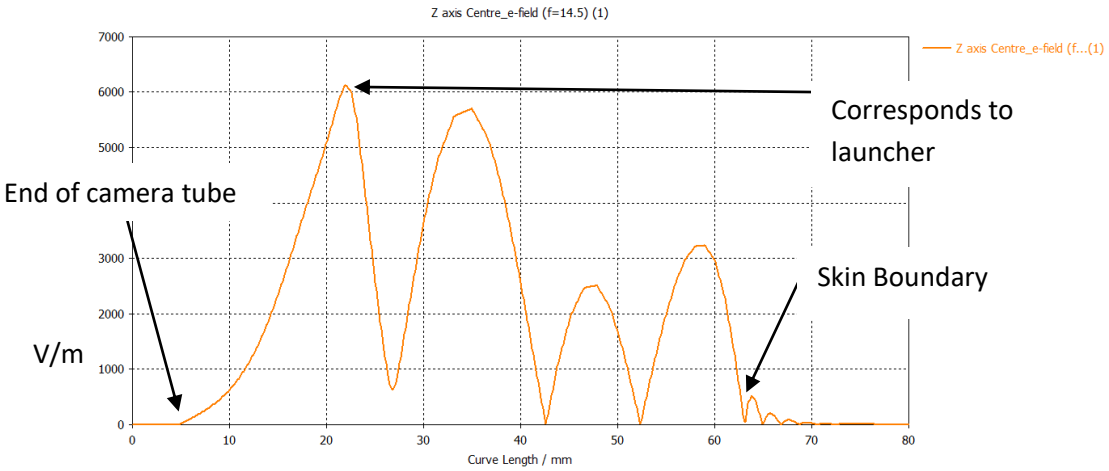


Figure 76 Magnitude of instantaneous E Field along Z axis (NMSA front to back axis)

Figure 76 shows the CST representation of the E Field for the NMSA with the outside end of the camera tube sealed off to simulate the effect of the camera installed. Note that this is the magnitude of the instantantaneous E field and shows the reduction in field strength down the camera tube, the field strength in the cavity may be seen as far greater and longer wavelength than in the skin.

Loss Profile Graphs of the NMSA

Guide to loss graphs

The following set of graphs shows the simulated loss profile across the aperture at depths into the skin of 0.1-2.0mm along horizontal lines at 0.9, 0.75 and 0.5 of the height of the top half; along the central horizontal axis; and along the central vertical axis. The arrangement of these lines is shown below:

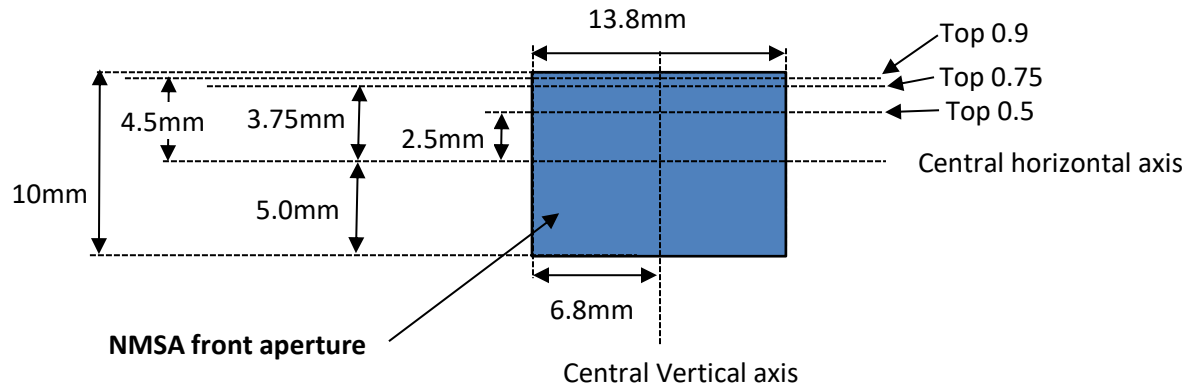


Figure 77 Location of loss curves in relation to aperture

Note that the length of these lines is wider (or taller in the case of the vertical loss line) than the actual aperture. Consequently it is simpler to interpret the horizontal axis in the following graphs from the perspective of the centre of the aperture – 23mm for the horizontal lines and 11mm for the vertical loss line. The vertical axis is shown in units of Watts/m^3.

Loss Graphs of NMSA at depths 0.1-2.0mm

Loss along Vertical axis

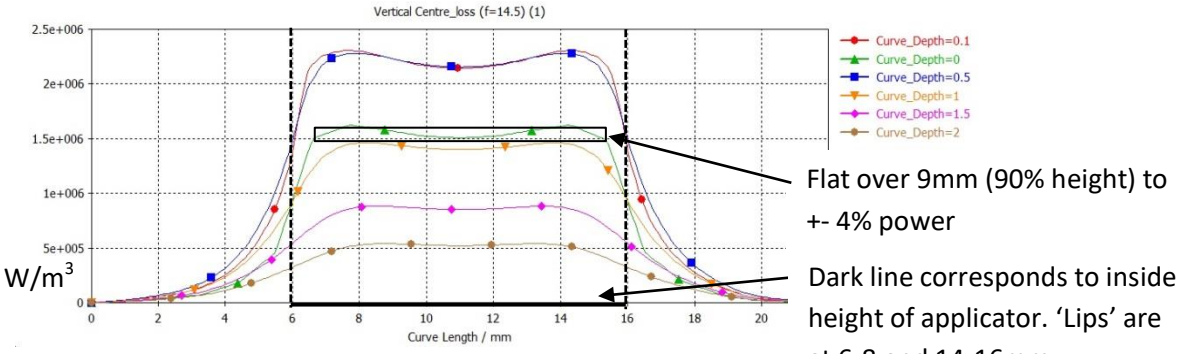


Figure 78 Loss profile along vertical centre line at various depths

The loss profile along the vertical centre axis is shown in Figure 78. Note that the losses appear to increase with greater depth and after discussion [4], it was assumed that this is due to the simulation mesh size near the skin boundary (0.4mm) being larger that the differences in depths and the possibility that the simulation extrapolates the fields within the mesh cells. The simulation was checked for resonances within the skin tissue (10mm thick) and none found so this would appear to be a reasonable explanation. The losses from 0.5mm to 2mm decrease with increasing depth and so appear more realistic. The losses along curve at depth 0mm may well be averaged between the air inside the applicator and the loss in the skin, hence the lower value of approximately 1.5 x 10^6 W/m^3 compared to 2.2x10^6 W/m3 at 0.5mm deep.

Given that the skin depth (as given in chapter 1) for IFAC 'SkinDry' [5] is 0.00216m hence attenuation constant $\alpha \simeq 463$ Nepers/metre $\simeq$ 4022dB/m (using 1Neper = 20$Log_{10}e \simeq 8.6859$ from [6]) , and using the formula for field strength given in Chapter 1, power at point d2 which is x metres deeper than point d1:

$$P_{d2} = P_{d1}\, e^{-2\alpha x}$$

Hence for 0.5mm difference, power would decrease by a factor of 0.63 which appears approximately correct for the centre at 0.5mm deep in relation to 1mm deep (2.2x10^6 vs 1.4x10^6 W/m^3 inverse ratio 0.64), 1mm deep in relation to 1.5mm deep (1.4x10^6 vs 0.85x10^6 inverse Ratio 0.61) and 1.5mm deep in relation to 2mm deep (0.85x10^6 vs 0.505x10^6 inverse ratio 0.59). Hence the loss at the surface centre may be in the region of 2.2x10^6/0.63 = 3.49x10^6 which is 2.32 times larger than the value on the graph so lending some weight to the possibility that the loss at 0mm depth is averaged between air and skin.

As may be see on the graph annotation, and assuming the profile of the loss at depth 0mm (i.e. the surface) is correct albeit reduced as described above, the power profile is flat within +-4% over 90% of the aperture height measured at the centre.

E field intensity $|E|^2$ may be related to the power loss curves P_{Loss} W/m^3 (valid for lossy media only):

$$|E|^2 = \frac{2 * P_{Loss}}{\sigma}\ \mathrm{V}^2/\mathrm{m}^2$$ Where for this design σ = 13.27 S/m

Loss along Horizontal axis

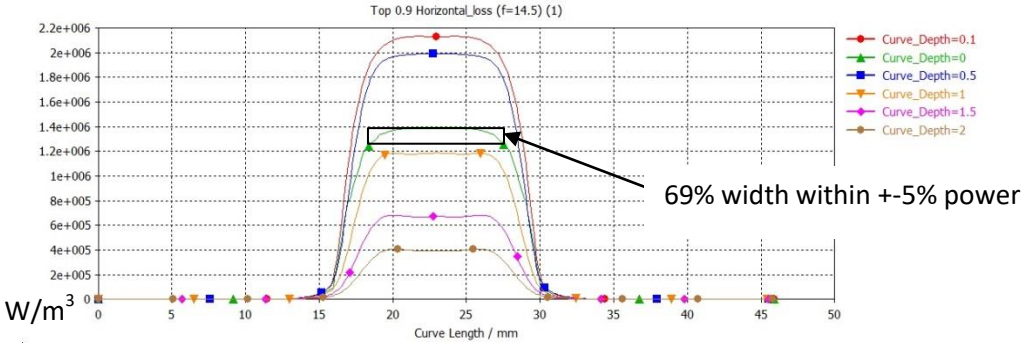


Figure 79 Top 0.9 Loss

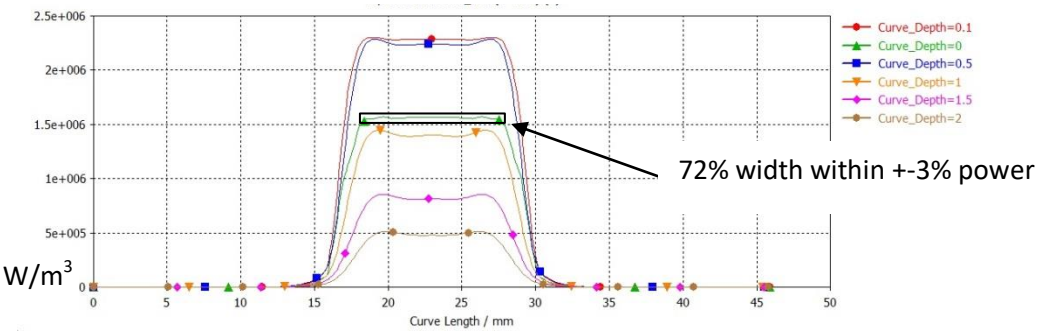


Figure 80 Top 0.75 Loss

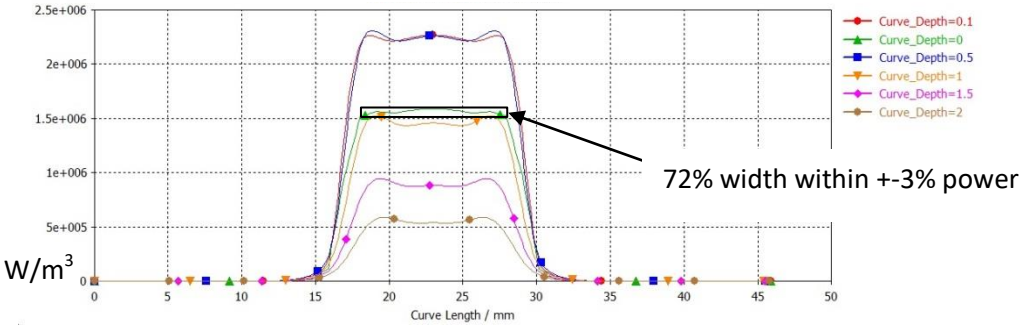


Figure 81 Top 0.5 Loss

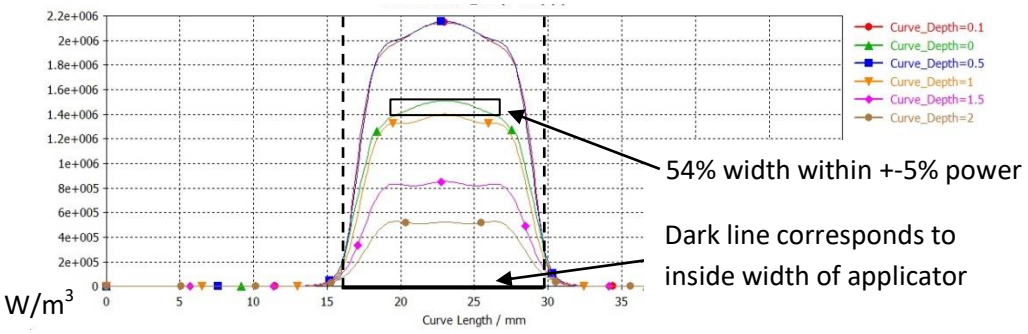


Figure 82 Centre Horizontal loss

Figure 79 to Figure 82 may be interpreted together with Figure 77 which provides the location of the loss curves. The horizontal centre demonstrates the least flat profile with 54% of the width having power within +-5%. This improves away from the centre the top 0.5 and top 0.75 demonstrating 72% of the width within +-3% and the top 0.9 with 69% width within +- 5%.

Loss Contours at depths 0.1mm – 2.00mm

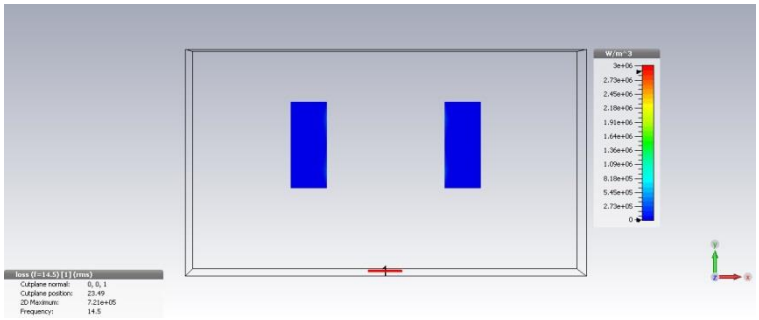


Figure 83 Loss in PTFE side blocks, included to show relative width of the aperture

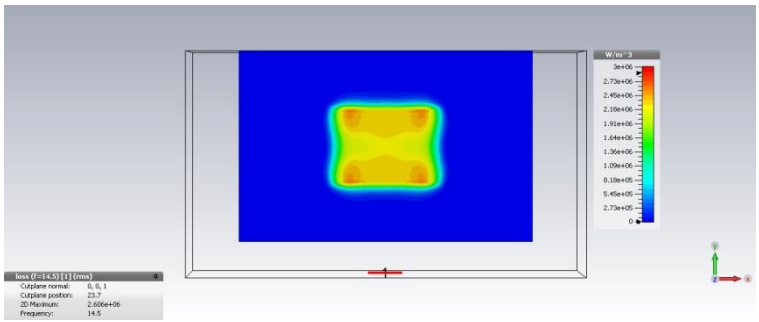


Figure 84 Loss at depth 0.1mm

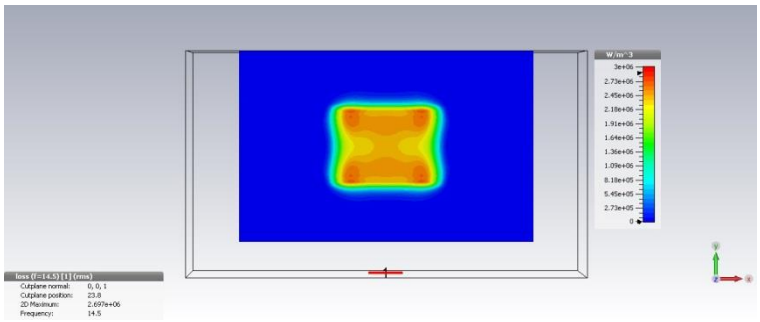


Figure 85 Loss at depth 0.2mm

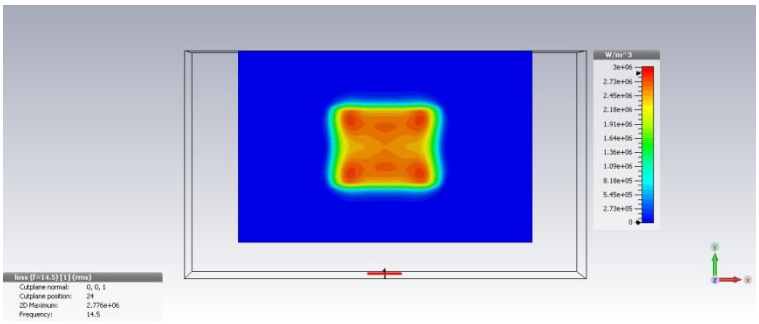


Figure 86 Loss at depth 0.4mm

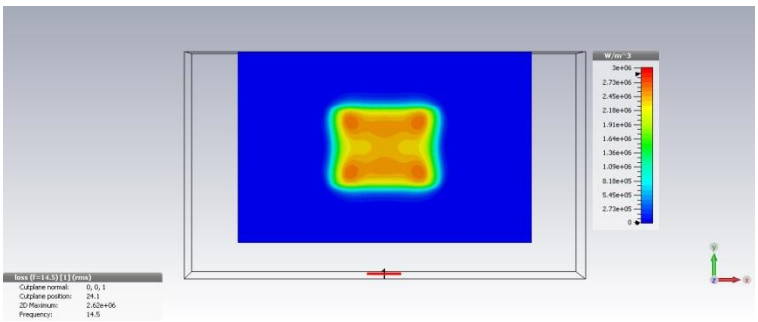


Figure 87 Loss at depth 0.5mm

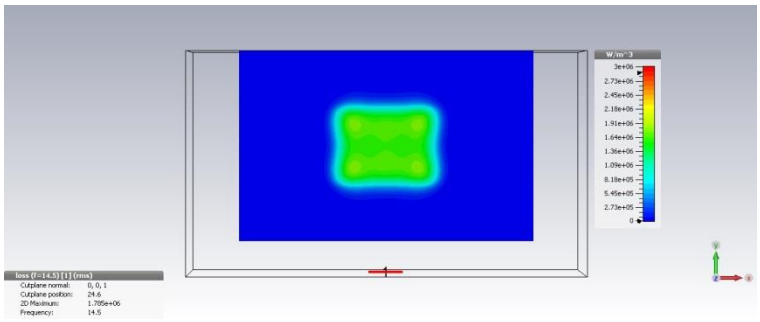


Figure 88 Loss at depth 1.0mm

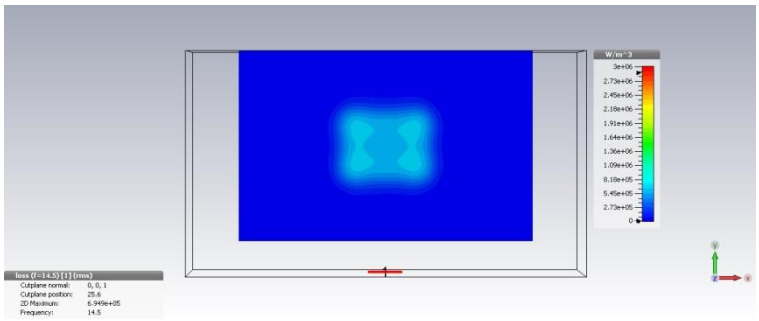


Figure 89 Loss at depth 2.0mm

Figure 84 to Figure 89 provide the loss contours with the inclusion of Figure 83 to relate the horizontal width. As previously discussed, the loss appears to increase until 0.4mm depth.

Block diagram of Test system

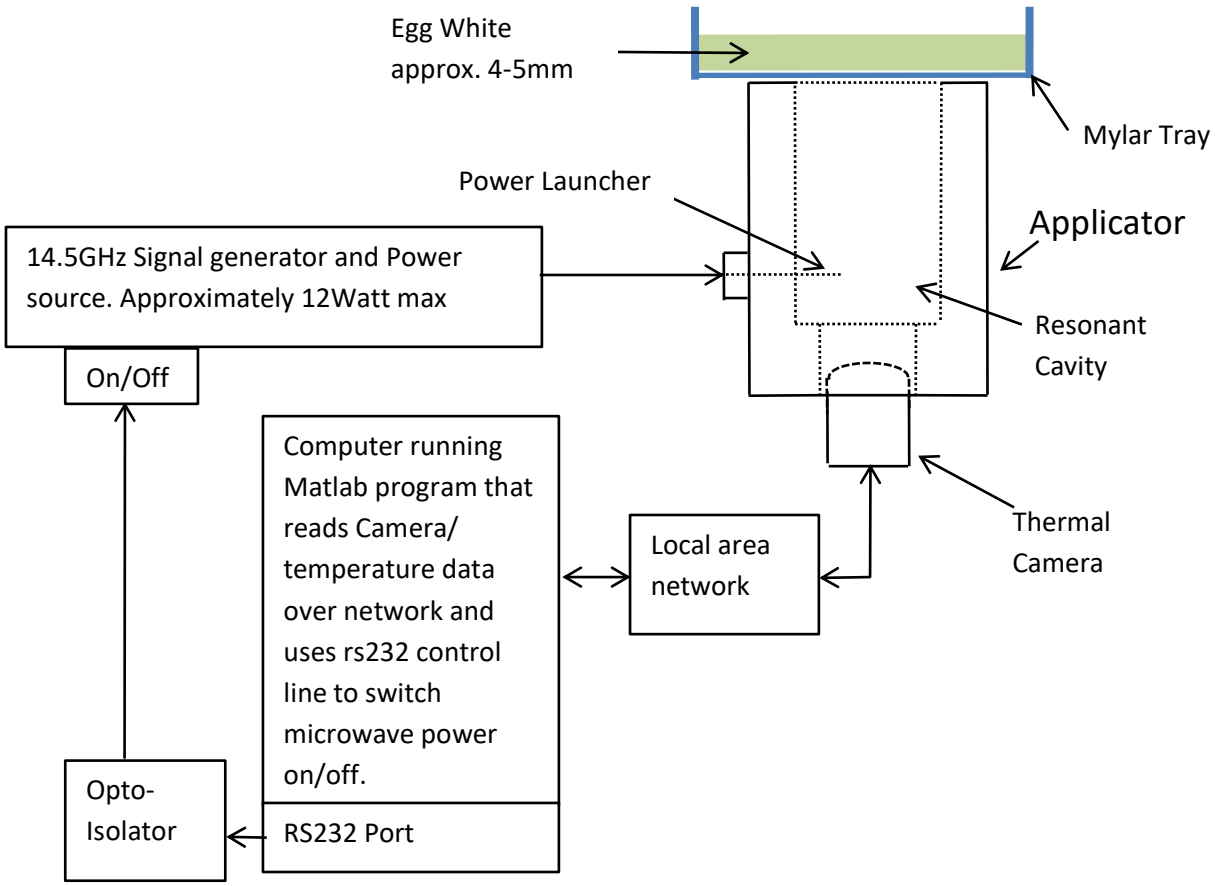


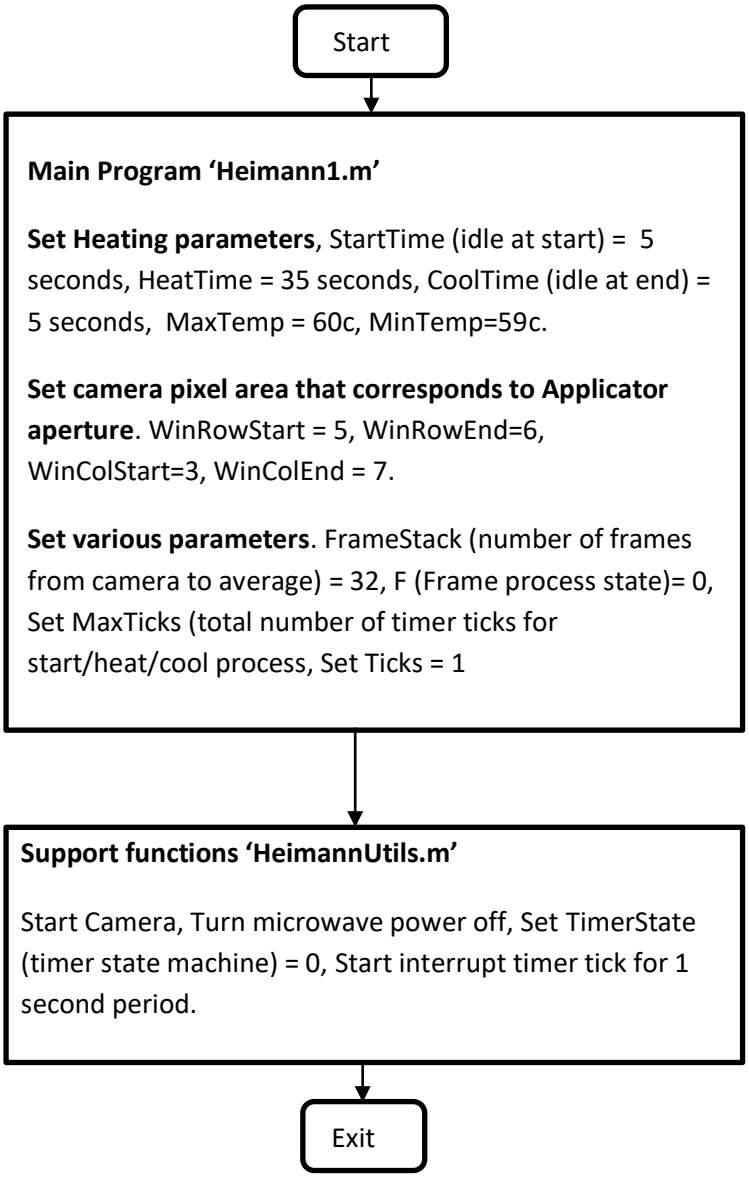
Figure 90 Block Diagram of test system

Explanatory notes:

1) The Heimann [1] HTPA8x8 thermal camera utilises a UDP network connection that , once initiated, provides a constant stream of frame based data.
2) The opto-isolator, driven by the RS232 port provides a simple means of On/Off control of the 14.5Ghz poweramplifier.
3) Power monitoring not shown.

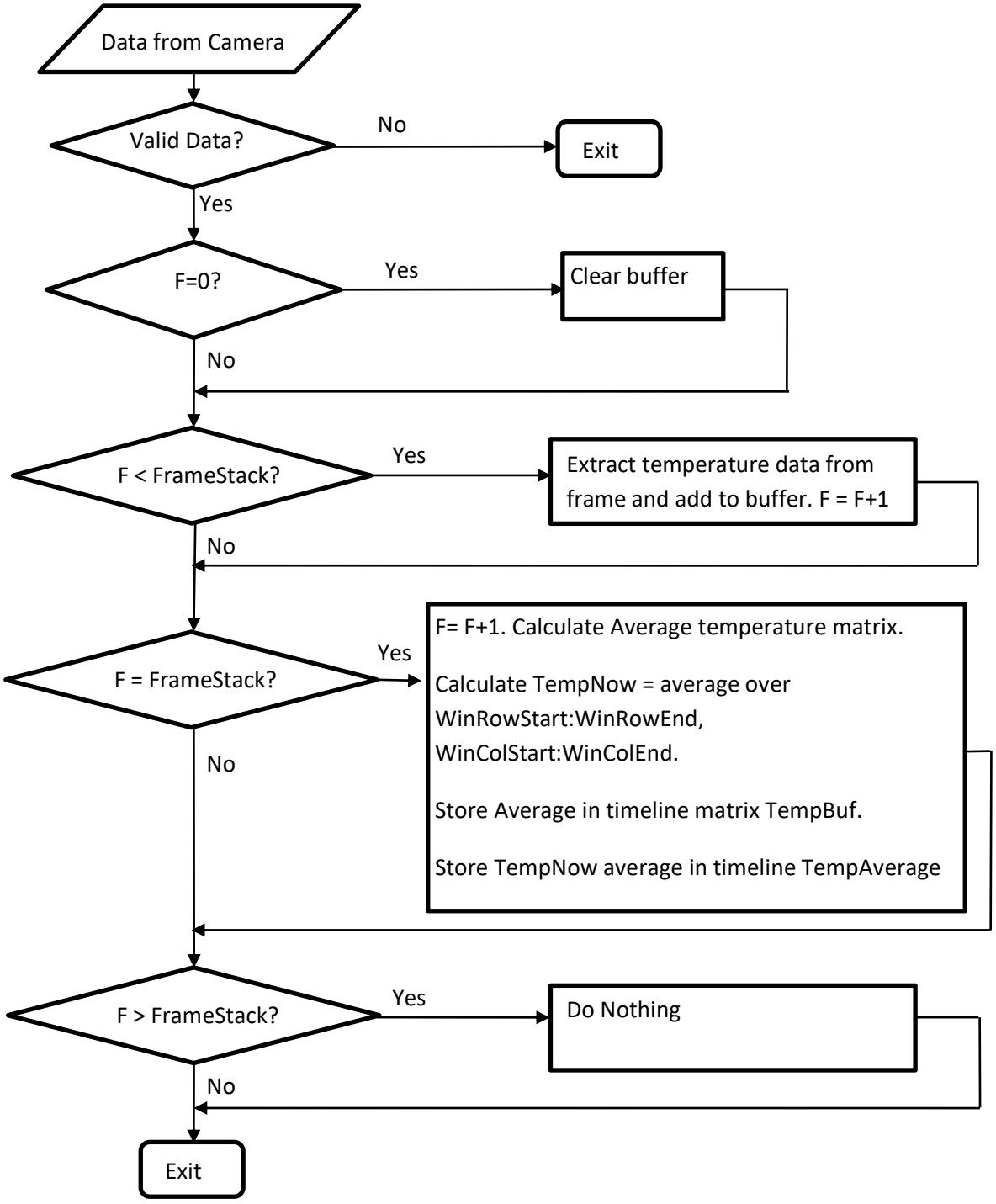
Description of Control Program

The Matlab program listings for 'Heimann1.m' and 'HeimanUtils.m' are shown in Appendix 3A – see footnote [10]



[10] Disclaimer: This code has been created for the control of power based on thermal feedback only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

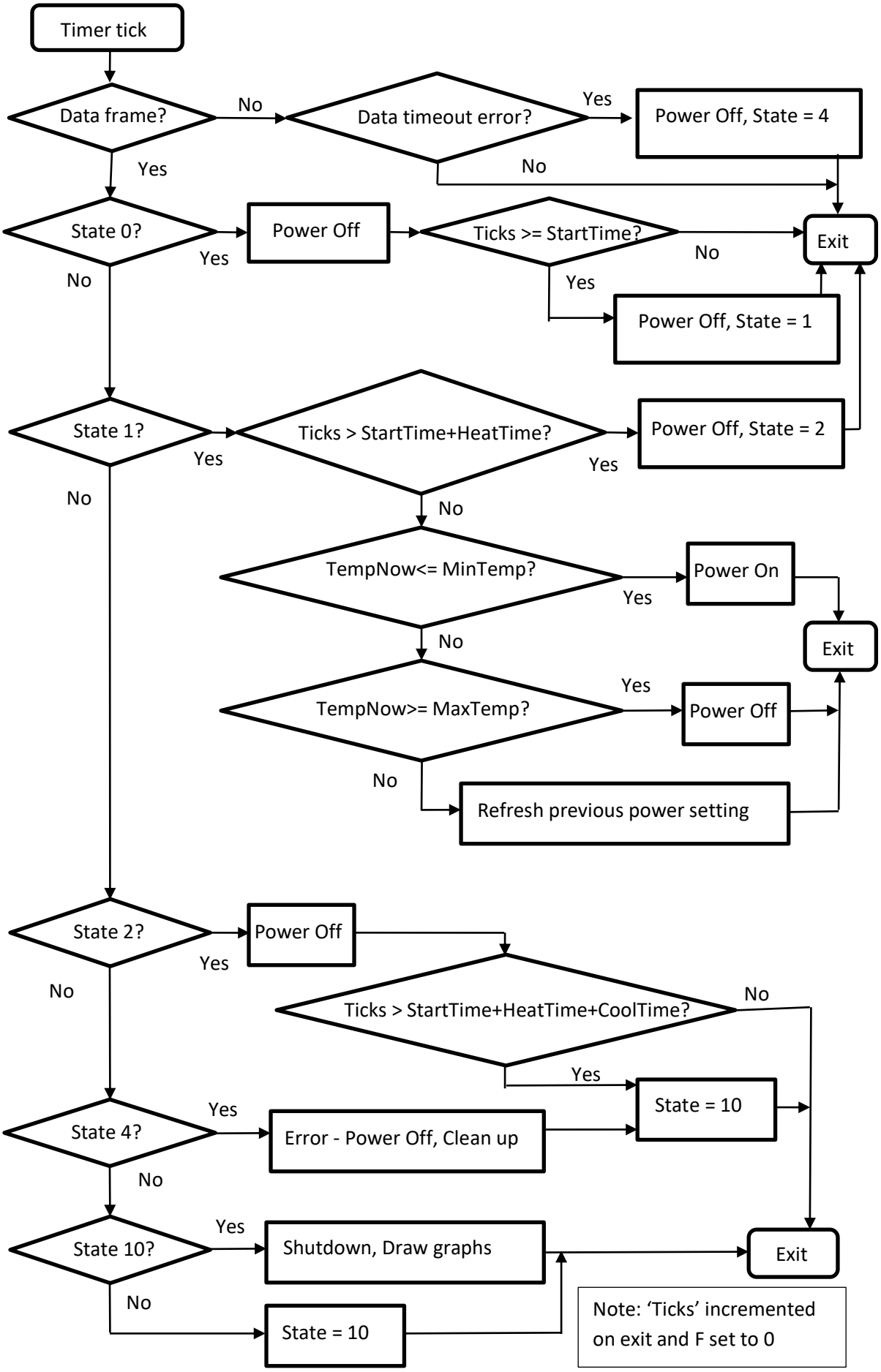
Data reception from camera flow chart



Explanatory notes:

The Heimann camera outputs a continual data stream. In order to avoid buffer overflows and lockups it is important to ensure all data is read from the camera even if the data is discarded. The timer tick function is responsible for removing the data on a periodic basis and making power on/off decisions based on instantaneous temperature (TempNow) and then freeing the buffer for further data storage.

Timer tick function flow chart



Preliminary Test of basic operation the System

The experimental system was agreed and supervised by Prof C Hancock who was satisfied with the low risk of harmful emission from an interference and safety perspective. A Nardalert XT [7] (used by engineers installing *Ku* band base stations which includes 14.5GHz [8]) and a spectrum analyser was positioned with 1 metre of the equipment and emission levels from the NMSA were deemed to be very low at the time. The experimental configuration is shown in Figure 90 with the Thermal camera partially inserted into the camera tube and involved approximately 35 seconds of power delivery at 14.5GHz at approximately 12Watts at source.

Future experiments will need to be under tightly controlled conditions possibly including a sealed EM enclosure for the Applicator and test material. The results presented here are for verification of viability of the design approach but are nevertheless useful for future comparison.

For use on human tissue, a full EMC study of the NMSA in accordance with regulatory requirements would need to be performed and would likely require the use of an anechoic chamber with measurements taken at 3m [8].

Plot showing matrix of Egg-White surface temperature over the Aperture from the Thermal Camera situated inside the NMSA

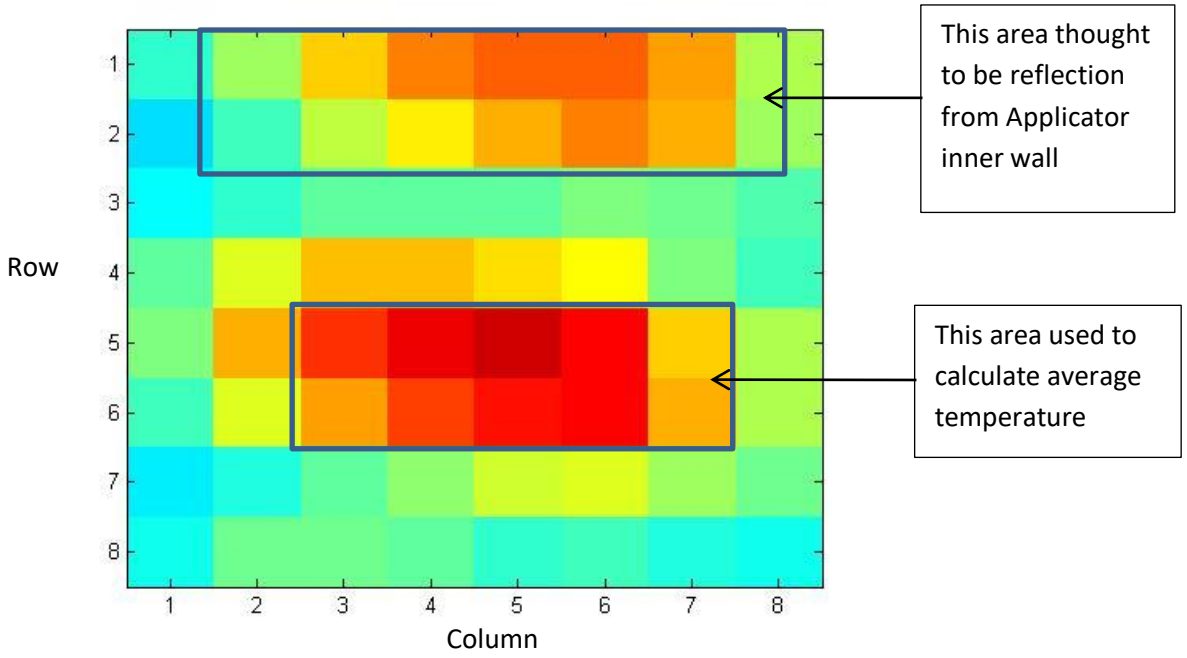


Figure 91 Temperature Matrix returned from Thermal Camera

Figure 91 above shows the Temperature Matrix obtained from the Heimann HTPA 8x8 [1] camera during the controlled heating test where the colour range is Red=Hottest, Blue = Coldest. The average temperature used to control the On/Off heating is calculated from Rows5-6 and Columns 3-7. A colour bar to indicate temperature is not available but an indication of the temperatures within the averaged temperature window may be obtained from Figure 92 (overleaf) with reference to Figure 93 at T=45 seconds.

Temperature vs Time for 2 Thermal Camera Sensor Rows

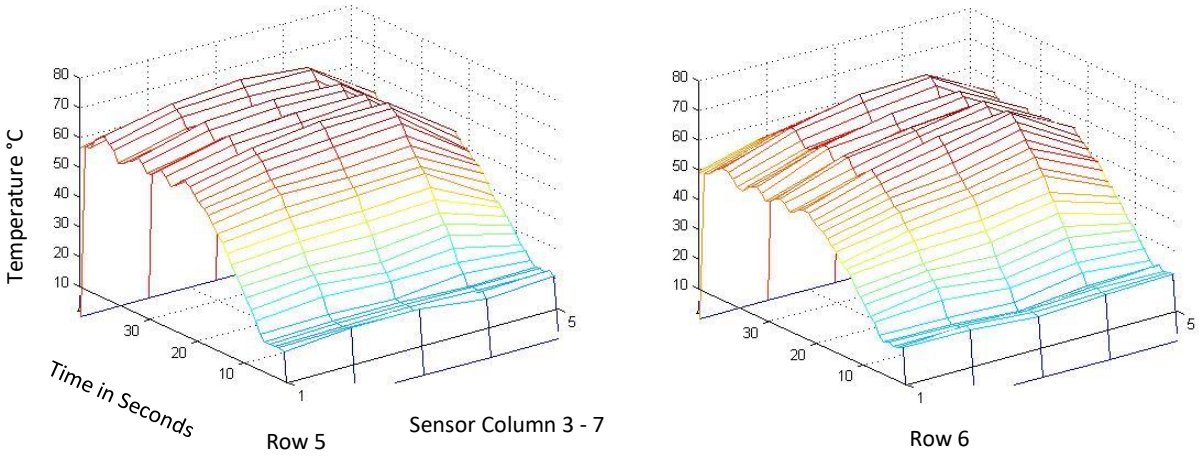


Figure 92 Heat profile of the 2 sensor rows used to calculate average temperature for power control

Figure 92 Shows the temperature profile for the 2 sensor rows. During the stable period from approximately 20-40 seconds there would appear to be approximately 10°C between the centre and outside indicating a flatter profile that would be obtained with a cosine shaped power profile. Note that the graphs' column annotations 1-5 correspond to the camera's sensor columns 3-7 respectively.

Average temperature of Egg-White surface over Applicator Aperture and power on/off profile recorded using Heimann thermal camera and Matlab program.

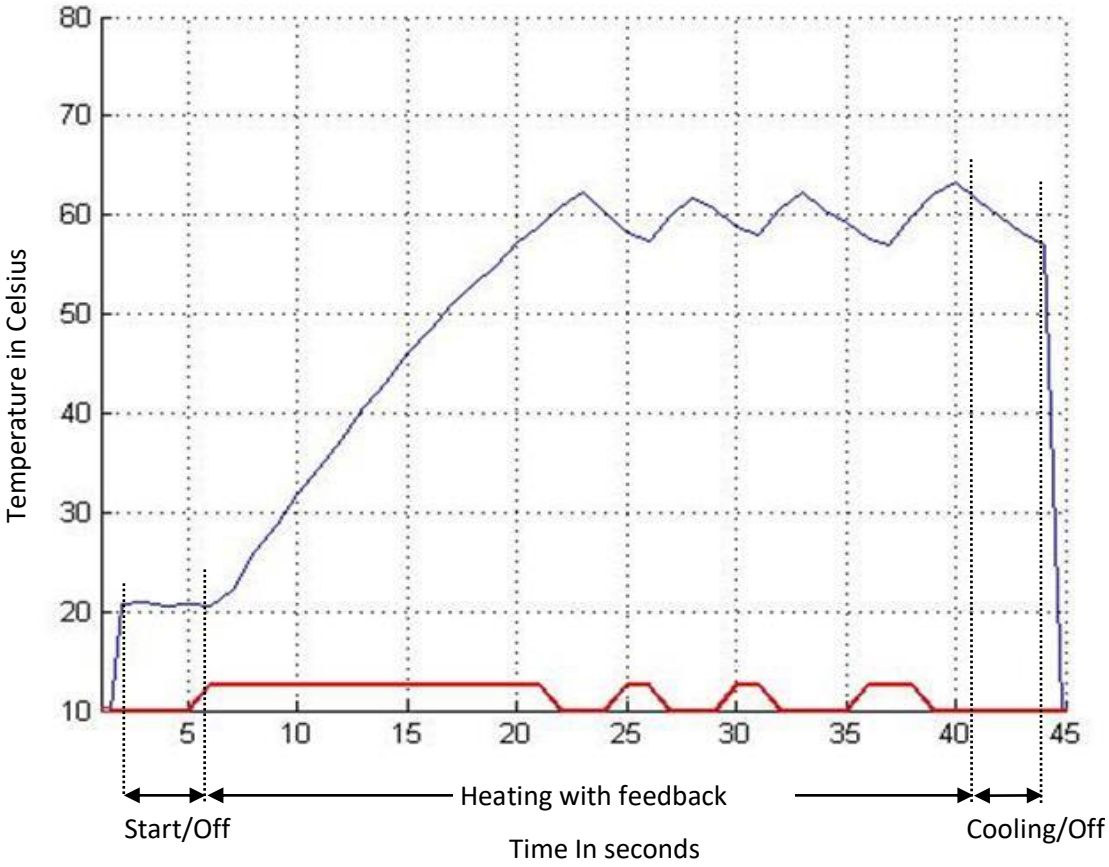


Figure 93 Average Temperate over Aperture and Power Profile

Figure 93 Demonstrates operation of the Temperature Regulation system. The Temperature is shown in Blue and the Power On/Off profile shown in Red. Power turns on at time = 6 seconds and off at time = 22 seconds. The displayed Power profile should have vertical slopes displaced by +1 seconds. The Lag from power on to temperature rise may be caused by averaging the temperature over 32 frames and sampling every second. The plot is valid at time = 1 second to time = 44 seconds. i.e. the steep slopes at beginning and end are not valid temperatures. Excursions of temperature during heating period may be due to lag caused by temperature averaging and sampling at 1 second intervals.

The slope of the temperature rise is approximately 40°C in 15 seconds which is far slower than the Thermal Simulation results in Chapter 2 which gives a rise of 38°C in 2 Seconds although the thermal simulation was based on the original applicator which has approximately 3x the surface area and 4x the power. This means that the temperature rise measured here is approximately 5.3x slower than it should be. This could be accounted for by differences in Permittivity and Conductivity between Egg white and Human Skin hence, less power delivered due to mismatch, losses in the cables, thermal properties of Egg white and errors in the thermal simulation. However, even with these differences, the target temperature of 60°C is achieved.

Future Work

To advance the development of the NMSA, the following list provides some of the requirements. As noted earlier, the results presented here are for verification of basic operation.

1) Rigorous experimentation and correlation with the original thermal simulations.
2) Additional means to verify object temperature such as thermocouples to determine temperature at various depths. Comparison of these results with the simulations previously performed.
3) Accurate correlation between thermal camera pixel location and Applicator aperture.
4) Repeat experimentation with a range of materials of known Permittivity and Conductivity.
5) Improvement of the program to account for lag due to sampling and averaging.
6) Provision of better mechanical fixing for camera to Applicator with some thermal insulation. The camera is affected by changes to ambient temperature and although the Applicator itself may have high thermal mass, it is likely to be affected by losses and heat conduction from the target object.
7) Improved focus and resolution of the Thermal Camera.
8) Addition of a synchronised visible light camera recording the coagulation of egg-white or discolouration of any other material would be helpful. This could include some application of thermochromic pigments in a standardised skin phantom.
9) Simultaneous independent recording of delivered power.
10) Further CST simulation of camera tube and possibly enclosure of the camera to further minimise emissions if required.
11) Recalculate CST loss curves to check and improve presentation of power loss/dissipation in skin tissue.
12) As suggested [9], evaluate the potential beneficial effects of tapered PTFE side blocks – this could result in a wider useful aperture area without the need to redesign the body of the NMSA; Further, this may provide a simple additional means of improving the loss profile.

Works Cited

[1] *Heimann Sensor GmbH, Maria-Reiche-Str 1, 01109 Dresden, Germany: HTPA 8x8 L7.0A Thermopile Array.*

[2] *The Mathsworks Inc, Matlab R2012A Student Edition,* Natick, MA, USA, 2012.

[3] *CST Microwave Studio,* CST GmbH, Bad Nauheimer Str. 19, 64289 Darmstadt, Germany.

[4] *Discussion of CST results with Malcolm White, 25 October 2016,* Creo Office, Bath.

[5] D. Andreuccetti, R. Fossi and C. Petrucci, "An Internet resource for the calculation of the dielectric properties of body tissues in the frequency range 10 Hz - 100 GHz, Based on data published by C.Gabriel et al. in 1996.," IFAC-CNR, Florence (Italy), 1997. [Online]. Available: Website at http://niremf.ifac.cnr.it/tissprop/. [Accessed 08 9 2016].

[6] "Neper," Wikipedia, 20 November 2016. [Online]. Available: https://en.wikipedia.org/wiki/Neper. [Accessed 20 November 2016].

[7] *Narda Safety Test Solutions GmbH, Sandwiesenstr. 7, 72793 Pfullingen, Germany.*

[8] C. Hancock, *Private Communication, November 2016.*

[9] C. Duff, *Viva, 22 February 2017,* Bangor.

Chapter 5 Conclusion and Future Work

Contents

Conclusion

The main aims of this work were set out in Chapter 1 which was to enhance the uniformity and control of the existing Microwave Skin Applicator for the ablation of skin lesions of 4-5mm ablation depth was set while minimising collateral damage to viable tissue.

Combined Thermal and EM modelling

The combined EM and thermal simulations of Chapter 2 of a volume of tissue treated by the existing Applicator indicated that direct thermal feedback from the target tissue provided compensation for some variation in power delivery and some variation in dermal thermal parameters. Three frequencies were investigated in Chapter 2 namely 5.8GHz, 14.5GHz and 24.125GHz with 5.8GHz appearing to provide the greatest ablation depth with the higher frequencies providing better damage margins and with 14.5GHz providing slightly better ablation depth than 24.125GHz.

The effect of surface cooling was investigated by simulating various wind speeds across the ablated surface, the results indicating that the improvement in ablation depth would be 1.25mm, 0.5mm and 0.25mm for 5.8GHz, 14.5GHz and 24.125GHz respectively for a wind speed of 100m/s at 20°C with damage margins increasing a similar amount for 5.8GHz and 24.125GHz but with a lower increase of damage margin for 14.5GHz.

The effect of Temperature bias was investigated which indicated that all three frequencies demonstrated greater ablation depth with reduced damage margin with indications that ablation depths of 7.5, 4.25 and 3.0mm at 5.8GHz, 14.5GHz and 24.125GHz respectively may be achievable if the skin surface is forced to 0°C. This implies that energy at 14.5GHz could meet the original aim of 4-5mm ablation although the simple mechanism of thermal feedback would not be viable.

A further outcome of these simulations indicate that the presence of subcutaneous fat of 0-3mm thickness at depths of 1-3mm had a significant effect on the heating profile and where these thicknesses were not known, operation at 5.8GHz may produce large damage margin with larger variation of the ablation depth. For example, supposing a definite 1.0mm ablation depth was required, a 5 second heating procedure at 5.8GHz might produce 1-2.75mm ablation with 3.25-4.5mm damage margin compared with 3 seconds at 14.5GHz producing with 1-1.25mm with damage margin 2-2.25mm, and 4 seconds at 24.125GHz producing 1mm ablation and 1.75-2mm damage margin.

In sum there appears to be a clear tradeoff where the higher frequencies demonstrate better damage control while the lowest frequency provides the best ablation depth although temperature bias and surface cooling may improve the ablation depths sufficiently to achieve the original aims albeit with the complication of interference with surface temperature regulation.

Improvements to heating uniformity

This was investigated in chapter 3 by simulation of the existing Applicator at 14.5GHz with the outcome that PTFE side blocks in the resonant chamber provided the simplest means of flattening the power dissipation profile across the treatment area. Other methods were investigated but the results obtained did not have the desired effect, possibly due to the actual implementation of the simulation and the parameters chosen.

Design of the New Applicator

Due to the wealth of existing information from previous studies at 14.5GHz [1] [2], a new Applicator was designed and built for operation at 14.5GHz incorporates a thermal camera and PTFE side blocks. A Matlab [3] program was written which controls the ablation process based on the required surface temperature by using the output of the thermal camera. Preliminary testing indicated that the concept of using a thermal camera to control the surface temperature may be viable and also provides some confirmation of the flatter power dissipation profile. This new design is partially customisable in that the internal PTFE and internal aluminium side blocks are replaceable without the need for a new body and that some adjustment to resonance is provided by means of a tuning screw.

Future work

Simulation programs

The simulations coded and experimented with have provided a great amount of insight into the propagation and dissipation of electromagnetic energy together with the corresponding thermal effects. These simulations could be greatly enhanced by use of subgridding to improve resolution and the simultaneous use of multiple processors to decrease execution time which can be very long for even the coarsest grids (several hours for multiple runs for comparison purposes). These improvements would be advantageous in order to use the Matlab 3D EM model as input to the thermal modelling, especially to reduce the simulation time in cases such as the comparisons for various fat thicknesses; and for modelling comparatively thin structures such as the epidermis.

Although some validation of these numerical models has been achieved, it is necessary to use another independent means of verification. In the case of the 3D EM model, this issue has been

resolved by the use of CST Microwave Studio [4] and the thermal model was checked informally by simulation of heated water but the model still requires rigorous validation which could be at least partially achieved by further experimentation with the New Applicator. Also, the assumptions made such as constant Thermal Conductivity during ablation require re-evaluation and refinement.

Further Experimentation with the New Applicator and enhancements

As outlined previously, the New Applicator has only been given preliminary testing to establish basic operation. Further experimentation is needed to:

- Ensure EM radiation does not escape in accordance with regulatory requirements under any conditions of use. This will require further evaluation of the EM Emissions from the aperture/skin boundary and evaluation of the effects of small 'air gaps' around the periphery and the EM Emissions from the camera tube. The addition of an EM leakage detector to the Applicator itself may be feasible.
- Confirmation of temperatures reached in the target tissue and comparison with the thermal modelling.
- Further use of the Thermal Camera and small doses of heating might provide useful information regarding the skin structure and characteristics. It may be possible to extrapolate some aspects of the skin Permittivity and Conductivity from the Applicator's S_{11}. The use of a known skin phantom should at least provide the ability to calibrate the New Applicator and so provide useful feedback as to power absorption.
- Some skin lesions appear as protrusions above the skin surface, the effect of these perturbations on the resonance of the antenna cavity will require investigation. The application of a gel or similar substance that could be placed around the lesion in order to obtain a flat surface. Alternatively a smaller Applicator aperture or 'Mask' may be helpful to treat small lesion areas or for use on uneven surfaces.
- The mixed use of 5.8GHz and 24.125GHz might be useful to attain the wider combination of ablation depths. This may be challenging to achieve in a single device but more viable from a regulatory standpoint due to the ISM frequency bands.
- The addition of an optical camera to position the Applicator over a target lesion before treatment would be helpful.
- Investigation of any beneficial effects of cooling the skin surface immediately after ablation in order to minimise the damaged margin.
- The possible use of tunable meta-materials [5] to replace the PTFE. This may be especially valuable for dual or multiple frequency operation.
- The addition of a form of lockout that disables the power amplifier if the Applicator is not in full contact with the skin. This could take the form of a combination of checking S_{11} , an internal visible light sensor and external illumination and pressure switches along the Applicator periphery.

In conclusion, some of the original aims have been partially met - it would appear that a temperature regulated system should provide better ablation control and a PTFE loaded applicator evens out the heating profile, surface temperature bias or cooling deepens the ablation. These now need to be rigorously verified against the simulated results with the New Applicator.

Works Cited

[1] C. Hancock, “A New Wave in Electrosurgery - Therapeutic Applications of Microwave/RF Energy and Novel Antenna Structures,” in *1EEE MTT-S International Microwave Workshop Series on RF and Wireless Technologies for Biomedical and Healthcare Applications (IMWS-Bio 2013)*, Singapore, 2013.

[2] C. Hancock, “Skin Cancer Treatment System, Medical Microwaves Systems Research Group,” Bangor University, Bangor, 2012.

[3] *The Mathsworks Inc, Matlab R2012A Student Edition,* Natick, MA, USA, 2012.

[4] *CST Microwave Studio,* CST GmbH, Bad Nauheimer Str. 19, 64289 Darmstadt, Germany.

[5] J. Zhong, Y. Huang, G. Wen, H. Sun and W. Zhu, “The Design and Applications of Tunable Metamaterials,” *Procedia Engineering,* vol. 29, pp. 802-807, 2012.

Appendices

Contents

Appendix 1A Effect of Surface Cooling

Effect of Surface Cooling at 5.8GHz

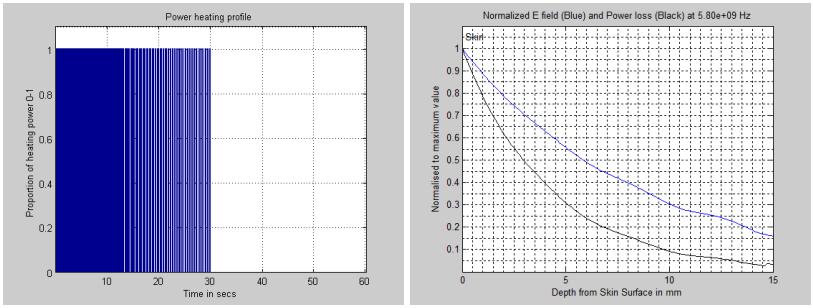


Figure 94 Power on/off Profile and Power Dissipation Profile at 1 m/s Wind Speed

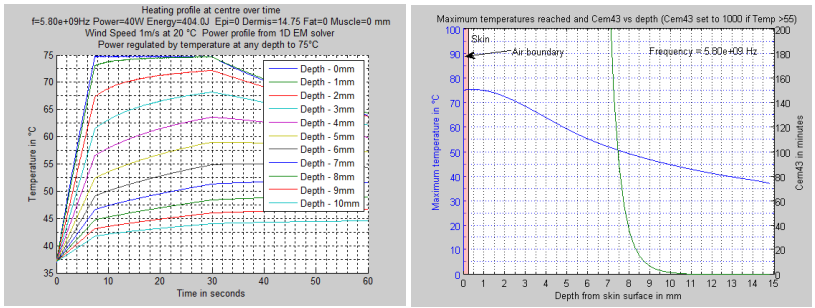


Figure 95 Temperature vs time and ablation/safe depths at 1 m/s Wind Speed

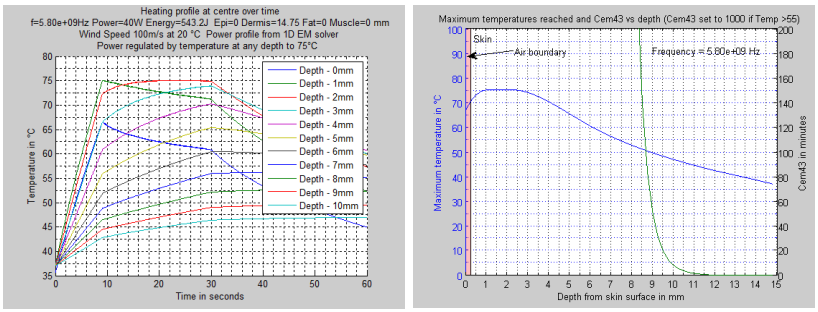


Figure 96 Temperature vs time and ablation/safe depths at 100 m/s Wind Speed

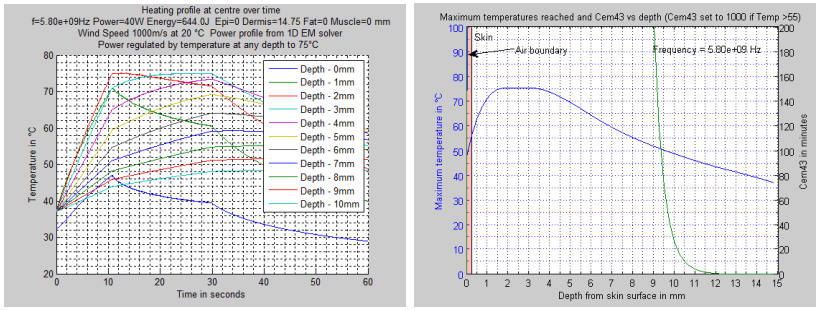


Figure 98 Temperature vs time and ablation/safe depths at 1000 m/s Wind Speed

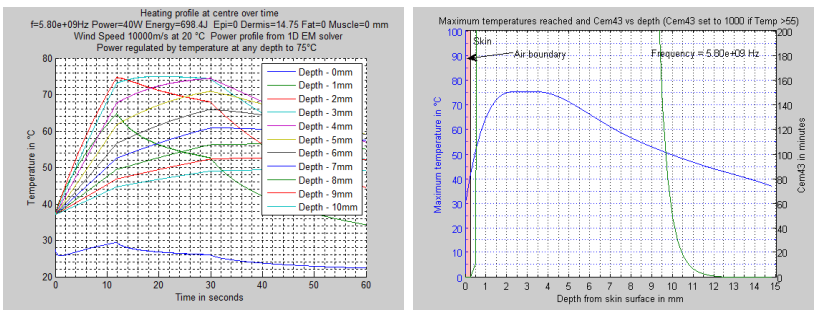


Figure 97 Temperature vs time and ablation/safe depths at 10000 m/s Wind Speed

Effect of Surface Cooling at 14.5GHz

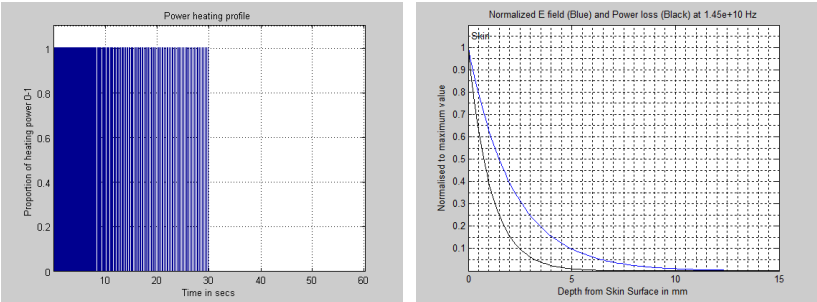


Figure 100 Power on/off Profile and Power Dissipation Profile at 1 m/s Wind Speed

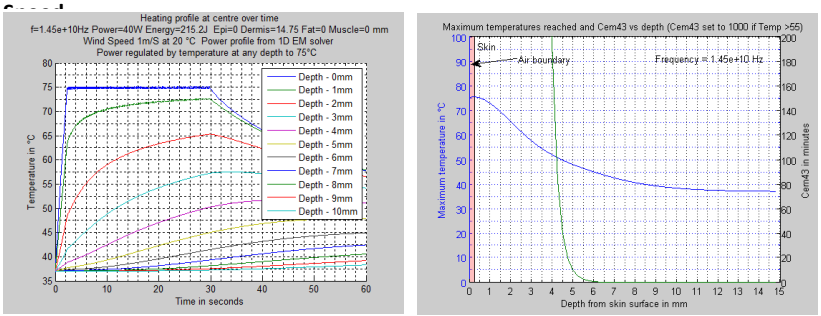


Figure 99 Temperature vs time and ablation/safe depths at 1 m/s Wind Speed

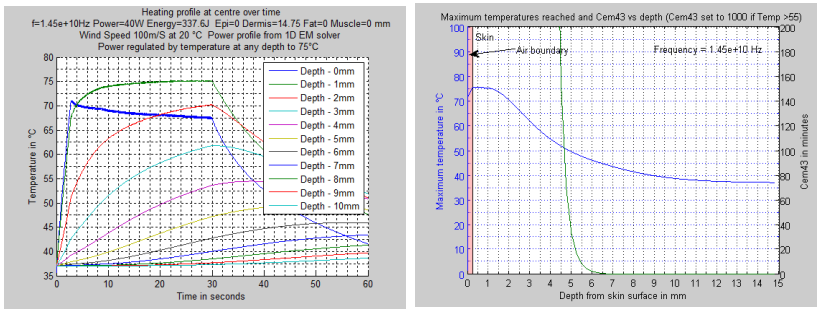


Figure 101 Temperature vs time and ablation/safe depths at 100 m/s Wind Speed

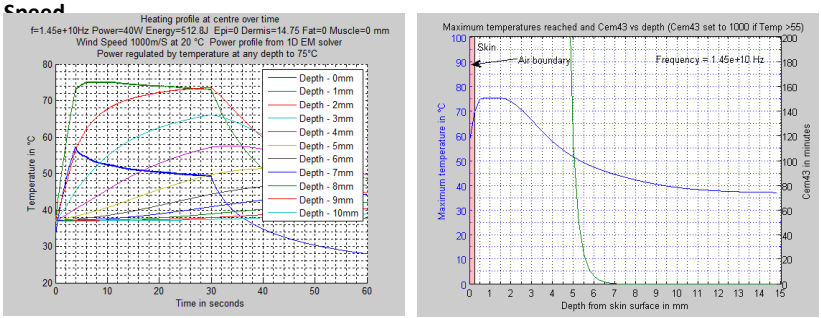


Figure 102 Temperature vs time and ablation/safe depths at 1000 m/s Wind Speed

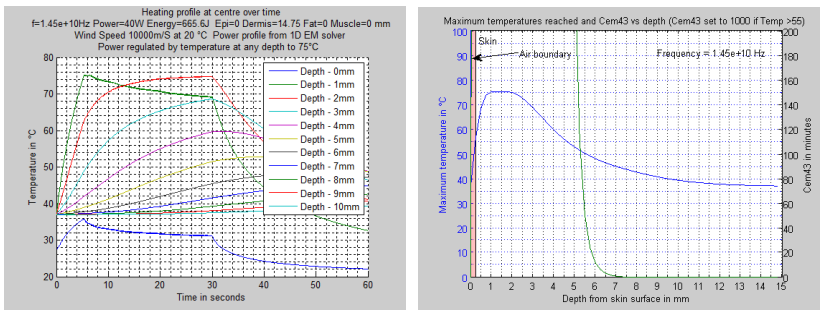


Figure 103 Temperature vs time and ablation/safe depths at 10000 m/s Wind Speed

Effect of Surface Cooling at 24.125GHz

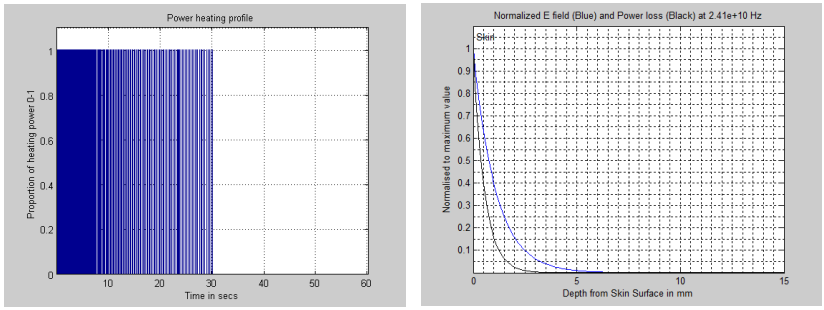


Figure 104 Power on/off Profile and Power Dissipation Profile at 1 m/s Wind Speed

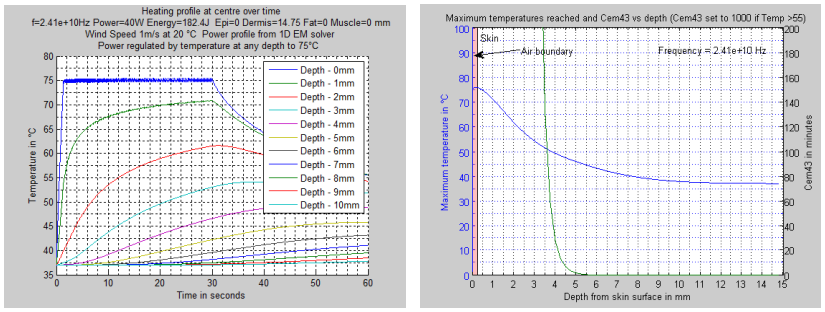


Figure 105 Temperature vs time and ablation/safe depths at 1 m/s Wind Speed

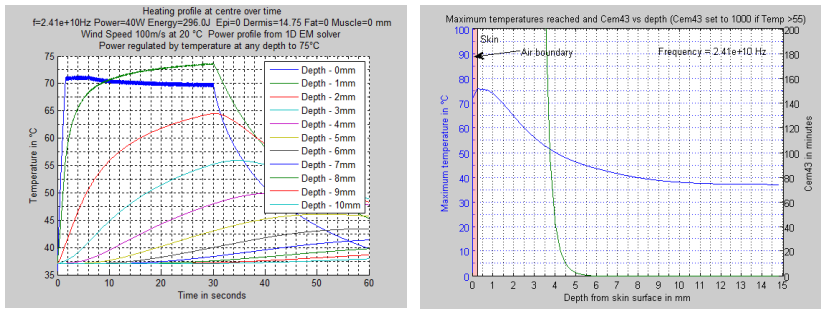


Figure 106 Temperature vs time and ablation/safe depths at 100 m/s Wind Speed

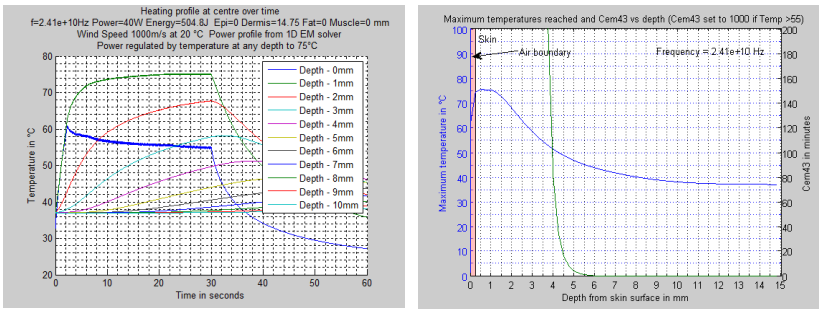


Figure 107 Temperature vs time and ablation/safe depths at 1000 m/s Wind Speed

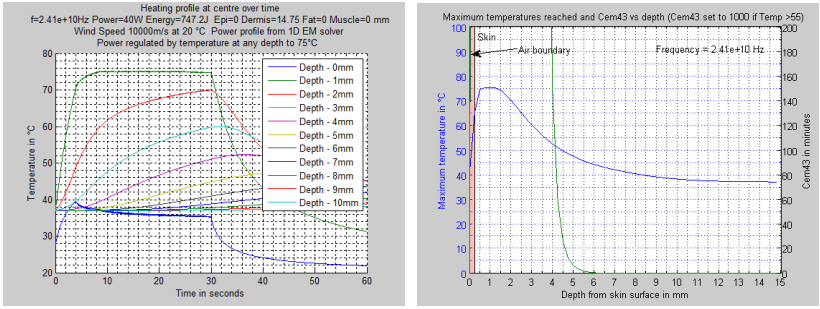
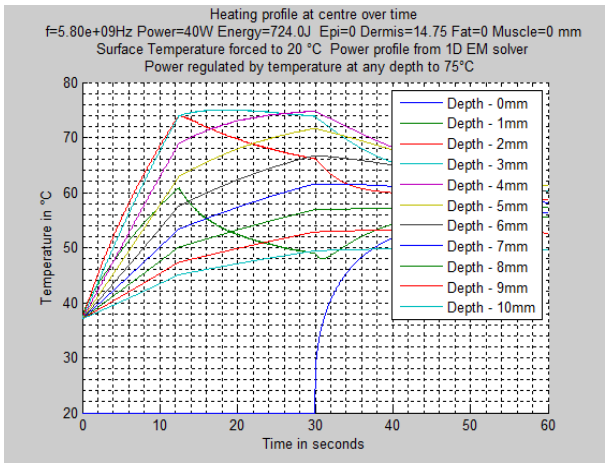
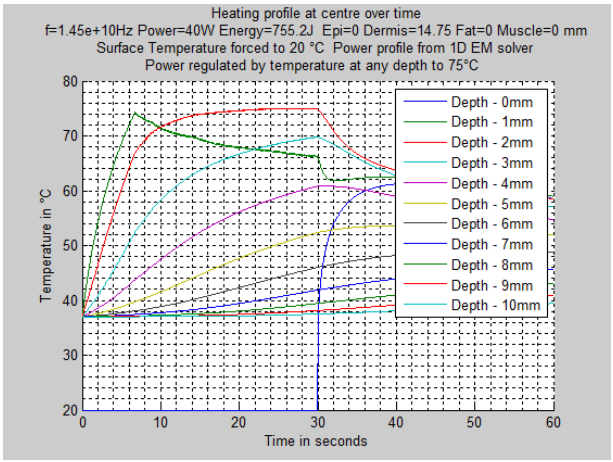


Figure 108 Temperature vs time and ablation/safe depths at 10000 m/s Wind Speed

Appendix 1B Heating Profile with forced aperture temperature at 20°C





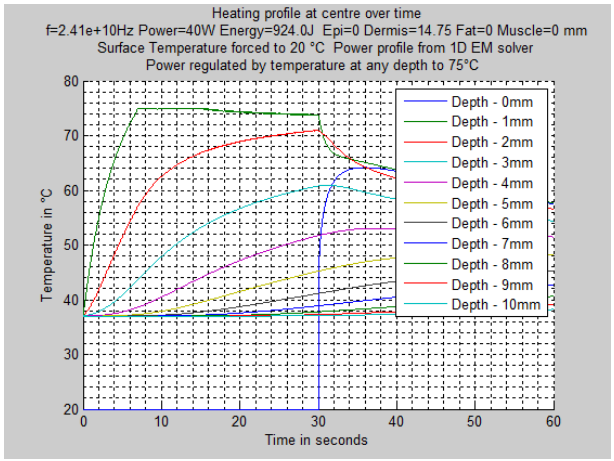


Figure 109 Heating profile for forced cooling for 5.8GHz (top), 14.5GHz (Middle) and 24.125GHz (bottom)

Appendix 1C Effect of Subcutaneous fat

1mm of Fat at 3mm Depth with Ablation 0-3mm

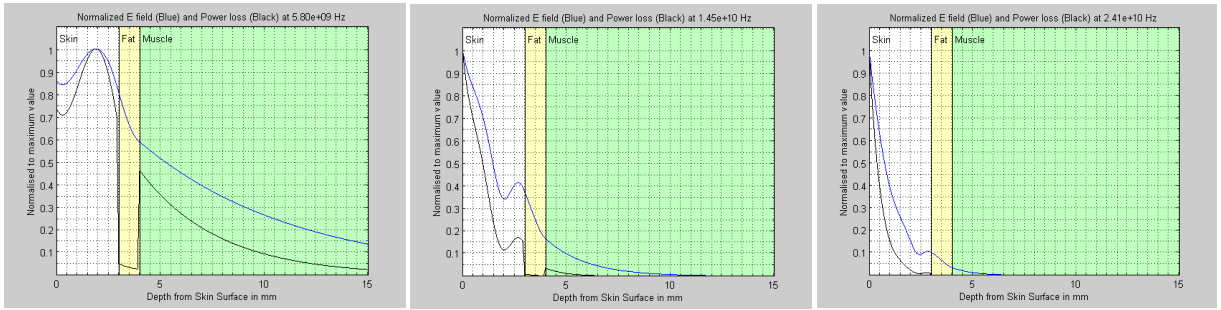


Figure 110 EM Profile for 1mm of fat at 3mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

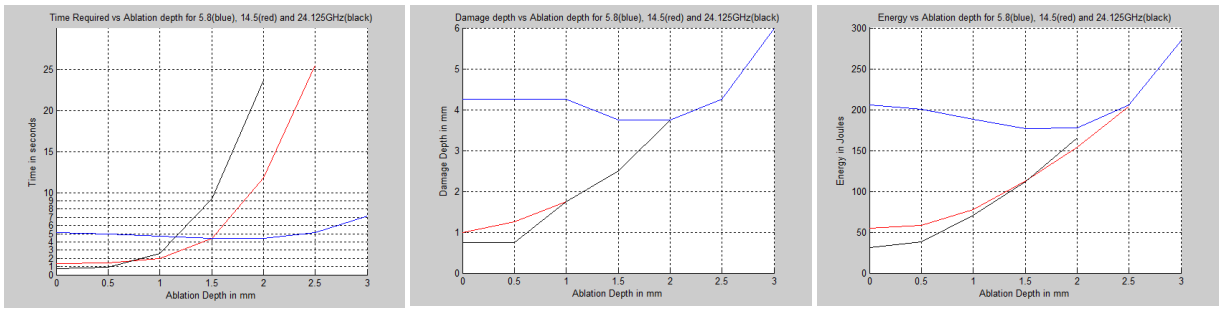


Figure 111 Time vs Ablation (Left), Damage vs Ablation (Middle) and Energy vs Ablation (Right) for 1mm Fat at 3mm depth

1mm of Fat at 2.0mm Depth with Ablation 0-3mm

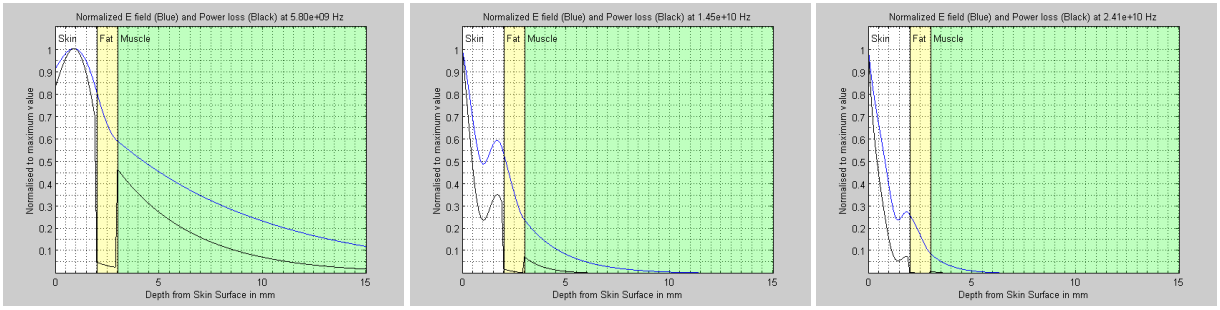


Figure 112 EM Profile for 1mm of Fat at 2mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

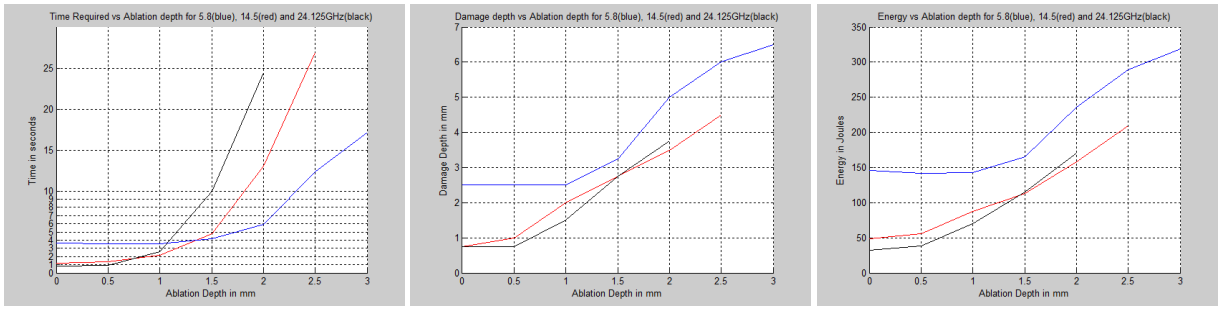


Figure 113 Time vs Ablation (Left), Damage vs Ablation (Middle) and Energy vs Ablation (Right) for 1mm Fat at 2mm depth

1mm of Fat at 1.0 mm Depth with Ablation 0-3mm

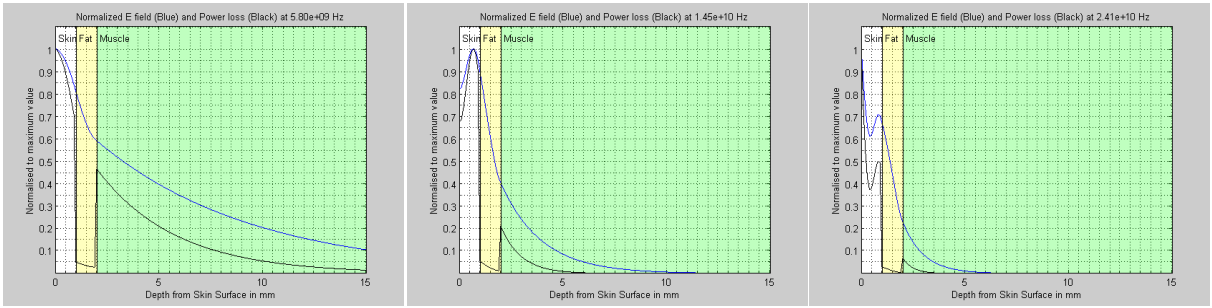


Figure 114 EM Profile for 1mm of Fat at 1mm depth for 5.8GHz (Left) 14.5GHz (Middle) and 24.125GHz (right)

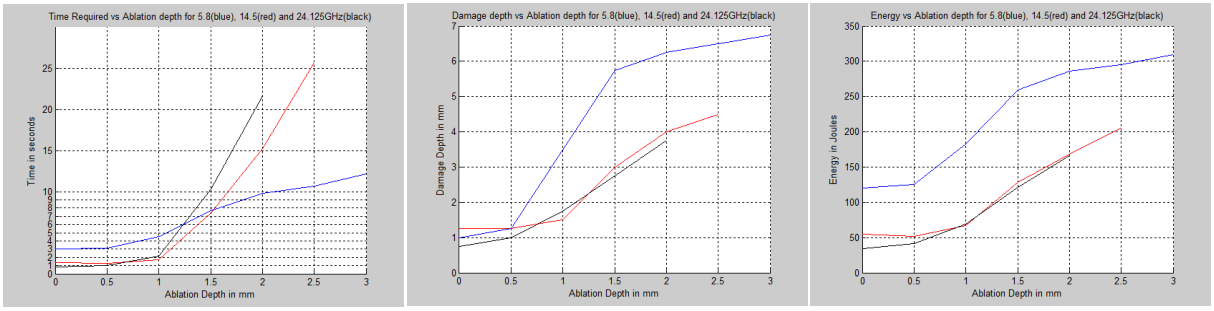


Figure 115 Time vs Ablation (Left), Damage vs Ablation (Middle) and Energy vs Ablation (Right) for 1mm Fat at 1mm depth

Appendix 1D comparison of Reference heating profile with different Convective parameters

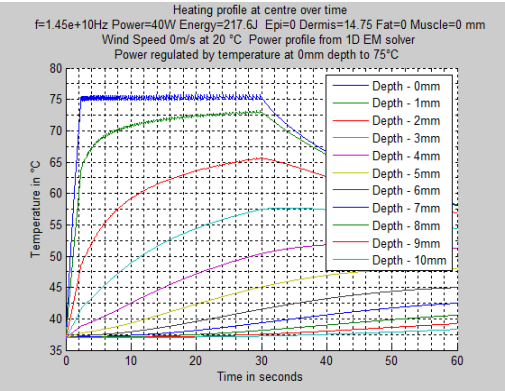


Figure 116 Reference simulation with Convective value of 10 W/M^2 °C

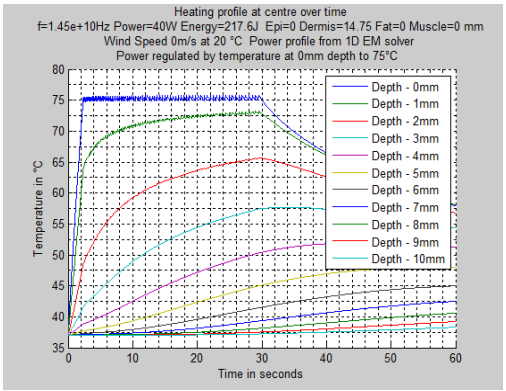


Figure 117 Reference simulation with Convective value Ha of 10.5 W/M^2 °C

There appears to be no obvious differences between the two simulations above and the total power delivered is the same at 217.6J

Appendix 1E Heat5run.m - Calling Script to run EM and Thermal Modeller

Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has
not been verified as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
% This Script uses HEAT5 (Thermal) and Planewaveldv3 (EM) to perform various thermal
% simulations. In general the EM modeller is invoked initially to generate
% a Power profile which is then passed to HEAT5 to perform the thermal
% modelling
%
%Paul Horwitz OCT 2016
%--------------------------------------------------------------------------------
clear all
close all

Epi_Thickness = 0.000;         %Epidermis - same perm/cond values used as dermis
Dermis_Thickness = 0.001;      %Dermis thickness max 0.01475 for 0.25mm step - same perm/cond values used as epidermis i.e.
'Skin'
Fat_Thickness   = 0.00;         %m Thickness of fat

Muscle_Thickness = 0.01475 - (Dermis_Thickness+Fat_Thickness); %Thickness of muscle but not used - goes to end of boundary

Tissue_Plot_Thickness = 0.015;  %Graph this depth
Smooth_Fat_Len = 0.00;          %Linear transition for skin--fat

Type_F = 0;                     %Frequency and tissue parameter chooser for testing 0=14.5, 1 = 5.8, 2 = 24.125
%
Hi_Res_Half = true;             %True is low res
if Hi_Res_Half ==true           %maybe use twice resolution, true = low res
    Delta_T =  0.02;            %Time step 0.05?
    Delta_Len = 0.00025;        %cell size in m .00025
else
    Delta_T =  0.008;           %Time step
    Delta_Len = 0.0001 ;%25;    %cell size
end

Run_Time = 60;                  %Thermal run time total
Input_Power = 40;               %Incident power over aperture
Use_Power_Profile = true;       %if true use input power profile from EM, else use calculated e-ax for skindry for whole area

Plots_At_Runtime = false;       %no real time plots

Force_Aperture_Temp = false;    %Set true to force aperture temp
Force_T = 20;                   %Temp we are forcing aperture to
%
Ambient_T = 20;                 %Temp at Face of skin
Wind_Speed = 0;                 %m/s dear et al: H_a * V^.6, set high to approximate forced temp

Dial_A_Temp = false;            %Set true to enable power off when temp reached
Dial_A_Temp_Temp = 60;          %Turn off power when this temp reached
Dial_A_Temp_Depth = 3;          %mm, measure temp at this depth

Auto_Regulation = true;         %If true use temp to regulate power, else use Heat_Profile
Temp_Depth = 0;                 %Measure here for auto temp mm , negative for any depth
Auto_Temp = 75;                 %Regulate to this temp, if reached
Auto_Runtime = 30;              %Max time to supply power when in auto regulation mode
Heat_Profile(1:Run_Time) = 0;   %If auto regulation false, this power profile used, each entry is for 1 second period and has
values 0-1;

Alt_Dermis = false;             % try different dermis vars = see HEAT5
Water_Test = false;             %Used to check model - see code
Power_Adjust_Percent = 0;       %quick adjustment percent but easier to modify input power only for heat modeller supplied
power profile
Skin_Lesion = false;            %makes a square lesion in skin - see code
%--------------------------------------------------------------------------
if 1 == 0;                      %run EM solver
           [E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
           Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
       = Planewaveldv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);
end
if 1 == 0;                      %run heat solver
           [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
           Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
           Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                     ]...
                 = Heat5( ...
                       Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                 ...
                       Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                       Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                       Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                       Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,          ...
                       Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                       Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
```

```
                        );

end
%-------------------------------------------------------------------------
if 1 == 0;                      %Effect of Wind Speed Wind speed set to 1 to run
Wind_Vals = [1,10,100,250,500,1000,2000,5000,10000];  %Wind speed values
Wind_Runs = 9;
Wind_Effect = zeros(Wind_Runs,6);
for ii  = 1:Wind_Runs
Wind_Speed = Wind_Vals(ii);

Heat_Profile(1)=0.25; Heat_Profile(2)=0;    %1 second pulse at 1/4 power

      [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                  = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                  ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,          ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...
                        );

   Stability_Check1
   Stability_Check2
   Wind_Effect(ii,:) = [Wind_Speed, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];

end
   ff=figure;
   hold on
  plot(  log10(Wind_Effect(:,1)),(Wind_Effect(:,5)), 'color','b'    )
  plot(  log10(Wind_Effect(:,1)),(Wind_Effect(:,3)), 'color','r'    )
  grid on
  hold off
  title(['Effect of Wind Speed to Preserved (Red) and Ablated Depth (Blue) at ' num2str(f,'%5.2e') 'Hz' ])
  xlabel(' Log_1_0(Wind Speed) in m/s' );
  ylabel('Depth in mm');
  set(ff,'ylim',[0 8] );

end
%-------------------------------------------------------------------------
if 1 == 0;                      %Effect of Surface Temperature Regulation - compare heat profile by getting profile and
returning it The
Temp_Depth = 0;                 %Monitor on surface
Power_Runs = 4;
Input_Power = 40;
Power_Vals = [40,36,32,28];
Power_Effect_Fixed = zeros(Power_Runs,6);
Power_Effect_Auto = zeros(Power_Runs,6);

      [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                  = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                  ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,          ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...
                        );

   Stability_Check1
   Stability_Check2

   %Make second based heat profile
   [xx yy]=size(Heat_Profile_S);

   for ii = 1:1/Delta_T: yy;                    %Make average power over Delta_T cycles
       zz = (sum(Heat_Profile_S(ii:ii-1+(1/Delta_T)))) / (1/Delta_T) ;  % (1/Delta_T)  = number of samples in a second
       Heat_Profile(1 + ( (ii-1)*Delta_T)) = zz;
   end


   %now compare fixed profile to regulated temp
   for ii = 1: Power_Runs;
       Input_Power = Power_Vals(ii);
   %run it again
   Auto_Regulation = false;
      [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                  = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime, Alt_Dermis,             ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,          ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                    %Auto Desired
temperature, at depth, desired temp
                        );
    Power_Effect_Fixed(ii,:) = [Input_Power, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];

  %now in auto
  Auto_Regulation = true;
     [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
```

```
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                    = Heat5( ...
                          Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis ,                  ...
                          Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                          Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                          Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                          Force_Aperture_Temp, Force_T,Ambient_T,Wind_Speed ,           ...
                          Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                          Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                       %Auto Desired
temperature, at depth, desired temp
                          );
   Power_Effect_Auto(ii,:) = [Input_Power, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];
   end

   %plot
  ff=figure;
   hold on
  plot(  (Power_Effect_Fixed(:,1)),(Power_Effect_Fixed(:,5)), 'color','b'    )
  plot(  (Power_Effect_Auto(:,1)) ,(Power_Effect_Auto(:,5)), 'color','r'    )
  grid on
  hold off
  TEXT1 = ['Effect of power variation to ablation depth at ' num2str(f,'%5.2e') 'Hz'];
  TEXT2 = ['Fixed power profile in blue, Surface temperature regulation in red'];
  title( {[TEXT1];[TEXT2]} );
  xlabel('Incident Power in Watts' );
  ylabel('Ablation depth in mm');
  set(gca,'Xdir','reverse');
  hold off
end
%-------------------------------------------------------------------------
if 1==  0;                      %get times required for depths vs freq
   Time_Result = zeros(3,7,8);
   F_Vals = [0,1,2];
 for ff = 1:3
    Type_F = F_Vals(ff);
    [E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
           Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
        = Planewave1dv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);

    Dial_A_Temp = true;             %Set true to enable power off when temp reached
    Dial_A_Temp_Temp = 60;          %Turn off power when this temp reached
    Dial_A_Temp_Depth = 0;          %mm, measure temp at this depth
    Dial_Temp_Vals = [0, 0.5, 1, 1.5, 2, 2.5, 3]
    for ii = 1:7
        Dial_A_Temp_Depth = Dial_Temp_Vals(ii)  ; % 7 of
       [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                    = Heat5( ...
                          Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                  ...
                          Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                          Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                          Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                          Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,          ...
                          Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                          Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                       %Auto Desired
temperature, at depth, desired temp
                          );
    Time_Result(ff,ii,:) = ...
                        [ Type_F, Dial_A_Temp_Depth, Dial_A_Temp_Time, Found_Ablation,
Ablation_Depth,Found_Damage,OK_Damage_Depth,Total_Energy];
    Dial_A_Temp_Depth
    Dial_A_Temp_Time
    end
 end

 ff=figure;
 jj=0;          %work out how much to plot , Dial_A_Temp_Time = 0 means it didnt reach it
 kk=0;
 ll=0;
 for zz = 1:7
   if  (Time_Result(1,zz,3)) >0
       jj = jj+1;
   end
   if  (Time_Result(2,zz,3)) >0
       kk = kk+1;
    end
   if  (Time_Result(3,zz,3)) >0
       ll = ll+1;
     end
 end
 hold on
  if jj>0
 plot(  (Time_Result(1,1:jj,2)),(Time_Result(1,1:jj,3)), 'color','r'    ); % 1 is 14.5
  end
  if kk>0
 plot(  (Time_Result(2,1:kk,2)),(Time_Result(2,1:kk,3)), 'color','b'    ); % 2 is 5.8
  end
  if ll>0
 plot(  (Time_Result(3,1:ll,2)),(Time_Result(3,1:ll,3)), 'color','k'    ); % 2 is 5.8
  end
 hold off
 grid on
 set(gca,'ytick',[0 1 2 3 4 5 6 7 8 9 10 15 20 25]);
 title('Time Required vs Ablation depth for 5.8(blue), 14.5(red) and 24.125GHz(black)')
 xlabel('Ablation Depth in mm');
 ylabel('Time in seconds');

 ff2 = figure;      %Plot damage vs required ablation depth
 hold on
  if jj>0
```

```
 plot(  (Time_Result(1,1:jj,2)),(Time_Result(1,1:jj,7)), 'color','r'    ); % 1 is 14.5
  end
  if kk>0
 plot(  (Time_Result(2,1:kk,2)),(Time_Result(2,1:kk,7)), 'color','b'    ); % 2 is 5.8
  end
  if ll>0
 plot(  (Time_Result(3,1:ll,2)),(Time_Result(3,1:ll,7)), 'color','k'    ); % 2 is 5.8
  end
 hold off
 grid on
 title('Damage depth vs Ablation depth for 5.8(blue), 14.5(red) and 24.125GHz(black)')
 xlabel('Ablation Depth in mm');
 ylabel('Damage Depth in mm');


 ff3=figure;
 hold on
  if jj>0
 plot(  (Time_Result(1,1:jj,2)),(Time_Result(1,1:jj,8)), 'color','r'    ); % 1 is 14.5
  end
  if kk>0
 plot(  (Time_Result(2,1:kk,2)),(Time_Result(2,1:kk,8)), 'color','b'    ); % 2 is 5.8
  end
  if ll>0
 plot(  (Time_Result(3,1:ll,2)),(Time_Result(3,1:ll,8)), 'color','k'    ); % 2 is 5.8
  end
 hold off
 grid on
 title('Energy vs Ablation depth for 5.8(blue), 14.5(red) and 24.125GHz(black)')
 xlabel('Ablation Depth in mm');
 ylabel('Energy in Joules');

end
%-------------------------------------------------------------------------
if 1 == 0;                      % get ablation depths for fixed times w various fat
    Type_F = 1;                 %freq
    Dial_A_Temp = false;        %Set true to enable power off when temp reached
    Auto_Regulation = true;     %If true use temp to regulate power, else use Heat_Profile
    Temp_Depth = 0;             %Measure here for auto temp mm , negative for any depth
    Auto_Temp = 75;             %Regulate to this temp, if reached
    Run_Time = 60;              %Thermal run time total
    Input_Power = 40;           %Incident power over aperture
    Use_Power_Profile = true;   %if true use input power profile from EM, else use calculated e-ax for skindry for whole area

    Depth_Result = zeros(3,3,8,9);
    Auto_Runtime_Vals = [1, 1.5, 2, 4, 5, 10, 13.5, 15]; %8 of
    Fat_Thickness_Vals = [0,0.001,0.002]
    Dermis_Thickness_Vals = [0.001,0.002,0.003]


   for kk=1:3
   Dermis_Thickness = Dermis_Thickness_Vals(kk);

    for jj = 1:3
        Fat_Thickness = Fat_Thickness_Vals(jj);

        [E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
               Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
               = Planewave1dv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);
  %-------------------------------------------------------------------------
        for ii = 1:8            %Different times - no EM field changes within times
            Auto_Runtime = Auto_Runtime_Vals(ii); % 8 of

        [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
        Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
        Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                    ]...
                = Heat5( ...
                       Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                  ...
                       Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                       Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                       Water_Test,Plots_At_Runtime,Skin_Lesion,               ...
                       Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,           ...
                       Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                       Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                       %Auto Desired
temperature, at depth, desired temp
                       );

            Depth_Result(kk,jj,ii,:) = ...
              [ Type_F,Auto_Runtime, Dermis_Thickness, Fat_Thickness, Found_Ablation,
Ablation_Depth,Found_Damage,OK_Damage_Depth,Total_Energy];

        end
   %-------------------------------------------------------------------------
     end
   end

   figure
   for jjj=1:3
      for kkk = 1:3
        hold on
        plot(  squeeze(Depth_Result(jjj,kkk,:,2)),squeeze(Depth_Result(jjj,kkk,:,8)), 'color','r'    );
        plot(  squeeze(Depth_Result(jjj,kkk,:,2)),squeeze(Depth_Result(jjj,kkk,:,6)), 'color','b'    );
        hold off
      end
   end
   grid on
   TEXT1 = ['Ablation Depth (Blue) and Damage Depth (Red) at ' num2str(f,'%5.1e') 'Hz'];
   TEXT2 = ['For Fat thickness 0,1,2mm and Dermis thickness 1,2,3mm'];
   title ( {  [TEXT1];[TEXT2]  });
   grid on
   xlabel('Time in seconds');
   ylabel('Depth in mm');
    set(gca,'xtick',[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]);
```

```
end
%------------------------------------------------------------------------
if 1 == 0;                      %Effect of forced temp set to 1 to run
Auto_Regulation = true;         %If true use temp to regulate power, else use Heat_Profile
Temp_Depth = -1;                %Measure here for auto temp mm , negative for any depth
Auto_Temp = 75;                 %Regulate to this temp, if reached
%Force_Vals = [-10,0,10,20,30,40,50,60,70];  %Alternate
Force_Vals = [0,10,20,30,40,45,50,55,60,65,70];
Force_Runs = 11;
Force_Effect = zeros(Force_Runs,6);
Force_Aperture_Temp = true;
Type_F = 2;

               [E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
               Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f]    ...
    = Planewave1dv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);

for ii  = 1:Force_Runs
Force_T = Force_Vals(ii);

      [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                    ]...
                  = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime, Alt_Dermis,                 ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,           ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
                        );

   Stability_Check1
   Stability_Check2
   Force_Effect(ii,:) = [Force_T, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];

end
   ff=figure;
   hold on
  plot(  (Force_Effect(:,1)),(Force_Effect(:,5)), 'color','b'    )
  plot(  (Force_Effect(:,1)),(Force_Effect(:,3)), 'color','r'    )
  grid on
  hold off
  TEXT1 = ['Effect of Forced Temperature Bias to'] ;
  TEXT2 = ['Preserved (Red) and Ablated Depth (Blue) at ' num2str(f,'%5.2e') 'Hz'];
  title({ [TEXT1] [TEXT2] });
  xlabel(' Aperture Temperature in °C' );
  ylabel('Depth in mm');
  ylim([2 8]);

end
%------------------------------------------------------------------------
if 1 == 0;                      %Effect of Surface Temperature Regulation - compare different skin vars
Temp_Depth = 0;                 %Monitor on surface, try different K, Cp and p
Alt_Dermis = false;
Skin_Runs = 1;
Skin_Effect_Fixed = zeros(Skin_Runs,6);
Skin_Effect_Auto = zeros(Skin_Runs,6);

           [E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...       %Get EM profile
           Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
       = Planewaveldv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);


      [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                    ]...
                  = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                  ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,           ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
                        );

   Stability_Check1
   Stability_Check2

   %Make second based heat profile
   [xx yy]=size(Heat_Profile_S);

   for ii = 1:1/Delta_T: yy;                    %Make average power over Delta_T cycles
       zz = (sum(Heat_Profile_S(ii:ii-1+(1/Delta_T)))) / (1/Delta_T) ;  % (1/Delta_T)  = number of samples in a second
       Heat_Profile(1 + ( (ii-1)*Delta_T)) = zz;
   end


   %now compare fixed profile to regulated temp
   for ii = 1: Skin_Runs;
       Alt_Dermis = true;
   %run it again
   Auto_Regulation = false;

            [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
            Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
```

```
            Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                    = Heat5( ...
                            Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime, Alt_Dermis,            ...
                            Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                            Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                            Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                            Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,           ...
                            Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                            Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
                            );
    Power_Effect_Fixed(ii,:) = [Input_Power, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];

  %now in auto
  %Alt_Dermis = true;
  Auto_Regulation = true;
     [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                    = Heat5( ...
                            Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis ,                ...
                            Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                            Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                            Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                            Force_Aperture_Temp, Force_T,Ambient_T,Wind_Speed ,           ...
                            Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                            Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
                            );
    Power_Effect_Auto(ii,:) = [Input_Power, Found_Damage, OK_Damage_Depth, Found_Ablation, Ablation_Depth, Total_Energy ];
    end

    %plot
  ff=figure;
    hold on
  plot(  (Skin_Effect_Fixed(:,1)),(Skin_Effect_Fixed(:,5)), 'color','b'    )
  plot(  (Skin_Effect_Auto(:,1)) ,(Skin_Effect_Auto(:,5)), 'color','r'    )
  grid on
  hold off
  TEXT1 = ['Effect of Dermis thermal parameter variation to ablation depth at ' num2str(f,'%5.2e') 'Hz'];
  TEXT2 = ['Fixed power profile in blue, Surface temperature regulation in red'];
  title( {[TEXT1];[TEXT2]} );
  xlabel('Incident Power in Watts' );
  ylabel('Ablation depth in mm');
  set(gca,'Xdir','reverse');
  hold off
end
%-------------------------------------------------------------------------
if 1 == 1;                      %Short pulse to see heating
Type_F = 0;
Run_Time = 10;
Auto_Regulation = false;
Heat_Profile(1)=0.25; Heat_Profile(2)=0;

[E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
            Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
        = Planewave1dv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);


         [Skin_er, Skin_s, f, Ptot, EF_1, Analytic_Input_Power, Skin_C, Dial_A_Temp_Result,SAR_Basic,Power_Basic, ...
      Dial_A_Temp_Time, Heat_Profile_S, Dial_A_Temp_Tracking, Time_Line,Stability_Check1,Stability_Check2,Cem43,...
      Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy...
                        ]...
                    = Heat5( ...
                            Delta_T, Delta_Len, Type_F, Input_Power, Auto_Runtime,Alt_Dermis,                ...
                            Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness, ...
                            Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,        ...
                            Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                            Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed ,           ...
                            Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth,Run_Time, ...
                            Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile   ...                      %Auto Desired
temperature, at depth, desired temp
                            );



end
%%
```

Appendix 1F Planewave1dv3.m EM modeller

Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has
not been verified as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
%Simulate 1D wave travelling along x axis with various Permittivity and
% Conductivity to get an idea of E intensity
%Excitation is soft source immidiately adjacent to PMC at a distance from
%skin surface. Output is E field and P power profiles which need to be
%scaled for further use.
%All wavelengths in given materials have been checked for > 10 waves with
%cell size of 0.1mm
%Simulation runs for chosen number of cycles and the captures E Field and
%Power loss in the last cycle.
%
%PH 26-9-2016
%-----------------------------------------------------------------------
function[E_Fieldp10,E_Fieldp25,Power_Loss_1Cyclep10,Power_Loss_1Cyclep25, ...
            Energy_Loss_1Cycle,Conductivity_Vec,EM_Delta_Len,EM_Delta_T,f] ...
        = Planewaveldv3(Type_F,Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness);

e_0 = 8.854e-12;
mu_0 = 4*pi*1e-7;
c= 1/sqrt(e_0*mu_0);

EM_Delta_T = 0.25e-13;
EM_Delta_Len = 0.0001;
Cycles = 40;

Capture_Start_Cycle = Cycles-2;
Capture_End_Cycle = Capture_Start_Cycle+1;
%
Total_Thickness = 0.050;                %length of line we are simulating
Air_Thickness = 0.01;                   %Skin starts after this
%Epi_Thickness = 0.000;                 %Epidermis - same perm/cond values used as dermis
%Dermis_Thickness = 0.002               %Dermis thickness - same perm/cond values used as epidermis i.e. 'Skin'
Skin_Thickness  = Epi_Thickness+Dermis_Thickness;               %Thickness of skin total
%Fat_Thickness   = 0.001;               %m Thickness of fat
%Muscle_Thickness = .010;               %Thickness of muscle but not used - goes to end of boundary
%Tissue_Plot_Thickness = 0.010;         %Graph this depth
Smooth_Fat_Len = 0.00;                  %Linear transition for skin--fat
%
Pulse_Excitation = false;               % Set true to do 1 cycle at chosen F_Type
%
%Type_F = 0;                             %Frequency and tissue parameter chooser for testing
%
if Type_F == 0;                         % 14.5e9 GHz
f= 14.5e9;                              %Freq
Skin_er = 26.88;                        %Rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;                 %Abs permittivity
Skin_s = 13.27;                         %Conductivity 13.27 14.5G, 3.717 5.8G
Skin_l = 0.00216;                       %Wavelength
%
Fat_e = e_0*4.29;
Fat_s = 0.9;
Fat_l = 0.00989;
%
Muscle_e = e_0*36.954;
Muscle_s = 17.194;
Muscle_l = 0.00327;                      %
end
if Type_F == 1;                         %5.8e9 GHz
f= 5.8e9;                               %freq
Skin_er = 35.114;                       %
Skin_e = Skin_er * e_0;                 %Abs permittivity
Skin_s = 3.717;                         %Conductivity 13.27 14.5G, 3.717 5.8G
Skin_l = 0.0086;                        %wavelength
%
Fat_e = e_0*4.95;
Fat_s = 0.293;
Fat_l = 0.0231;
%
Muscle_e = e_0*48.485;
Muscle_s = 4.9615;
Muscle_l = 0.00733;
end
if Type_F == 2;                         % 24.125e9 GHz
f= 24.125e9;                            %freq
Skin_er = 18.909;                       % skin rel perm
Skin_e = Skin_er * e_0;                 %
Skin_s = 22.944;                        % conductivity
Skin_l = 0.00264;
%
Fat_e = e_0*3.83;
Fat_s = 1.496;
Fat_l = 0.00628;
```

```
%
Muscle_e = e_0*27.294;
Muscle_s = 29.577;
Muscle_l = 0.00222;
end
if Type_F == 3;                         % 61.125e9 GHz
f= 61.125e9;                            %freq
Skin_er = 7.8501;                       %rel perm
Skin_e = Skin_er * e_0;                 %
Skin_s = 36.554;                        %conductivity
Skin_l = 0.0015;                        %wavelength
%
Fat_e = e_0*3.127;
Fat_s = 2.8428;
Fat_l=  0.00275;
%
Muscle_e = e_0*12.656;
Muscle_s = 53.227;
Muscle_l = 0.0012;
end
if Type_F == 4;                         % f= 2.56e9 GHz
f= 2.56e9;                              % freq
Skin_er = 37.887;                       % skin rel perm
Skin_e = Skin_er * e_0;                 %
Skin_s = 1.5163;                        % conductivity
Skin_l = 0.0188;
%
Fat_e = e_0*5.2687;
Fat_s = 0.10939;
Fat_l = 0.0509;
%
Muscle_e = e_0*52.595;
Muscle_s = 1.8147;
Muscle_l = 0.016;
end
if Type_F == 5;                         %TEST TEST TEST
Skin_er = 4.29;                         % fat at 14.5
Skin_e = Skin_er * e_0;                 %
Skin_s = 0.90;                          %
f= 14.5e9;                              %
Fat_e = e_0*4.29;
Fat_s = 0.9;
Muscle_e = e_0*36.954;
Muscle_s = 17.194;
end
if Type_F == 6;                         %TEST TEST TEST
Skin_er = 35; %20;                      %skin rel perm
Skin_e = Skin_er * e_0;
Skin_s = 0;                             %conductivity
f= 14.5e9;
Fat_e = e_0*35;   %4.29;
Fat_s =5; %5; %0.9;%14.5e9 or 5.8e9
Muscle_e = e_0*36.954;
Muscle_s = 17.194;
end
if Type_F>6                             %Error
    error('Invalid frequency selection');
end;

Time_Steps = fix(Cycles/(f*EM_Delta_T));
Length_Cells = fix(Total_Thickness/EM_Delta_Len);

E_Loss = 0;                             %Energy lost in run time
E_Loss2_1Cycle = 0;                     %Energy lost in run time

%Set up matrices
Null_M(1:Length_Cells) = 0;
E_Y = Null_M;
E_Y_P = Null_M;
H_Z = Null_M;
H_Z_P = Null_M;
SC1 = Null_M;
SC2 = Null_M;
Space_Eps = Null_M;
Space_Cond = Null_M;
RMS = Null_M;
XX = Null_M;
ZZ(1:Time_Steps) = 0;
SAR = Null_M;
jj=1;
Plot_P=0;                       %Counter for making runtim plot

Space_Eps(1:end) = e_0;         %Default
Space_Cond(1:end) = 0;

Air_Start_Cell = 1;
Air_Cells =  round(Air_Thickness/EM_Delta_Len);
Air_End_Cell = Air_Start_Cell+Air_Cells-1;                  %Actually the next cell
%
Skin_Start_Cell = Air_End_Cell+1;
Skin_Depth_Cells = round(Skin_Thickness/EM_Delta_Len);
Skin_End_Cell = Skin_Start_Cell+Skin_Depth_Cells-1;
%
Fat_Start_Cell = Skin_End_Cell+1;
Fat_Depth_Cells = round(Fat_Thickness/EM_Delta_Len);
Fat_End_Cell = Fat_Start_Cell+Fat_Depth_Cells-1;
%
Muscle_Start_Cell = Fat_End_Cell+1;
Muscle_Depth_Cells = round(Muscle_Thickness/EM_Delta_Len);
Muscle_End_Cell = Muscle_Start_Cell+Muscle_Depth_Cells-1;  %If used, go to end of boundary
%
Space_Eps(Skin_Start_Cell:Skin_End_Cell)  = Skin_e;
Space_Cond(Skin_Start_Cell:Skin_End_Cell) = Skin_s;
%
```

```
Tissue_End_Cell = Skin_Start_Cell+(Tissue_Plot_Thickness/EM_Delta_Len);

 if Fat_Depth_Cells > 0;                                  %smooth over several fat cells maybe
    Space_Eps(Fat_Start_Cell:Fat_End_Cell)  = Fat_e;
    Space_Cond(Fat_Start_Cell:Fat_End_Cell) = Fat_s;
 if Smooth_Fat_Len ~=0
Smooth_Cells=round(Smooth_Fat_Len/EM_Delta_Len);
    if Smooth_Cells < 2*Fat_Depth_Cells
        for iii = 1:Smooth_Cells-1
        Space_Eps(Fat_Start_Cell+iii-1)  = Space_Eps(Fat_Start_Cell-1)  - ( (Space_Eps(Fat_Start_Cell-1)  -
Space_Eps(Fat_Start_Cell+Smooth_Cells)) *iii   /Smooth_Cells);
        Space_Cond(Fat_Start_Cell+iii-1) = Space_Cond(Fat_Start_Cell-1) - ( (Space_Cond(Fat_Start_Cell-1) -
Space_Cond(Fat_Start_Cell+Smooth_Cells)) *iii   /Smooth_Cells);
        end
    end
end
 end


 if Muscle_Depth_Cells > 0
    Space_Eps(Muscle_Start_Cell:Muscle_End_Cell)  = Muscle_e;
    Space_Cond(Muscle_Start_Cell:Muscle_End_Cell) = Muscle_s;
 end

 if Muscle_Start_Cell <= Length_Cells;              %Fill rest of space with skin
    Space_Eps(Muscle_Start_Cell:end)  = Muscle_e;
    Space_Cond(Muscle_Start_Cell:end) = Muscle_s;
 end

if Muscle_End_Cell>Length_Cells
    error('Thicknesses of individual layers add up to greater than simulation depth of 30mm');
end


% DO NOT PUT ANY VAR CHANGES BELOW
SC1 = SC1+  ((2.*Space_Eps) - (EM_Delta_T.*Space_Cond))./ ((2.*Space_Eps) + (EM_Delta_T.*Space_Cond));
SC2 = SC2+  (2*EM_Delta_T)./ ((2.*Space_Eps) + (EM_Delta_T.*Space_Cond));

%--------------------------Run simulation-------------------------
for ii = 1:Time_Steps

E_Y(2:end) = (SC1(2:end).* E_Y_P(2:end)) -( (SC2(2:end)./EM_Delta_Len).*(H_Z(2:end)-H_Z(1:end-1)));
H_Z(1:end-1) =               H_Z_P(1:end-1) - ( (EM_Delta_T/(mu_0*EM_Delta_Len)).*(E_Y(2:end) - E_Y(1:end-1)) );

if Pulse_Excitation == true            %Actually a 1 cycle 'pulse' or CW wave
    if ii   <= 1+(1/(f*EM_Delta_T));
     E_Y(5)=  E_Y(5)+ ((1/100)*sin(2 * pi * f * EM_Delta_T*(ii-1)) );
    else
    %H_Z(1)=0;                          %PMC
    %E_Y(1)=0;                          %PEC
    end

    if ii == fix(1+(1/(f*EM_Delta_T)));
    E_Space = sum(  ((Space_Eps.*E_Y.^2)+(mu_0*H_Z.^2)) /2  ) + E_Loss;  %add losses in case we are pulsing into lossy media
    E_In;
    end

else
    E_Y(2)=  E_Y(2)+ ((1/20)*sin(2 * pi * f * EM_Delta_T*(ii-1)) );     %Else soft source CW wave
end

H_Z(1) = 0;                            %PMC at start
%H_Z(end-1)=0;                         %PMC at end
%E_Y(end) = 0;                         %PEC at end (default)
H_Z_P = H_Z;
E_Y_P = E_Y;

E_Loss = E_Loss+ (sum( ( ((( (E_Y.^2) .* Space_Cond))*EM_Delta_T) ))) ;       %total energy dissipated instantaneous power so
no 1/2

if (f * EM_Delta_T*ii) >= Capture_Start_Cycle   %Capture power profile after system is settled - for CW only
 if (f * EM_Delta_T*ii) < Capture_End_Cycle
    RMS = RMS+(E_Y.*E_Y);
    %
    %
    ZZ(jj)=E_Y(1);
    jj=jj+1;
    E_Loss2_1Cycle = E_Loss2_1Cycle + (sum( (     ((( (E_Y.^2) .* Space_Cond))*EM_Delta_T) ))) ;
 end
end

Plot_P = Plot_P+1;
    if Plot_P>=500;                    %Do less plots to keep execution time down
        Plot_P=0;
    figure(1);
    hold off
    plot(E_Y,'color','b');
    hold on
    plot(50*H_Z,'color','r');
    hold off
    axis([1 Length_Cells -1 +1]);
    grid on
    drawnow
    end
end
%---------------- End of simulation loop--------------------------------

    EE2=(RMS.*(2*f*EM_Delta_T)).^0.5;  %integral of sin^2 to get back E
    yy=linspace(0, Tissue_Plot_Thickness*1000 , 1+(Tissue_Plot_Thickness)/EM_Delta_Len );
    figure;
    plot(yy,(EE2(Skin_Start_Cell:Tissue_End_Cell)./max(EE2(Skin_Start_Cell:Tissue_End_Cell))));

    S_Start = 0;
    F_Start = Skin_Thickness*1000;
```

```
    M_Start = 1000*(Skin_Thickness+Fat_Thickness);
    P_End =   1000*(Tissue_Plot_Thickness);

    text(S_Start,1.05,' Skin');

    if Fat_Depth_Cells> 0
        if F_Start<M_Start
            Pat=patch( [F_Start M_Start M_Start F_Start], [0 0 1.1 1.1],'y' );
            set(Pat,'FaceAlpha',0.25);
            text(F_Start,1.05,' Fat');
        end
    end

    if Muscle_Depth_Cells> 0
        if M_Start<P_End
            text(M_Start,1.05,' Muscle');
            Pat=patch( [M_Start P_End P_End M_Start], [0 0 1.1 1.1],'g' );
            set(Pat,'FaceAlpha',0.25);
        end
    end
%    keyboard  %Debug
    title([ 'Normalized E field (Blue) and Power loss (Black) at ', num2str(f,'%5.2e'),' Hz']);
    vv = axis;
    vv(4)=1.1;
    vv(3)=0;
    axis(vv);
    hold on
    plot(yy,
((EE2(Skin_Start_Cell:Tissue_End_Cell).^2).*Space_Cond(Skin_Start_Cell:Tissue_End_Cell))./max(((EE2(Skin_Start_Cell:Tissue_En
d_Cell)).^2).*Space_Cond(Skin_Start_Cell:Tissue_End_Cell)),'color','k' );
    xlabel('Depth from Skin Surface in mm');
    ylabel('Normalised to maximum value');
    set(gca,'ytick',([ 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1]))
    hold off
    grid on;
    grid minor;
%keyboard    %Debug

if Pulse_Excitation==true;      %Display vars
E_Space
E_Loss
end

E_Fieldp10 = EE2(Skin_Start_Cell:end);
Energy_Loss_1Cycle = E_Loss2_1Cycle;
Conductivity_Vec = Space_Cond(Skin_Start_Cell:end);

Power_Loss_1Cyclep10 = ((EE2(Skin_Start_Cell:end).^2).*Space_Cond(Skin_Start_Cell:end));

%Scale 0.1mm output to 0.25mm
[SizeX,SizeY]= size(Power_Loss_1Cyclep10);
Power_Loss_1Cyclep05(1:size(SizeY*2)) = 0;          %make power profile in 0.05 mm steps from 0.1mm steps
E_Fieldp05(1:size(SizeY*2)) = 0;
for kk = 1: SizeY;
Power_Loss_1Cyclep05((kk*2)-1) = Power_Loss_1Cyclep10(kk);
Power_Loss_1Cyclep05((kk*2))   = Power_Loss_1Cyclep10(kk);
E_Fieldp05((kk*2)-1) = E_Fieldp10(kk);
E_Fieldp05((kk*2))   = E_Fieldp10(kk);
end

ll = 1;
for kk = 1: 5: SizeY*2;                %Scale power profile for use by heat solver
Power_Loss_1Cyclep25(ll) = (Power_Loss_1Cyclep05(kk)  + Power_Loss_1Cyclep05(kk+1) ...
                         + Power_Loss_1Cyclep05(kk+2)+ Power_Loss_1Cyclep05(kk+3) + Power_Loss_1Cyclep05(kk+4) )/5;

E_Fieldp25(ll) = (E_Fieldp05(kk) + E_Fieldp05(kk+1) + E_Fieldp05(kk+2) + E_Fieldp05(kk+3) + E_Fieldp05(kk+4))/5;

ll=ll+1;
end


end
```

Appendix 1G Heat5.m Thermal Modeller

Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has
not been verified as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
%Simple model for solving heat equation
%
% Based on FDTD computation of temperature rise in the human head for
% portable phones Wang & Fujiware 1999 IEEE transactions on Microwave
% Theory Vol 47 no 8.
%
% Also includes method by Bernardi et al Specific Absorption Rate and
% Temperature Elevation in a subject exposed in the Far-Field of
% Radio-Frequency Sources Operating in the 10-900MHz Range, IEEE
% Transactions on Biomedical Engineering Vol 50 no 3 2003
%
%--------------------------------------------------------------------------
%Revision History
%
%16-8-2013 Added second heating profile for mixed modes , note that the addition is
%not strictly correct as the absolute value is added - in real life this
%would be equivalent to multiplexing modes - otherwise the efields would
%subtract at certain positions
%
%PH 31-7-2013
% rev 2, 2-12-2013 to have automatic heat application to achieve set temp
% at a set depth
%
%PH 15-9-2016 various changes to graph labelling and flags at start of
%program to select skin thermal parameters
%
%PH 18-9-2016 addition calculation method based on Bernardi 2003 to check
%results
%
%PH 19-10-2016 Changed erroneous H_a from 10 to 10.5 - will not make any
%difference to forced or convective for wind speed >0, but probably very slight
%difference in other sims
%-------------------------------------------------------------------------------------------
function[Skin_er,Skin_s,f,Ptot,EF_1,Analytic_Input_Power,Skin_C,Dial_A_Temp_Result,SAR_Basic,Power_Basic,...
                        Dial_A_Temp_Time,Heat_Profile_S Dial_A_Temp_Tracking, Time_Line
Stability_Check1,Stability_Check2,Cem43,...
                        Found_Damage,OK_Damage_Depth,Found_Ablation,Ablation_Depth,Total_Energy ...
                        ]...
                = Heat5( ...
                        Delta_T, Delta_Len, Type_F, Input_Power,Auto_Runtime,Alt_Dermis,               ...
                        Epi_Thickness,Dermis_Thickness,Fat_Thickness,Muscle_Thickness,Tissue_Plot_Thickness,...
                        Power_Loss_1Cyclep10,Power_Loss_1Cyclep25,Use_Power_Profile,     ...
                        Water_Test,Plots_At_Runtime,Skin_Lesion,              ...
                        Force_Aperture_Temp,Force_T, Ambient_T,Wind_Speed,          ...
                        Power_Adjust_Percent,Dial_A_Temp,Dial_A_Temp_Temp,Dial_A_Temp_Depth, Run_Time, ...
                        Auto_Regulation, Temp_Depth,Auto_Temp, Heat_Profile     ...                      %Auto Desired
temperature, at depth, desired temp
                        )

e_0 = 8.854e-12;
mu_0 = 4*pi*1e-7;
c= 1/sqrt(e_0*mu_0);
EF_1 = 3200;                %V/M of excitation, may be overriden if input power supplied
EF_2 = 1400;
Skin_X = 0.034;             %Total width  5mm margin around applicator
Skin_Y = 0.028;             %Total height
Skin_Z = 0.015;             %Total depth
Applicator_X = 0.024;       %heated area centred Width
Applicator_Y = 0.018;       %heated area centred height
%-------------------------------------------------------
%FOR TESTING

Constant_E_Test = false;    %Set = true for constant E field

%Water_Test = false;        %set = true to test basic operation with water for all medium params
%Plots_At_Runtime = true;   %Set true for making plots during the run
%Skin_Lesion = false;       %set true for cheng/herman 2011 skin lesion
%Plus_5Power = false;       %set true to get 5% more power
%Plus_10Power = false;      %set true to get 10% more power
%Minus_5Power = false;
%Minus_10Power= false;
%Power_Adjust_Percent = 0;  %+ - power adjust
%Hi_Res_Half = false;       %true means Delta_len and Delta_T are halved

%Force_Aperture_Temp = false;  % set true to force exposed skin to amb temp

%Epidermis = false;         % set true to get 1 cell at z=2 for epidermis
Numerical_Method = 0;       % main method = 0, alternative = 1..
%-------------------------------------------------------
%added 11/9/16 to dial a temp at a depth
%Dial_A_Temp = true;        %Set true if we want this
```

```
%Dial_A_Temp_Depth = 2;     %Where we are looking in mm depth
%Dial_A_Temp_Temp = 60;     %Temp we are after
Dial_A_Temp_Time = 0;       %How long it took
Dial_A_Temp_Ok_Temp = 50;   %will find out depth under this temp thats ok
Dial_A_Temp_Stop = false;   %dont stop yet = flag used by program
%-------------------------------------------------------
%Type_F = 0;                % Skin Parameters vs Frequency  0 = 14.5, 1 = 5.8 , 2 = 24.125G, 3 = 61.125, 4= 2.56G
if Type_F==0
Skin_er = 26.88;            % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 13.27;             % conductivity 13.27 14.5G, 3.717 5.8G
f= 14.5e9;                  %14.5e9 or 5.8e9
end
if Type_F == 1
Skin_er = 35.114;           % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 3.717;             % conductivity 13.27 14.5G, 3.717 5.8G
f= 5.8e9;                   %14.5e9 or 5.8e9
end
if Type_F == 2
Skin_er = 18.909;           % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 22.944;            % conductivity 13.27 14.5G, 3.717 5.8G
f= 24.125e9;                %14.5e9 or 5.8e9
end
if Type_F == 3
Skin_er = 7.8501;           % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 36.554;            % conductivity 13.27 14.5G, 3.717 5.8G
f= 61.125e9;                %14.5e9 or 5.8e9
end
if Type_F==4  ;             %2.56e9
Skin_er = 37.887;           % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 1.5163;            % conductivity 13.27 14.5G, 3.717 5.8G
f= 2.56e9;                  %14.5e9 or 5.8e9
end
if Type_F==5  ;             %TEST TEST TEST Fat
Skin_er = 4.29;             % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 0.90;              % conductivity 13.27 14.5G, 3.717 5.8G
f= 14.5e9;                  %14.5e9 or 5.8e9
end
if Type_F==6  ;             %TEST TEST TEST Fat
Skin_er = 4.95;             % skin rel perm 26.88 14.5G, 35.114 5.8G
Skin_e = Skin_er * e_0;     %
Skin_s = 0.293;             % conductivity 13.27 14.5G, 3.717 5.8G
f= 5.8e9;                   %14.5e9 or 5.8e9
end
if Type_F>4                 %Error
    error('Invalid frequency selection');
end;
%-------------------------------------------------------
%Following is used when we know input power and want to calculate E for use
%with simple model based on prop constant
%uses formulae from Electromagnetic field theory fundamentals by Guru and
%Hiziroglu 2004, Cambridge University press, pg 367-368 , 8.57, 8.60a
Neta =  (  (1i*2*pi*f*mu_0)/(Skin_s+ (1i*2*pi*f*Skin_e)))^0.5;
Neta_Abs = abs(Neta);
Neta_Angle = atan(imag(Neta)/real(Neta));
Calc_E = ((Input_Power * 2*Neta_Abs) / ((Applicator_X*Applicator_Y*cos(Neta_Angle))) )^0.5;
Analytic_Input_Power = (Applicator_X*Applicator_Y*(EF_1^2)*cos(Neta_Angle))  /(2*Neta_Abs);
if Input_Power ~= 0
    EF_1 = Calc_E;
end
%-------------------------------------------------------
Found_Damage = false;       %Cem43 damage flag
OK_Damage_Depth = 0;        %where skin is damaged to, Cem43 value of 40 for fat or 80 elsewhere see end
Found_Ablation = false;
Ablation_Depth = 0;         %60 deg C is threshold
%-------------------------------------------------------
%{
if Hi_Res_Half ==true       %maybe use twice resolutio
Delta_T =  0.05;            %Time step 0.05?
Delta_Len = 0.00025;        %cell size in m .00025
else
Delta_T =  0.008;           %Time step
Delta_Len = 0.0001 ;%25;    %cell size
end
%}
%-------------------------------------------------------
%Run_Time =  3;             % Seconds total run time
Stop_Time = 5;              %
%Ambient_T =  20;           %ambient temp in C at front of aperture, might force to this temp
Applicator_Temp =  20;      %temperature of applicator body
H_a = 10.5;                 %W/m^2 C convection to ambient 10.5 is free air

%Epidermis_Depth = 0.0001;  %depth in metres
%Dermis_Depth = 0.002;
%Fat_Depth = 0.002;
%Muscle_Depth = (0.015 - sum of above) To end of space

%Conductivity
Skin_K_Const = 0.5;         %conductivity 0.5
Epidermis_K_Const = 0.322;
Dermis_K_Const = 0.5;
Fat_K_Const = 0.22;
Muscle_K_Const = 0.6;
Water_K_Const = 0.598;      %20C
http://www2.bren.ucsb.edu/~dturney/WebResources_13/WaterSteamIceProperties/PropOfWaterFrom0to100Celcius.pdf

%Specific Heat
Skin_C_Const = 3500;        %specific heat
Epidermis_C_Const = 3569;
```

```
Dermis_C_Const = 3500;
Fat_C_Const = 2300;
Muscle_C_Const = 3500;
Water_C_Const = 4181;

%Perfusion
Skin_b_Const = 8652 ;       %perfusion
Epidermis_b_Const = 0;
Dermis_b_Const = 8652;
Fat_b_Const = 816;
Muscle_b_Const =3488;
Water_b_Const = 0;

%Density
Skin_p_Const = 1010;        %skin density
Epidermis_p_Const = 1200;
Dermis_p_Const = 1010;
Fat_p_Const = 920;
Muscle_p_Const =  1040;
Water_p_Const = 998;

Body_T    = 37;             %Body temp in C
Blood_T   = 37;             %Blood temp


if Alt_Dermis == true;      %Simple method to compare another set skin thermal params
 Skin_K_Const = 0.445
 Skin_C_Const = 3300
 Dermis_p_Const = 1200
 beep
end

%Wind_Speed = 0;            %used to modify convection coefficient M/S Dear et al 1997
  if Wind_Speed ~= 0
      H_a = 10.3*Wind_Speed^.6;
  end;

%-------------------------------------------------------------
%Variables for automatic temperature regulation
%Auto_Regulation =  true ;                %automatic heat regulation if false then fixed profile
%Temp_Depth = 0.2                         %in mm, 0 = surface where to measure for auto
%Auto_Temp = 72;                          %Auto Desired temperature
%Auto_Runtime = 30;                      %time to run/HEAT IT FOR IN AUTO
Auto_Temp_Upper = Auto_Temp+1;           %upper limit hysteresis for on/off control Turn off power if above this
Auto_Temp_Lower = Auto_Temp;           %Lower limit hysteresis for on off control Turn on power if below this
%-------------------------------------------------------------
%Heat_Profile(1:Run_Time) = 0;            %heat profile
%following profiles created by experiment to keep constant temp
if Auto_Regulation == false
 %Heat_Profile(1:30) = [1,1,1,1,1,1,1,1,1,1 ,1,1,1,1,1,1,1,1,1,1 ,0,0,0,0,0,0,0,0,0,0];
 %Heat_Profile(1:30) = [1,1,1,1,1,1,1,1,1,1 ,1,1,1,0,1,1,0,1,1,0 ,1,1,0,1,1,0,0,0,0,0];
 %New because power profile moved in by 1 cell because of air
 %this profile good for 72C approx at 3200v/m with std ifac values and wang skin
 %Heat_Profile(1:30) = [0.99,0.99,0.65,0.20,0.20,0.15,0.15,0.15,0.15,0.12 ,0.11,0.11,0.11,0.11,0.11,0.11,0.10,0.10,0.10,0.10
,0.09,0.09,0.09,0.08,0.08,0.08,0.08,0.07,0.07,0];
 % Heat_Profile(1:30) = [1,1,1,0.6,0.3,0.3,0.3,0.3,0.3 ,0.25,0.25,0.25,0.25,0.25,0.25,0.25,0.25,0.25,0.25
,0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2,0.2];
end
%-------------------------------------------------------------
Attenuation_C = real( ((1i*2*pi*f*mu_0)*(Skin_s + (1i*2*pi*f*Skin_e)))^0.5 ); %Electromagnetic field theory fundamantals Guro
& Hiziroglu
%                                                                                %2004 Cambridge University Press pg366 8.50
Skin_Depth = 1/Attenuation_C;
EF1 = EF_1 * ((1+Power_Adjust_Percent/100)^0.5);

TE1_X = 0;                                  %transverse electric mode number for heat profile 0=uniform
TE2_X = 2;                                  %additional transverse electric mode number of modes for heat profile 0=none or
TE1_X = 0 = none
%-----------------------------------------------------
%-----------------------------------------------------
%% set up matrices
Cells_X = Skin_X/Delta_Len;
Cells_Y = Skin_Y/Delta_Len;
Cells_Z = Skin_Z/Delta_Len;

Centre_X = round(Cells_X/2);
Centre_Y = round(Cells_Y/2);

App_Cells_X = round(Applicator_X/Delta_Len);
App_Cells_Y = round(Applicator_Y/Delta_Len);

App_Lower_X = round(Centre_X-(App_Cells_X/2));
App_Upper_X = round(Centre_X+(App_Cells_X/2)-1);
App_Lower_Y = round(Centre_Y-(App_Cells_Y/2));
App_Upper_Y = round(Centre_Y+(App_Cells_Y/2)-1);

Time_Steps = Run_Time/Delta_T;
Stop_Steps = Stop_Time/Delta_T;

Dial_A_Temp_Tracking = zeros(Cells_Z,1);        %used to find max unablated depth BEWARE IF APERTURE FACE IS FORCED TEMP

Skin_Power =  zeros(Cells_X,Cells_Y,Cells_Z);   %power dissipation per cell
Skin_Temp_C = zeros(Cells_X,Cells_Y,Cells_Z);   %current temperature
Skin_Temp_P = zeros(Cells_X,Cells_Y,Cells_Z);   %previous temperature

Skin_K = zeros(Cells_X,Cells_Y,Cells_Z);        %thermal conductivity W/m C
Skin_C = zeros(Cells_X,Cells_Y,Cells_Z);        %specific heatJ/kg C
Skin_b = zeros(Cells_X,Cells_Y,Cells_Z);        %perfusion W/m^3 C
Skin_p = zeros(Cells_X,Cells_Y,Cells_Z);        %skin density kg/m^3

Power_Profile_EM = zeros(Cells_X,Cells_Y,Cells_Z); %used to make power/SAR from input power profile

Dial_A_Temp_Result = Skin_C;                    %Default for return call
```

```
E_Field = zeros(Cells_X,Cells_Y,Cells_Z);       %E strength in volts/m
SAR = zeros(Cells_X,Cells_Y,Cells_Z);           % Specific absorbion rate = Conductivity * E^2 / ( 2 * Density)

Time_Line = zeros(Time_Steps,Cells_Z);
Cem43(1:Cells_Z)=0;
%%
%set up skin as default
Skin_K(2:end-1,2:end-1,2:end-1) = Skin_K_Const; %conductivity  Defaults should be entirely overwritten
Skin_C(2:end-1,2:end-1,2:end-1) = Skin_C_Const; %specific heat
Skin_b(2:end-1,2:end-1,2:end-1) = Skin_b_Const; %perfusion
Skin_p(2:end-1,2:end-1,2:end-1) = Skin_p_Const; %skin density
%
%
Epi_Start_Cell = 2;                             %Start here, (1) is air/boundary
Epi_Depth_Cells = round(Epi_Thickness/Delta_Len);
Epi_End_Cell = Epi_Start_Cell+Epi_Depth_Cells-1;
%
Dermis_Start_Cell = Epi_End_Cell+1;
Dermis_Depth_Cells = round(Dermis_Thickness/Delta_Len);
Dermis_End_Cell = Dermis_Start_Cell+Dermis_Depth_Cells-1;
%
Fat_Start_Cell = Dermis_End_Cell+1;
Fat_Depth_Cells = round(Fat_Thickness/Delta_Len);
Fat_End_Cell = Fat_Start_Cell+Fat_Depth_Cells-1;
%
Muscle_Start_Cell = Fat_End_Cell+1;
%Muscle_Depth_Cells = round(Muscle_Thickness/Delta_Len);
%Muscle_End_Cell = Muscle_Start_Cell+Muscle_Depth_Cells-1;  %If used, go to end of boundary
Muscle_End_Cell = Cells_Z;
Muscle_Depth_Cells = Muscle_End_Cell-Muscle_Start_Cell+1;
%
if Epi_Depth_Cells > 0
Skin_K(2:end-1,2:end-1,Epi_Start_Cell:Epi_End_Cell) =           Epidermis_K_Const;   %Conductivity
Skin_C(2:end-1,2:end-1,Epi_Start_Cell:Epi_End_Cell) =           Epidermis_C_Const;   %Specific Heat
Skin_b(2:end-1,2:end-1,Epi_Start_Cell:Epi_End_Cell) =           Epidermis_b_Const;   %Perfusion
Skin_p(2:end-1,2:end-1,Epi_Start_Cell:Epi_End_Cell) =           Epidermis_p_Const;   %Density
end
%
if Dermis_Depth_Cells>0
Skin_K(2:end-1,2:end-1,Dermis_Start_Cell:Dermis_End_Cell ) =    Dermis_K_Const;
Skin_C(2:end-1,2:end-1,Dermis_Start_Cell:Dermis_End_Cell ) =    Dermis_C_Const;
Skin_b(2:end-1,2:end-1,Dermis_Start_Cell:Dermis_End_Cell ) =    Dermis_b_Const;
Skin_p(2:end-1,2:end-1,Dermis_Start_Cell:Dermis_End_Cell ) =    Dermis_p_Const;
end
%
if Fat_Depth_Cells>0
Skin_K(2:end-1,2:end-1,Fat_Start_Cell:Fat_End_Cell ) =          Fat_K_Const;
Skin_C(2:end-1,2:end-1,Fat_Start_Cell:Fat_End_Cell ) =          Fat_C_Const;
Skin_b(2:end-1,2:end-1,Fat_Start_Cell:Fat_End_Cell ) =          Fat_b_Const;
Skin_p(2:end-1,2:end-1,Fat_Start_Cell:Fat_End_Cell ) =          Fat_p_Const;
end
%
if Muscle_Depth_Cells>0
Skin_K(2:end-1,2:end-1,Muscle_Start_Cell:end-1 ) =              Muscle_K_Const;
Skin_C(2:end-1,2:end-1,Muscle_Start_Cell:end-1 ) =              Muscle_C_Const;
Skin_b(2:end-1,2:end-1,Muscle_Start_Cell:end-1 ) =              Muscle_b_Const;
Skin_p(2:end-1,2:end-1,Muscle_Start_Cell:end-1 ) =              Muscle_p_Const;
end
%
Skin_K(2:end-1,2:end-1,1) = Skin_K(2:end-1,2:end-1,2);      %Boundary convection uses nearest cell K
%
if Water_Test == true %OVERRIDE WITH WATER MATERIAL, NO PERFUSION
Skin_K_Const = Water_K_Const;   %conductivity
Skin_C_Const = Water_C_Const;   %specific heat
Skin_b_Const = Water_b_Const;   %perfusion
Skin_p_Const = Water_p_Const;   %skin density
end

%keyboard
%% Set up alternate matrices for method 1 Bernardi et al 2003
if Numerical_Method == 1
CO1 =     (H_a * Delta_T)./(Skin_C.*Skin_p.*Delta_Len);
CO2 = (Skin_b .* Delta_T)./(Skin_C.*Skin_p);
CO3 =           (Delta_T)./(Skin_C.*Skin_p);
CO4 = Delta_T./(Skin_C.*Skin_p.*(Delta_Len)^2);


Skin_K(:,:,1)=0; %air has no conductivity

CO_0IM = zeros(Cells_X,Cells_Y,Cells_Z);
CO_0IP = zeros(Cells_X,Cells_Y,Cells_Z);
CO_0JM = zeros(Cells_X,Cells_Y,Cells_Z);
CO_0JP = zeros(Cells_X,Cells_Y,Cells_Z);
CO_0KM = zeros(Cells_X,Cells_Y,Cells_Z);
CO_0KP = zeros(Cells_X,Cells_Y,Cells_Z);
NEXT   = zeros(Cells_X,Cells_Y,Cells_Z);


CO_0IM(2:end-1,2:end-1,2:end-1) = ...
(2*Skin_K(1:end-2,2:end-1,2:end-1).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(1:end-2,2:end-1,2:end-1)+ Skin_K(2:end-1,2:end-
1,2:end-1)).*CO4(2:end-1,2:end-1,2:end-1) ;

CO_0IP(2:end-1,2:end-1,2:end-1) = ...
  (2*Skin_K(3:end  ,2:end-1,2:end-1).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(3:end ,2:end-1,2:end-1)+ Skin_K(2:end-
1,2:end-1,2:end-1)).*CO4(2:end-1,2:end-1,2:end-1) ;

CO_0JM(2:end-1,2:end-1,2:end-1) = ...
  (2*Skin_K(2:end-1,1:end-2,2:end-1).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(2:end-1,1:end-2,2:end-1)+ Skin_K(2:end-
1,2:end-1,2:end-1)).*CO4(2:end-1,2:end-1,2:end-1) ;

CO_0JP(2:end-1,2:end-1,2:end-1) = ...
  (2*Skin_K(2:end-1,3:end  ,2:end-1).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(2:end-1,3:end  ,2:end-1)+ Skin_K(2:end-
1,2:end-1,2:end-1)).*CO4(2:end-1,2:end-1,2:end-1) ;
```

```
CO_0KM(2:end-1,2:end-1,2:end-1) = ...
  (2*Skin_K(2:end-1,2:end-1,1:end-2).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(2:end-1,2:end-1,1:end-2)+ Skin_K(2:end-
1,2:end-1,2:end-1)).*CO4(2:end-1,2:end-1,2:end-1) ;

CO_0KP(2:end-1,2:end-1,2:end-1) = ...
  (2*Skin_K(2:end-1,2:end-1,3:end  ).*Skin_K(2:end-1,2:end-1,2:end-1))./(Skin_K(2:end-1,2:end-1,2:end-1)+ Skin_K(2:end-
1,2:end-1,3:end  )).*CO4(2:end-1,2:end-1,2:end-1) ;

NEXT(:,:,:) = 0;  %all have 0 air faces except...
NEXT(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,2) = 1;  %front face
NEXT(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) = 5;  %Air, but may not be actually evaluated
end

%%
if Skin_Lesion== true %cheng herman 2011    simple experiment for interest
%defaults
Skin_K(:,:,:) = 0.445;              %conductivity
Skin_C(:,:,:) = 3300;               %specific heat
Skin_b(:,:,:) = 5000;               %perfusion
Skin_p(:,:,:) = 1200;               %skin density


Skin_b(50:60,50:60,2:9) = 25000;    %TEST TEST
Skin_K(50:60,50:60,2:9) = 0.558;
Skin_p(50:60,50:60,2:9) = 1030;
Skin_C(50:60,50:60,2:9) = 3852;     %specific heat
end
%-----------------------------------
%Make Coeeficients
A1 = Delta_T ./ Skin_C;
A2 = (Delta_T .* Skin_b) ./ (Skin_p .* Skin_C);
A3 = (Delta_T .* Skin_K) ./ (Skin_p .* Skin_C .* (Delta_Len^2));

%Initialise
Skin_Temp_C(:,:,:) = Body_T;
Skin_Temp_P = Skin_Temp_C;
%--------------
Plot_Interval = 5;                  %number of seconds between plot captures
pp=(Plot_Interval/Delta_T)+1;       %plot at time 0
ppp=0;
%--------------
%set up E field vector
%based on E = E_0*e^i(wt-bz) * e^-az
% main energy profile
E_Offset = 1;                       %Miss air cell at st front
if TE1_X == 0                       %Heat profile even CHANGED TO SKIP 1st cell 13/9/16 as 1st cell = air
    for zz=1:Cells_Z-2;
        E_Field(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,zz+E_Offset) = abs(EF_1 * exp(-Attenuation_C * (zz-1) *
Delta_Len));
    end

else                                %else its cos profile, or sine profile from one end

    X_Index = 0;
  for xx = App_Lower_X: App_Upper_X

    for zz=2:Cells_Z-1;
        E_Field(xx,App_Lower_Y:App_Upper_Y,zz+E_Offset) = ...
              abs((EF_1 * exp(-Attenuation_C * (zz-1) * Delta_Len)) * sin(TE1_X*pi*X_Index*Delta_Len/Applicator_X));
    end
    X_Index = X_Index+1;
  end

  %additional energy profile
    if TE2_X == 0
    else
        X_Index = 0;
        for xx = App_Lower_X: App_Upper_X
            for zz=2:Cells_Z-1;
                E_Field(xx,App_Lower_Y:App_Upper_Y,zz+E_Offset) = ...
                E_Field(xx,App_Lower_Y:App_Upper_Y,zz+E_Offset)+ abs(((EF_2 * exp(-Attenuation_C * (zz-1) * Delta_Len)) *
sin(TE2_X*pi*X_Index*Delta_Len/Applicator_X)));
            end
        X_Index = X_Index+1;
        end
    end

end

if Constant_E_Test == true %TEST WITH CONSTANT POWER
        E_Field(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1+E_Offset:Cells_Z-1) = EF_1;
end

Ptot=sum(sum(sum( E_Field .* E_Field * Skin_s / 2))) * Delta_Len^3 ;  %Total power calculated from cells

%keyboard %DEBUG

Power_Basic = (Skin_s/2) * (E_Field .* E_Field); %W/m3 for each cell
SAR_Basic = (Skin_s/2) * (E_Field .* E_Field) ./ Skin_p;  %W/kg
%
%-----------
%if we have input power profile override calculated SAR_Basic
if Use_Power_Profile == true
    if Delta_Len == 0.00025
       for kkk=2:Cells_Z-1
        Power_Profile_EM(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,kkk) = ...
           (Power_Loss_1Cyclep25(kkk-1)/sum( Power_Loss_1Cyclep25(1:Cells_Z-2))) *
Input_Power/(Applicator_X*Applicator_Y*Delta_Len);
        SAR_Basic = Power_Profile_EM ./ Skin_p;
        P_EM_Tot = sum(sum(sum(Power_Profile_EM))) * Delta_Len^3;
       end

    elseif Delta_Len == 0.0001
        for kkk=2:Cells_Z-1
```

```
        Power_Profile_EM(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,kkk) = ...
           (Power_Loss_1Cyclep10(kkk-1)/sum( Power_Loss_1Cyclep10(1:Cells_Z-2))) *
Input_Power/(Applicator_X*Applicator_Y*Delta_Len);
        SAR_Basic = Power_Profile_EM ./ Skin_p;
         P_EM_Tot = sum(sum(sum(Power_Profile_EM))) * Delta_Len^3;
        end

    else
        error('Input power profile enabled but not for required spacial resolution');
    end
end
%-----------
%keyboard                               %Debug
%-----------
Heat_Profile_S(1:Time_Steps)=0;         %heat profile for auto setting checked every timestep if auto seting used
%-------------------------------------------------------------------------
%Stability check, result must be less than 1. First one is worst case overall, 2nd is on cell by cell basis which should be
more realistic
Stability_Check1 = Delta_T * ( (12* max(max(max(Skin_K(2:end-1,2:end-1,2:end-1))))) + ( (Delta_Len^2) *
max(max(max(Skin_b(2:end-1,2:end-1,2:end-1)))) ) ) / ...
    (2 * min(min(min(Skin_p(2:end-1,2:end-1,2:end-1)))) * min(min(min(Skin_C(2:end-1,2:end-1,2:end-1)))) * (Delta_Len^2)) ;

Stability_Check2 = max(max(max((Delta_T * ((12.*Skin_K(2:end-1,2:end-1,2:end-1))+ ((Delta_Len^2).*(Skin_b(2:end-1,2:end-
1,2:end-1)))) ./ (2 *(Skin_p(2:end-1,2:end-1,2:end-1).*Skin_C(2:end-1,2:end-1,2:end-1).* (Delta_Len^2)))) )));
if Stability_Check1 >= 1;           %Uses worst case over whole model -not representitive
%    error('Failed Stability Check');
end
if Stability_Check2 >= 1;           %Evaluated on cell basis
    error('Failed Stability Check2');
end
%-------------------------------------------------------------------------
%Run simulation
Old_C_Sec = 9999; %make sure we sense a change in C_sec at start

%------------------Simulation loop----------------------------------------
for ii = 1:Time_Steps
pp=pp+1;                %Plot interval

C_Sec =( fix((ii-1)*Delta_T) +1);         %The second we are on

if Auto_Regulation == true
    if Temp_Depth<0   ; % Any depth is negative or specific depth
        if max(Skin_Temp_C(Centre_X,Centre_Y,:)) <= Auto_Temp_Lower; %Have we hit max at required depth
        Heat_Profile_S(ii) = 1;
        end
    else
        if Skin_Temp_C(Centre_X,Centre_Y,1+(Temp_Depth/(1000*Delta_Len))) <= Auto_Temp_Lower; %Have we hit max at required
depth
        Heat_Profile_S(ii) = 1;
        end
    end

    if Dial_A_Temp == true
        if Skin_Temp_C(Centre_X,Centre_Y,1+(Dial_A_Temp_Depth/(1000*Delta_Len))) >= Dial_A_Temp_Temp;
            if Dial_A_Temp_Stop ==false
            Dial_A_Temp_Time = (ii-1)*Delta_T;       %Save the time in seconds for how long it took
            disp(['TEMPERATURE REACHED AFTER: ' num2str(Dial_A_Temp_Time) ' Seconds']);
%keyboard
            Dial_A_Temp_Result = Skin_C;
            end
        Dial_A_Temp_Stop = true;
        end
    end ; %see if auto setting for reaching desired temp

    if Dial_A_Temp_Stop == true ; % Keep power off if so
         Heat_Profile_S(ii) = 0; %Keep Stopped
    end

    if Auto_Runtime < (ii-1)*Delta_T;  %if we have hit max on time turn it off anyway
    Heat_Profile_S(ii) = 0;
    end
    SAR = Heat_Profile_S(ii) * SAR_Basic;
end

if Auto_Regulation == false
SAR = Heat_Profile(C_Sec) * SAR_Basic;
Heat_Profile_S(ii) =  Heat_Profile(C_Sec);
end


if Numerical_Method == 0;%  Wang,Fujiware 1999 method
%Basic heat equation:
 Skin_Temp_C(2:end-1,2:end-1,2:end-1) = ...
   Skin_Temp_P(2:end-1,2:end-1,2:end-1) ...
      + (A1(2:end-1,2:end-1,2:end-1) .* SAR(2:end-1,2:end-1,2:end-1)) ...
      - (A2(2:end-1,2:end-1,2:end-1) .* ( Skin_Temp_P(2:end-1,2:end-1,2:end-1) - Blood_T))...
      + (A3(2:end-1,2:end-1,2:end-1) .* ...
                 (   Skin_Temp_P(3:end  ,2:end-1,2:end-1)...
                   + Skin_Temp_P(2:end-1,3:end  ,2:end-1)...
                   + Skin_Temp_P(2:end-1,2:end-1,3:end  )...
                   + Skin_Temp_P(1:end-2,2:end-1,2:end-1)...
                   + Skin_Temp_P(2:end-1,1:end-2,2:end-1)...
                   + Skin_Temp_P(2:end-1,2:end-1,1:end-2)...
                   - (6 * Skin_Temp_P(2:end-1,2:end-1,2:end-1))...
                 )...
         );

 %Boundary conditions
 % Z = 1 place is source of heat/output of applicator/air coolant

 Skin_Temp_C(:,:,1) = Applicator_Temp;  %Default

 Skin_Temp_C(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) = ...
```

```
     ((Skin_K(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) .*
Skin_Temp_P(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,2)) ./ (
Skin_K(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) + (H_a * Delta_Len)) )...
     +( ( Ambient_T * H_a * Delta_Len ) ./ ( Skin_K(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) + (H_a * Delta_Len))
);

 if Force_Aperture_Temp == true %Override convection if still running
     if Auto_Runtime >= (ii-1)*Delta_T
     Skin_Temp_C(App_Lower_X:App_Upper_X,App_Lower_Y:App_Upper_Y,1) = Force_T;
     end
 end;  %

  % Other planes are assumed body temperature
 Skin_Temp_C(:,  :,end) = Body_T;   % inner body Z end

 Skin_Temp_C(1,  :,:  ) = Body_T;   % X 1, end
 Skin_Temp_C(end,:,:  ) = Body_T;

 Skin_Temp_C(:,  1,:  ) = Body_T;   % Y 1 , end
 Skin_Temp_C(:,end,:  ) = Body_T;

end ;% Basic as per Wang 1999
%-------------------------------------------------------------
if Numerical_Method == 1  ;         % Bernardi et al 2003 IEEE Transaction vol 50, 3

%Basic heat equation:
 Skin_Temp_C(2:end-1,2:end-1,2:end-1) = (                          ...
                (  CO_0IM(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(1:end-2,2:end-1,2:end-1)  ...
                  +CO_0IP(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(3:end  ,2:end-1,2:end-1)  ...
                  +CO_0JM(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(2:end-1,1:end-2,2:end-1)  ...
                  +CO_0JP(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(2:end-1,3:end  ,2:end-1)  ...
                  +CO_0KM(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(2:end-1,2:end-1,1:end-2)  ...
                  +CO_0KP(2:end-1,2:end-1,2:end-1).* Skin_Temp_P(2:end-1,2:end-1,3:end  ) )...
                  ...
            + (Skin_Temp_P(2:end-1,2:end-1,2:end-1).*  ...
              ( 1 - CO_0IM(2:end-1,2:end-1,2:end-1) - CO_0IP(2:end-1,2:end-1,2:end-1)  ...
                  - CO_0JM(2:end-1,2:end-1,2:end-1) - CO_0JP(2:end-1,2:end-1,2:end-1)  ...
                  - CO_0KM(2:end-1,2:end-1,2:end-1) - CO_0KP(2:end-1,2:end-1,2:end-1)  ...
                     -(NEXT(2:end-1,2:end-1,2:end-1).*    CO1(2:end-1,2:end-1,2:end-1)) - CO2(2:end-1,2:end-1,2:end-1) )) ...
                ...
          + (NEXT(2:end-1,2:end-1,2:end-1).*CO1(2:end-1,2:end-1,2:end-1).*Ambient_T )...
            + (CO3(2:end-1,2:end-1,2:end-1).*SAR(2:end-1,2:end-1,2:end-1).*Skin_p(2:end-1,2:end-1,2:end-1))  ...
            + (CO2(2:end-1,2:end-1,2:end-1).* Blood_T)                                ...
                                       );

 %   %and cell 1
 %   Skin_Temp_C(2:end-1,2:end-1,1) = (                          ...
 %           + (Skin_Temp_P(2:end-1,2:end-1,1).*  ...
 %             ( 1  -(NEXT(2:end-1,2:end-1,1).*    CO1(2:end-1,2:end-1,1)) )) ...
 %            + (NEXT(2:end-1,2:end-1,1).*CO1(2:end-1,2:end-1,1).*Ambient_T )   );

 % Other planes are assumed body temperature
 %Skin_Temp_C(:,  :,end) = Body_T;   % inner body Z end

 Skin_Temp_C(1,  :,:  ) = Body_T;   % X 1, end
 Skin_Temp_C(end,:,:  ) = Body_T;

 Skin_Temp_C(:,  1,:  ) = Body_T;   % Y 1 , end
 Skin_Temp_C(:,end,:  ) = Body_T;


end

 Skin_Temp_P = Skin_Temp_C;

 Time_Line(ii,:)= squeeze(Skin_Temp_C(Centre_X,Centre_Y,1:end));

 Temp1 =  squeeze(Skin_Temp_C(Centre_X,Centre_Y,:));
 Temp2 = Dial_A_Temp_Tracking;
 Temp3= (Temp2 < Temp1);
 Dial_A_Temp_Tracking(Temp3)  =   Temp1(Temp3) ; %where new temp higher, update with max
 %
 if Plots_At_Runtime == true
        figure(3);
        hold off
        plot(squeeze(Skin_Temp_C(Centre_X,Centre_Y,1:end)));
        set(gca,'XTick',1:(.001/Delta_Len):Cells_Z)
        set(gca,'XTickLabel',0:14)
        v = axis;
        v(3)=30; v(4)=100;
        axis(v);
        grid on
        xlabel('Depth in mm')
        ylabel('Temperature in C')
        title('Heating profile at centre');
        drawnow;

    if pp*Delta_T >= Plot_Interval; %dont plot too often during runtim
        pp=0;
        figure(4);
        hold all
        plot(squeeze(Skin_Temp_C(Centre_X,Centre_Y,1:end)));
        set(gca,'XTick',1:(.001/Delta_Len):Cells_Z)
        set(gca,'XTickLabel',0:14)
        v = axis;
        v(3)=30; v(4)=100;
        axis(v);
        grid on
        xlabel('Depth in mm')
        ylabel('Temperature in C')
        title('Heating profile at centre');
        ppp=ppp+1;
        Leg{ppp} = ['Time - ',num2str( (ii-1)*Delta_T),'s'];
        legend(Leg);
```

```
        drawnow;
        hold off;
    end
 end

end
%--------------------------End of simulation loop----------------------
%plots
%-----------------------------------------------------
figure;  %Power profile
if Auto_Regulation == false
    bar(Heat_Profile,1);
    grid on;
    xlabel('Time in secs')
    ylabel('Proportion of heating power 0-1')
    title('Power heating profile');
    axis tight;
    v = axis;
    v(3)=0; v(4)=1.1;
    axis(v);

else %use this for regulation every time step
    bar(Heat_Profile_S,0.01);
    set(gca,'XTick',0:10/Delta_T:(Time_Steps))
    set(gca,'XTickLabel',0:10:Run_Time)
    grid on;
    xlabel('Time in secs')
    ylabel('Proportion of heating power 0-1')
    title('Power heating profile');
    axis tight;
    v = axis;
    v(3)=0; v(4)=1.1;
    axis(v);
end
%-------------------------------------------------------------------------
%%  Do Some 3D contour plots
LoCol=20;
HiCol=90;
C_Vec = [30,40,50,60,70];

figure;
title('Temperature contours at various depths');
subplot(3,2,1);
C_Depth = 0;
[X,Y] = meshgrid(0:1000*Delta_Len:1000*((Cells_X-1)*Delta_Len),0:1000*Delta_Len:1000*((Cells_Y-1)*Delta_Len) );
X=X';
Y=Y';
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));    %surface
xlabel('Width in mm');
ylabel('Height in mm');
zlabel('Temperature in C');
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);

subplot(3,2,2);
C_Depth = 1;
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));    %surface
xlabel('Width in mm');
ylabel('Height in mm');
zlabel('Temperature in C');
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);

subplot(3,2,3);
C_Depth = 2;
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);

subplot(3,2,4);
C_Depth = 3;
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);

subplot(3,2,5);
C_Depth = 4;
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);

subplot(3,2,6);
C_Depth = 5;
contourf(X,Y,squeeze(Skin_Temp_C(:,:,2+(round(C_Depth/(1000*Delta_Len))))));
caxis([LoCol HiCol])
C_Han=colorbar;
set(get(C_Han,'title'),'string','°C');
title(['Depth at ' num2str(C_Depth) 'mm ']);
drawnow;
%-------------------------------------------------------------
%% More plots
if 1==0  ; %  A group of plots we usually dont need
    figure;
    x=0:1000*Delta_Len:1000*((Cells_Z-1)*Delta_Len);
```

```
    plot(x,squeeze(Skin_Temp_C(Centre_X,Centre_Y,1:end)));
    set(gca,'XTick',0:1:1000*Delta_Len*Cells_Z)
    grid on
    xlabel('Depth in mm')
    ylabel('Temperature in C')
    title('Heating profile at centre');
    drawnow;
%-------------------------------------------------------------
    figure
    x=0:1000*Delta_Len:1000*((Cells_Y-1)*Delta_Len);
    plot(x,squeeze(Skin_Temp_C(Centre_X,1:Cells_Y,1+round(0.000/Delta_Len))) );
    grid on
    xlabel('Height in mm')
    ylabel('Temperature in C')
    title('Heating profile across height at 0mm depth');
    drawnow;
%-------------------------------------------------------------

    figure
    x=0:1000*Delta_Len:1000*((Cells_X-1)*Delta_Len);
    plot(x,squeeze(Skin_Temp_C(1:Cells_X,Centre_Y,round(1+0.002/Delta_Len))) );

    hold on
    x=[ (1000*(1)*Delta_Len), (1000*(App_Lower_X-1)*Delta_Len)];
    y = [Applicator_Temp,Applicator_Temp];
    %y = [37,37];
    plot(x,y,'r','LineWidth',2)
%% Plot heating at 2mm
    x=[ (1000*(App_Upper_X+1)*Delta_Len), (1000*(Cells_X)*Delta_Len)];
    y = [Applicator_Temp,Applicator_Temp];
    %y = [37,37];
    plot(x,y,'r','LineWidth',2)
    v=axis;
    v(3)=0;
    v(4)=(100);
    axis(v);
    grid on
    xlabel('Width in mm')
    ylabel('Temperature in C')
    title('Heating profile across width at 2mm depth');
    drawnow;
end
%% %Plot temp vs time at various depths
%-------------------------------------------------------------
% Plot temps down centre at various depths
clear Leg;
figure
%keyboard

yyy = linspace(0,Time_Steps*Delta_T,Time_Steps);
for dd=0:10  %Plot temp vs time at various depths

hold all
plot(yyy,Time_Line(:,1+(dd/(1000*Delta_Len))));
%set(gca,'XTick',0:5:(Time_Steps*Delta_T))
%set(gca,'XTickLabel',0:5:Run_Time)
grid minor;

Leg{dd+1} = ['Depth - ',num2str(dd),'mm'];
legend(Leg);
grid on
xlabel('Time in seconds')
ylabel('Temperature in °C')
end

if Use_Power_Profile == true
    Plot_P = P_EM_Tot;
    TEXT3 = ' Power profile from 1D EM solver ' ;
else
    Plot_P = Ptot;
    TEXT3 = [' Power profile analytic from e_r =' num2str(Skin_er) ' Cond = ' num2str(Skin_s) 'S/m']  ;
end

Total_Energy = sum(Delta_T*Plot_P .* Heat_Profile_S);

TEXT0 = ['f=' num2str(f,'%10.2e') 'Hz ' 'Power=' num2str(Plot_P) 'W ' 'Energy=' num2str(Total_Energy,'%5.1f' ) 'J  ' ...
                                      'Epi=' num2str(Epi_Depth_Cells*Delta_Len*1000) ' ' ...
                                      'Dermis=' num2str(Dermis_Depth_Cells*Delta_Len*1000) ' '...
                                      'Fat=' num2str(Fat_Depth_Cells*Delta_Len*1000) ' '...
                                      'Muscle=' num2str(Muscle_Depth_Cells*Delta_Len*1000) ' mm'];

if Force_Aperture_Temp == true
    TEXT1 = ['Surface Temperature forced to ', num2str(Force_T), ' °C ' ];
else TEXT1 = ['Wind Speed ' num2str(Wind_Speed) 'm/s at ',num2str(Ambient_T), ' °C ' ];
end


if Temp_Depth <0
    TEXT20 = 'any depth';
else
    TEXT20 = [num2str(Temp_Depth) 'mm depth'];
end


if Auto_Regulation == true
 TEXT2 = ['Power regulated by temperature at ' TEXT20 ' to ' num2str(Auto_Temp) '°C '];
else
 TEXT2 = ['Temperature determined by fixed power profile '] ;
end


 if Dial_A_Temp == true
 TEXT4 = ['Auto Power Off at ' num2str(Dial_A_Temp_Depth) 'mm/' num2str(Dial_A_Temp_Temp) '°C'];
 else
```

```
 TEXT4 = ' ';

 end

%{
TEXT3 = ['p= ' num2str(Skin_p_Const) 'Kg/m^3 ' 'SHeat C= ' num2str(Skin_C_Const) 'J/KgC '...
        'K=' num2str(Skin_K_Const) 'W/mC ' 'Prf= ' num2str(Skin_b_Const) 'W/m^3C ' 'h_a=' num2str(H_a)];
%}


TEXT00 = 'Heating profile at centre over time';
title({ [TEXT00] ; [TEXT0] ; [[TEXT1] [TEXT3]]; [[TEXT2] [TEXT4]] });
drawnow;
hold off;
%keyboard %DEBUG

%{
figure;
%plot max temps
x=find(Dial_A_Temp_Tracking<50,1,'first');
x*Delta_Len
x=0:1000*Delta_Len:  1000*((Cells_Z-1)*Delta_Len);
plot(x,Dial_A_Temp_Tracking)
grid on;
grid minor;
title('Maximum temperatures reached vs Depth');
xlabel('Depth in mm');
ylabel('Maximum Temperature in C');
drawnow;
%}

%% Create Cem43 and plot vs Max temps
      [x,y] = size(Time_Line);     % x = time, y = distance
      yy = linspace(0,((y-1)*Delta_Len*1000),y);
      Cem43(1:y)= 0;  %make matrix
      for jj=1:y; %depth
          for ii=1:x ; %steps
            if Time_Line(ii,jj) >= 43
            R = 0.5;
            else R = 0.25;
            end
            Cem43(jj) = Cem43(jj) + ( (1/60)*Delta_T * (R^(43-(Time_Line(ii,jj))) ));
              if(Time_Line(ii,jj)) >=55 Cem43(jj)=1000; end
          end
      end

    S_Start = Delta_Len*1000;
    F_Start = 1000*(Fat_Start_Cell-1)*Delta_Len;
    M_Start = 1000*(Muscle_Start_Cell-1)*Delta_Len;
    P_End = 1000*(Muscle_End_Cell)*Delta_Len;

      figure
      [AX,H1,H2] = plotyy(yy,squeeze(Dial_A_Temp_Tracking),yy,Cem43);
       set(AX(1),'ylim',[0 100],'ytick',[0 10 20 30 40 50 60 70 80 90 100]);
       set(AX(2),'ylim',[0 200],'ytick',[0 20 40 60 80 100 120 140 160 180 200]);
       set(AX(2),'xtick',[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]);
       set(get(AX(2),'Ylabel'),'String','Cem43 in minutes');
       set(get(AX(1),'Ylabel'),'String','Maximum temperature in °C');
       text(9,91,['Frequency = ' num2str(f,'%5.2e') ' Hz']);
      grid on
      grid minor
      xlabel('Depth from skin surface in mm');
      title('Maximum temperatures reached and Cem43 vs depth (Cem43 set to 1000 if Temp >55)');
       set(AX(2),'Ycolor','k')

      Pat=patch( [0 (Delta_Len*1000) (Delta_Len*1000) 0], [0 0 100 100],'r' ); %air/skin boundary
       set(Pat,'FaceAlpha',0.25);

      if (Epi_Depth_Cells+Dermis_Depth_Cells)>0
       text(S_Start,96,' Skin');
      end

      if Fat_Depth_Cells>0
            Pat=patch( [F_Start M_Start M_Start F_Start], [0 0 100 100],'y' );
            set(Pat,'FaceAlpha',0.25);
            text(F_Start,96,' Fat');
      end

      if Muscle_Depth_Cells>0
            text(M_Start,96,' Muscle');
            Pat=patch( [M_Start P_End P_End M_Start], [0 0 100 100],'g' );
            set(Pat,'FaceAlpha',0.25);
      end

      annotation('textarrow',[0.25 .135],[0.85 0.83],'String','Air boundary')
%%   %Calculate last known good from Cem43 and ablation from max
  OK_Damage_Depth = Cells_Z * Delta_Len;   %default damage at end
  Found_Damage= false;
  for pp = Cells_Z:-1:1  %Find damage depth
    if Found_Damage == false
      CEM_Thresh = 80;
            if Fat_Depth_Cells >0 %are we in fat?
                if pp >= Fat_Start_Cell
                    if pp <= Fat_End_Cell
                    CEM_Thresh = 40;
                    end
                end
            end

        if Cem43(pp)>= CEM_Thresh      %Over limit
            if pp<Cells_Z
               pp = pp+1;               %get next good
            end
                OK_Damage_Depth = (pp-1)*Delta_Len*1000
```

```
                Found_Damage = true;      %Stop
          end
      end
  end

  Found_Ablation = false;
  Ablation_Depth = 0;
  for pp = Cells_Z:-1:1  %Find ablation depth
      if Found_Ablation == false
         if Dial_A_Temp_Tracking(pp)>=60
          Ablation_Depth = (pp-1)*Delta_Len*1000
          Found_Ablation = true;
         end
      end
  end

end
```

Appendix 2 3D EM Modeller

Multiapp8.m

Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has
not been verified as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
%Run Appsim
% version to test 1 run of appsim 4, %23-6-2013 Version 6, simplify multi call, use appsim6 , %20/8/2013 added PMC,
%23/8/2013 added ferrite slabs
%-------------------------------------------------------------------------------------
close all
clear all

Debug = false;                              % not debugging sim
PlotNumber = 0;                             % for getting multiple plots to compare 0,10,20 etc
MovieName = 'C:\Users\Paul\Desktop\Applicator-Waves0.avi'; %name for animation
Delta_Len = 0.0005;                         % metres
Inject_F =  14.5e9;                         % frequency
Scale_Time = 1;                             % quick method to scale time and timesteps to see effect
Capture_Start_Cycles = 1;                   % dump this many cycles before capturing
%Block_re = 2.5;                             % Block relative e
Block_Thickness = 0.002;                    % metres
Skin_Gap = 0;                               % Gap between flange and skin
DontPlot = false;                           % set true to inhibit plots
Animate = true;                             % make wave movie
Make_Movie = true;                          % Write movie to disk
%Side_Blocks_PTFE = false;                        % different e_r blocks on sides of cavity

Skin = true;                                % model human skin yes/no
Skin_nr =  26.88        ;                    % permittivity dryskin e=26.88, cond = 13.27. lossT = 0.61199, pen depth =
0.0021617
Skin_s =   13.27;                              % conductivity

Pec_Method_1 = false;                       % method1 =specific E fields cleared at run time, else used physical and
calculated map


Strip_Height = 0.003;
Strip_Offset = 0.002;
Strip_Gap    = 0.001;

% ** ONLY 1 of Following allowed to be true!!***
Side_Blocks_META = false;
Simple_Side_Blocks_PMC = false;
Ferrite_Side_Blocks_PMC = false;
Side_Blocks_PTFE = false;                        % different e_r blocks on sides of cavity

%Ferrite parameters
B_0 = 0.45; %0.3; %0.4; %0.6;
E_r = 10;
U_r = 2; %5 ;2;                              % relative permeability that makes B inside the ferrite material from H_0 ( B =
u_r*u_0 * H_0)
A_Loss = 0.00; %0; %0.05;                    % Ferrite loss factor

Cycles = 10;                                % number of full cycles to do
Scale_Time = 1;
Capture_Start_Cycles = 5;                  % dump this many cycles before capturing
%Block_Min = 0.0035;
Block_Steps= 1;
DontPlot = true;

Block_re = 2.1;

DrawPlots = true;

if DrawPlots==true
% set up plot formats
 scrsz = get(0,'ScreenSize');
 Plot_Vec = [1,1; 1,2 ; 2,2; 2,2; 3,2; 3,2];
 Plot_Scale = [2,1.5; 2,2; 2,2; 2,2 ; 3,2; 3,2];
 Plot_Size= [ 200 80 (scrsz(3)*Plot_Scale(Block_Steps,2))/(1.8*2) (scrsz(4)*Plot_Scale(Block_Steps,1))/(1.1*3)];

 hh=figure;
 set(hh,'OuterPosition',[Plot_Size],'Name','Intensity for various block thicknesses');
 Az=35,El=15;
 %MAmp=0.05;
 Skin_Off=2;
 Str1 = ['E^2 Intensity '       ' At Skin Offset(Cells)= ' num2str(Skin_Off) ' Cycles ' num2str(Cycles) ' Ignore 1st '
num2str(Capture_Start_Cycles)];
 annotation('textbox','Position',[0 0.83 0.17 0.16], 'LineStyle', 'none',
'FitBoxToText','on','fontweight','b','String',char(Str1));
end
```

```
ii=0;
for PlotNumber=1:Block_Steps
 ii=ii+1;

Inject_F = 14.5e9 + ( (ii-1) * 0.5e9);
%Block_Thickness = Block_Min+(0.0005 * (ii-1));                    % steps of 0.5mm
%Block_re = Block_re + (0.5 * (ii-1)); %******
if Side_Blocks_PTFE == false; Block_Thickness = 0; end;
if Skin == false; Skin_nr = 0 ; Skin_s = 0; end;

[Vrecord Irecord E_X E_Y E_Z H_X H_Y H_Z E2 ZZ Animation TimeSteps Delta_T Cavity2_Upper_Boundary_Z] =
Appsim8(Debug,PlotNumber,MovieName,...
                                                                    Delta_Len, Inject_F, Cycles, Scale_Time,
Capture_Start_Cycles,B_0,U_r,E_r,A_Loss, ...
                                                                    Block_re, Block_Thickness, Strip_Height,
Strip_Offset, Strip_Gap, Skin_Gap, ...
                                                                    DontPlot, Animate, Make_Movie, Side_Blocks_PTFE,
Side_Blocks_META,Simple_Side_Blocks_PMC, Ferrite_Side_Blocks_PMC, Skin,Skin_nr,Skin_s,Pec_Method_1);

EI20(:,:,:,PlotNumber) = E2(:,:,Cavity2_Upper_Boundary_Z:Cavity2_Upper_Boundary_Z+9);
ZZ20(PlotNumber) = ZZ;
S_11 = 20*log10(abs((ZZ-50)/(ZZ+50)));
S11(PlotNumber) = S_11
Freq(PlotNumber) = Inject_F

if DrawPlots==true
 figure(hh);
 subplot( Plot_Vec(Block_Steps,1),Plot_Vec(Block_Steps,2),ii);
 contour3(squeeze(EI20(:,:,Skin_Off,ii))',60);
 view(Az,El);
 drawnow;
 test=zlim;
 title(['F=' num2str(Inject_F,'%10.2e') ' Block-t,e= ' num2str(1000*Block_Thickness) 'mm' ',' num2str(Block_re) ' Z= '
num2str(ZZ20(ii),'%10.1e') ' S11= ' num2str(S_11,'%10.1f')  ' Skin-nr,s= ' num2str(Skin_nr,'%10.1f') ','
num2str(Skin_s,'%10.1f')   ]);

 figure
 plot(Vrecord);
 figure
 plot(Irecord);
end

hhh=figure;
 plot(Irecord);

end
```

Appsim8.m

Disclaimer: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the simulation of thermal and/or electromagnetic characteristics only and has
not been verified as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
%simulator for skin applicator
%P horwitz 6/4/2013
%
%Simulation based on Inan and Marshall, Numerical Electromagnetics, The FDTD method , Cambridge University Press ISBN 978-0-
52119069-5
%
%Plan view
%    Cavity1       Cavity2
%                                      24mm
%   ----------------------------     18mm  I       X
%   I-----------I                     I    I       ^
%   I                                 I    I       I
%   I   O                             I    I       -------> Z
%   I                                 I    I
%   I-----------I                     I    I
%   ----------------------------      I
%   <----------->                 19mm
%               <------------->   23mm
%   <--------------------------->  42mm
%
%Side view                            18mm
%   ----------------------------    8mm    I
%   I-----------I                    I     I      Y
%   I   i                            I     I      ^
%   I-----------I                    I     I      --------> Z
%   ----------------------------     I
%   <---> 9mm (MonoPole_Z)
%
%front view
%   ----------------------                       Y
%   I  ----------------  I                       ^
%   I  I       i      I  I                       -------->X
%   I  ----------------  I
%   ----------------------
%      <--------------> 18mm
%   <---------------------->  24mm
%
% i = monopole, diameter = 1mm , length 2mm (Monopole_Y_Len)
%
%From Italian national research council website IFAC http://niremf.ifac.cnr.it/tissprop/htmlclie/htmlclie.htm#atsftag
%skin parameters at 15ghz
%Dry Skin
%cond S/m = 13.8, real? rel perm 26.4, loss tangent 0.63, wavelength 3.7mm, penetration 2mm
%
%Wet skin
%cond S/m = 14.6, real? rel perm 28.1, loss tang 0.62, wavelength 3.6, penetration 2mm
%
% Rev 2 various changes 4 face antenna
%
% Rev 5 10-6-2013 major 'BUG' blocks on side were only for E in Y
% direction, changed to x,y,z
%11-6-2013 added META blocks
%23-6-2013 return V and I vectors and Delta_T
%3-7-2013 50ohm source as per luebbers 1999
%20/8/2013 added PMC on sides of outer cavity
%23/8/2013 added Ferrite slabs on side of cavity with H0 in X direction -
%to simulate PMC
%--------------------------------------------------------------------------------
 function [Vrecord Irecord E_X E_Y E_Z H_X H_Y H_Z E2 ZZ Animation TimeSteps Delta_T Cavity2_Upper_Boundary_Z] =
Appsim8(Debug,PlotNumber,MovieName,...
                                                                    Delta_Len, Inject_F, Cycles, Scale_Time,
Capture_Start_Cycles,B_0,U_r,E_r,A_Loss, ...
                                                                    Block_re, Block_Thickness, Strip_Height,
Strip_Offset,Strip_Gap, Skin_Gap, ...
                                                                    DontPlot, Animate, Make_Movie, Side_Blocks_PTFE,
Side_Blocks_META, Simple_Side_Blocks_PMC, Ferrite_Side_Blocks_PMC, Skin,Skin_nr,Skin_s,Pec_Method_1);
% and H matrices

%% Constants

%clear all;
%close all;
%ticID=tic;
[ST,I]=dbstack('-completenames');
OurName=ST.name;

%Delta_Len = 0.0005;                    %Lambda_Free/10
%Inject_F =  14.5e9;                    %14.5e9; %13.2e9;
%Cycles = 20;                           % number of full cycles to do
%Scale_Time = 1; % 1;                   % quick method to scale time and timesteps to see effect
%Block_re = 2;                          % Block relative e
%Block_Thickness = 0.004;               %metres
%Skin_Gap = 0                           %Gap between flange and skin
```

```
%DontPlot                                  %set true to inhibit plots
%Animate = true;                           %make wave movie
%Side_Blocks = true;                       %different e_0 blocks on sides of cavity
%Skin = true;                              %model human skin yes/no
%DrySkin = true;                           % else wetskin re ifac @14.5Ghz

Delta_T =  0.50e-12 * Scale_Time;       % Time Step
TimeSteps = round(Cycles * (1/(Inject_F * Delta_T)))
%Stop_Cycles = 5;
%Stop_Time_Steps = round(Stop_Cycles * (1/(Inject_F * Delta_T)))

Dim_1_Antenna = false;                  %if true = single vec antenna, if false then 4 x e vecs and solid pec

PlotNumber = PlotNumber * 10;           % max 10 plots per run
%Interference_plane = false;

Draw_Sim1 = true;
Draw_Sim2 = false;
Draw_Sim3 = false;                       %E field vector plot and dipole current

Ramp_Cycles = 3;                        % No of cycles to ramp up input
% Capture_Start_Cycles = 15;            % This number of cycles before capturing E2

PML = true;
PML_Graded = true;                      % False just sets absorption to testss value
Simple_Source = false;                  % simple = soft addition of E_Y at centre, not simple - soft source and zeroed E_Y
along dipole
%
Draw_Perm = true;                       % Plot ferrite params
%-------------------------------------------------------------------------------------
%Constants
e_0 = 8.854e-12;
mu_0 = 4*pi*1e-7;
c= 1/sqrt(e_0*mu_0);

%if DrySkin==true;                       % Skin permittivity and cond for 14.5ghz
%Skin_e = 26.88 * e_0;                   % permittivity dryskin e=26.88, cond = 13.27. lossT = 0.61199, pen depth = 0.0021617
%Skin_s = 13.27;                         % conductivity
%else                                    %else its wetskin
%Skin_e = 28.621 * e_0;                   %
%Skin_s = 14.082;                         % conductivity
%end
Skin_e = Skin_nr * e_0;                  % not relative
%-------------------------------------------------------------------------------------
%Source
%For injecting sinusoid
%Num_Cycles = Inject_F * Delta_T * TimeSteps;
Inject_W = 2 * pi * Inject_F;           %rads/sec
Lambda_Free = c/Inject_F                %wavelength in free space

%-------------------------------------------------------------------------------------
%PML
PML_T = 4;                              %14 Thickness in cells
if Debug==true PML_T = 10; end;
PML_U = mu_0;
PML_E = e_0;
E_0 = e_0;
testss = 2.0;                           %absorbtion coeff not graded - test
PML_Eta = sqrt(PML_U/PML_E);            % for graded , See Inan and Marshall pg 10
PML_m = 3; %3;  2 slightly better with free space
%-------------------------------------------------------------------------------------
%Free space
Fe_0 = e_0;
Fmu_0 = mu_0;
%-------------------------------------------------------------------------------------
%ferrite
f_e = E_r * 8.854e-12;                  % Permittivity of 'true' ferrite


%B_0 = 1;                               % saturating field in tesla
%U_r = 2;                               % relative permeability that makes B inside the ferrite material from H_0 ( B =
u_r*u_0 * H_0)
%E_r = 1 ;                              % relative permittivity of 'true' ferrite
%A_Loss = 0.01;                         % Ferrite loss factor

%% Simulation space physical and impedance calc init
%Simulation space Small for debug
if Debug == true
Length_X = 0.05;              %width in metres
Length_Y = 0.05;              %height
Length_Z = 0.07;              %Length
else
%Simulation space
Length_X = 0.040;             %width in metres
Length_Y = 0.035;             %height
Length_Z = 0.07;              %length lots of space for far field in front
end


Yee_N_X = round(Length_X/Delta_Len);         % yee cells in X direction
Yee_N_Y = round(Length_Y/Delta_Len);
Yee_N_Z = round(Length_Z/Delta_Len);

CellNumX = Yee_N_X
CellNumY = Yee_N_Y
CellNumZ = Yee_N_Z

Delta_X = Delta_Len;
Delta_Y = Delta_Len;
Delta_Z = Delta_Len;

Centre_X = round(Yee_N_X/2);
Centre_Y = round(Yee_N_Y/2);
Centre_Z = round(Yee_N_Z/2);               %Not used unless debugging
```

```
Appl_Z = 14 ; % no of cells where back of applicator starts away from Z=1
%-------------------------------------------------------------------------------------------

Flange_Width = 0.003;      %Thickness of flange looking from front
Flange_Cells = round(Flange_Width/Delta_Len);

%Cavity defs
Cavity1_Height = 0.008;  %Y
Cavity1_Width = 0.018;   %X
Cavity1_Length = 0.019;  %Z

Cavity2_Height = 0.018;  %Y
Cavity2_Width = 0.024;   %X
Cavity2_Length = 0.023;  %Z

%Cavity indices
Cavity1_Lower_Boundary_Y = Centre_Y - ((Cavity1_Height/Delta_Len)/2);
Cavity1_Upper_Boundary_Y = Cavity1_Lower_Boundary_Y + (Cavity1_Height/Delta_Len);

Cavity1_Lower_Boundary_X = Centre_X - ((Cavity1_Width/Delta_Len)/2);
Cavity1_Upper_Boundary_X = Cavity1_Lower_Boundary_X + (Cavity1_Width/Delta_Len);

Cavity1_Lower_Boundary_Z = Appl_Z+1;
Cavity1_Upper_Boundary_Z = Cavity1_Lower_Boundary_Z + (Cavity1_Length/Delta_Len);


Cavity2_Lower_Boundary_Y = Centre_Y - ((Cavity2_Height/Delta_Len)/2);
Cavity2_Upper_Boundary_Y = Cavity2_Lower_Boundary_Y + (Cavity2_Height/Delta_Len);

Cavity2_Lower_Boundary_X = Centre_X - ((Cavity2_Width/Delta_Len)/2);
Cavity2_Upper_Boundary_X = Cavity2_Lower_Boundary_X + (Cavity2_Width/Delta_Len);

Cavity2_Lower_Boundary_Z = Cavity1_Upper_Boundary_Z+1;
Cavity2_Upper_Boundary_Z = Cavity2_Lower_Boundary_Z + (Cavity2_Length/Delta_Len)

Front_Lower_Boundary_X = Cavity2_Lower_Boundary_X-Flange_Cells;
Front_Upper_Boundary_X = Cavity2_Upper_Boundary_X+Flange_Cells;
Front_Lower_Boundary_Y = Cavity2_Lower_Boundary_Y-Flange_Cells;
Front_Upper_Boundary_Y = Cavity2_Upper_Boundary_Y+Flange_Cells;

%-------------------------------------------------------------------------------------------
%% Monopole position - orientated parallel to Y axiz
%-------------------------------------------------------------------------------------------
Monopole_Z = 0.0062;       %distance from rear

if Debug==true
Monopole_Y_Len = Lambda_Free* 0.5/2     %length
else
Monopole_Y_Len = 0.006; %0.0025;    %length
end

Monopole_X_Coord = Centre_X

if Debug==true
Monopole_Z_Coord = Centre_Z             %override for debug
else
Monopole_Z_Coord = Cavity1_Lower_Boundary_Z + round(Monopole_Z/Delta_Z)
end

Monopole_Y_Coord = Cavity1_Lower_Boundary_Y+1
Monopole_Y_Cells = round(Monopole_Y_Len/Delta_Y)

%surround monopole with dielectric
Dielectric_Rad = 0.002; %dielectric radius
Dielectric_Antenna = true; %surrounded by dielectric
Dielectric_e = 2.1;


%-------------------------------------------------------------------------------------------
%
Dipole_Len = (2* Monopole_Y_Cells*Delta_Len) + Delta_Len;   %+1 for the gap
Dipole_Params(Inject_F,Dipole_Len,Delta_Len)/2 ;              % get monopole = dipole/2

%-------------------------------------------------------------------------------------------
%% PEC objects   Make physical map of pec cells

App_Pec= false(CellNumX,CellNumY,CellNumZ);
ZeroEX=false(CellNumX,CellNumY,CellNumZ);
ZeroEY=false(CellNumX,CellNumY,CellNumZ);
ZeroEZ=false(CellNumX,CellNumY,CellNumZ);

%Cavity 1
if Debug == false  % do normal cavity pec method if not debugging
%Back
App_Pec(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1  ,Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,
Cavity1_Lower_Boundary_Z-1)=true;


%top
App_Pec(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z)=true;
%bot
App_Pec(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z)=true;
%left
App_Pec(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z)=true;
%right
App_Pec(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z)=true;


%rear wall of cavity2
```

```
%top
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,   Cavity1_Upper_Boundary_Y+1:Cavity2_Upper_Boundary_Y+1,
Cavity1_Upper_Boundary_Z)=true;
%bot
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,   Cavity2_Lower_Boundary_Y-1:Cavity1_Lower_Boundary_Y-1,
Cavity1_Upper_Boundary_Z)=true;
%left
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity1_Lower_Boundary_X-1,   Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,
Cavity1_Upper_Boundary_Z)=true;
%right
App_Pec(Cavity1_Upper_Boundary_X+1:Cavity2_Upper_Boundary_X+1,   Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,
Cavity1_Upper_Boundary_Z)=true;


%Cavity 2
%top
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,  Cavity2_Lower_Boundary_Z-
1:Cavity2_Upper_Boundary_Z)=true;
%bot
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,  Cavity2_Lower_Boundary_Z-
1:Cavity2_Upper_Boundary_Z)=true;
%left
App_Pec(Cavity2_Lower_Boundary_X-1,   Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,  Cavity2_Lower_Boundary_Z-
1:Cavity2_Upper_Boundary_Z)=true;
%right
App_Pec(Cavity2_Upper_Boundary_X+1,   Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,  Cavity2_Lower_Boundary_Z-
1:Cavity2_Upper_Boundary_Z)=true;


%Flange at front
%Bottom
App_Pec(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Cavity2_Lower_Boundary_Y-1,
Cavity2_Upper_Boundary_Z)=true;
%Top
App_Pec(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+1:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=true;
%Left
App_Pec(Front_Lower_Boundary_X:Cavity2_Lower_Boundary_X-1, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=true;
%Right
App_Pec(Cavity2_Upper_Boundary_X+1:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=true;
end

if Debug == true  %make pec model of ground plane
App_Pec(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-
1:Cavity2_Upper_Boundary_Z+1)=true;
end

if Dim_1_Antenna == false
 App_Pec(Monopole_X_Coord, Monopole_Y_Coord:Monopole_Y_Coord+Monopole_Y_Cells-1, Monopole_Z_Coord) = true;
end

for xx=1:CellNumX  %Make map of E fields to zero to simulate PEC
   for yy=1:CellNumY
       for zz=1:CellNumZ

         if App_Pec(xx,yy,zz)==true
           ZeroEX(xx,yy,zz)=true;
           ZeroEY(xx,yy,zz)=true;
           ZeroEZ(xx,yy,zz)=true;

           ZeroEY(xx+1,yy,zz)=true;
           ZeroEZ(xx+1,yy,zz)=true;

           ZeroEX(xx,yy+1,zz)=true;
           ZeroEZ(xx,yy+1,zz)=true;

           ZeroEZ(xx+1,yy+1,zz)=true;

           ZeroEX(xx,yy,zz+1)=true;
           ZeroEY(xx,yy,zz+1)=true;

           ZeroEY(xx+1,yy,zz+1)=true;

           ZeroEX(xx,yy+1,zz+1)=true;
         end
       end
   end
end

%keyboard
%% Simple blocks on each side of cavity2
Block_Cells = round(Block_Thickness/Delta_X);
%% Meta blocks on each side
Strip_Height_Cells = round(Strip_Height / Delta_Len);
Strip_Offset_Cells = round(Strip_Offset / Delta_Len);
Strip_Gap_Cells = round(Strip_Gap / Delta_Len);

META_Offset = 2; %2 cells away from wall

%% Skin variables
%patch of skin at end
Skin_Thickness = 0.005;     %thickness of skin mm
Skin_Thickness_Cells = round(Skin_Thickness/Delta_Len);

Skin_Overlap = Flange_Width; %0.002;       %how much skin patch overlaps cavity mm
Skin_Overlap_Cells = round(Skin_Overlap/Delta_Len);

Skin_Lower_Boundary_X = Cavity2_Lower_Boundary_X-Skin_Overlap_Cells;
Skin_Upper_Boundary_X = Cavity2_Upper_Boundary_X+Skin_Overlap_Cells;
Skin_Lower_Boundary_Y = Cavity2_Lower_Boundary_Y-Skin_Overlap_Cells;
Skin_Upper_Boundary_Y = Cavity2_Upper_Boundary_Y+Skin_Overlap_Cells;
Skin_Lower_Boundary_Z = Cavity2_Upper_Boundary_Z+1+Skin_Gap;
```

```
Skin_Upper_Boundary_Z = Skin_Lower_Boundary_Z+Skin_Thickness_Cells-1;

SC1 = ( (2* Skin_e) - (Skin_s*Delta_T))/( (2* Skin_e) + (Skin_s*Delta_T)); %coefficients for skin update equ
SC2 = (2 * Delta_T)/( (2* Skin_e) + (Skin_s*Delta_T));
%% various impedance calc vars
%ZZSUM = 0;                  % rolling sum of impedance to work out input impedance
%ZZAVE = 0;                  % Allocate storage of the number
I_s = 0;
E_s = 0;
I_si = 0;                    %Injected current
ZZ=0;
%% CFL limit
CFL = Delta_T * c * sqrt( (1/Delta_X)^2 + (1/Delta_Y)^2 + (1/Delta_Z)^2 ) %
%-------------------------------------------------------------------------------------------------
%% Initialise Ferrite and Characteristics and draw Relative perm vs B_0 and I_freq
%the following adapted from  Inan and marshall pg 281 onwards

Y_e = 1.76e11;       % gyromagnetic ratio for electron, in radians/sec-Tesla
F_0 =  Y_e * B_0/(2 * pi);
F_M =  Y_e * B_0 * U_r/(2 * pi);
W_0 = -2*pi*F_0;
W_M = -2*pi*F_M;

MS= B_0 * U_r / mu_0
MS_4PI = MS * 1e-3

U11 = 1 +  (W_M* (W_0 + 1i*Inject_W*A_Loss) / ( (W_0+1i*Inject_W*A_Loss)^2 - Inject_W^2) )       %test 1i replacement for i
U12 = (1i*Inject_W*W_M) / ( (W_0+(1i*Inject_W*A_Loss))^2 - (Inject_W^2) )

% plot U11 and U12 vs B0 at current inject freq
B0V = 0.1:0.01:1.1;
F0V =  Y_e * B0V/(2 * pi);
FMV =  Y_e * B0V * U_r/(2 * pi);
W0V = -2*pi*F0V;
WMV = -2*pi*FMV;
U11V = 1 + (WMV.*(W0V + 1i*Inject_W*A_Loss)./((W0V+ (1i*Inject_W*A_Loss)).^2 - Inject_W^2));
U12V = (1i*Inject_W.*WMV) ./ ( (W0V+(1i*Inject_W*A_Loss)).^2 - (Inject_W^2) );

if Draw_Perm == true;
figure(20);
plot(B0V,real(U11V),'-r',B0V,imag(U11V),'--g', B0V,real(U12V),'-b', B0V,imag(U12V),'--k');
xlabel(['Field strength in Tesla for Inject Freq = ' num2str(Inject_F,'%10.2e') 'Hz , A-Loss = ' num2str(A_Loss,'%10.2f') ' ,
U_r = ' num2str(U_r,'%10.2f')]); ylabel('Relative Permeability');
hleg1=legend('Real-U11','Imag-U11','Real-U12','Imag-U12');
grid on;
drawnow;

%Plot U11 and U12 vs inject freq at selected B_0
IFreq = 1:0.1:40;     %ghz
IW = IFreq * 2 * pi * 1e9;
U11V = 1 +  (W_M * (W_0 + (1i * IW * A_Loss) ) ./ ( (W_0 + (1i * IW * A_Loss)).^2 - IW.^2) );       %test 1i replacement for
i
U12V = (1i * IW * W_M) ./ ( (W_0 + (1i * IW * A_Loss)).^2 - (IW.^2) );
figure(21);
plot(IFreq,real(U11V),'-r',IFreq,imag(U11V),'--g', IFreq,real(U12V),'-b', IFreq,imag(U12V),'--k');
xlabel(['Inject frequency in GHz for field strength B0 = ' num2str(B_0,'%10.2f') 'T , A-Loss = ' num2str(A_Loss,'%10.2f') ' ,
U_r = ' num2str(U_r,'%10.2f')]); ylabel('Relative Permeability');
hleg1=legend('Real-U11','Imag-U11','Real-U12','Imag-U12');
grid on;
drawnow;
end;


Ferrite_Loc(1:Yee_N_X,1:Yee_N_Y,1:Yee_N_Z)=false;
Ferrite_Loc(Cavity1_Lower_Boundary_X+1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = true;
Ferrite_Loc(Cavity1_Upper_Boundary_X-1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = true;
Ferrite_Loc(Cavity2_Lower_Boundary_X+1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
Ferrite_Loc(Cavity2_Upper_Boundary_X-1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;

%% Animation

AnimationSteps = 1;  %no of steps to do before doing an image capture
Astep = 1;
Animation = [];      %might need to pass back an array
if Animate == true
    if Make_Movie == true
      writerObj = VideoWriter(MovieName);
      writerObj.FrameRate = 60;
      open(writerObj);
      %aviObj = avifile(MovieName,'compression','None');
      Fig9=figure(9+PlotNumber);
      set(gca,'nextplot','replacechildren');
      set(gcf,'Renderer','zbuffer');
      %axis([1 CellNumZ 1 CellNumX -0.05 0.05 -0.05 0.05]);
      %pbaspect([CellNumZ/CellNumX 1 1 ]);
      %view(30,40);
      %set(gcf,'Visible', 'off');
      set(0,'CurrentFigure',9+PlotNumber);
      drawnow;
    end
end
%% Initialse sim Matrices and other vars

%Permittivity matrix

R_e_X = (1/e_0) * ones(Yee_N_X,Yee_N_Y,Yee_N_Z);
R_e_Y = (1/e_0) * ones(Yee_N_X,Yee_N_Y,Yee_N_Z);
R_e_Z = (1/e_0) * ones(Yee_N_X,Yee_N_Y,Yee_N_Z);

% E and D fields are located on Yee cell boundaries
```

```
E2 = zeros(CellNumX,CellNumY,CellNumZ);    % used to make aggregate intensity

E_X = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (1/2, 0 ,  0 )
E_Y = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (0  ,1/2 , 0 )
E_Z = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (0  , 0  ,1/2)

E_X_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (1/2, 0 ,  0 )
E_Y_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (0  ,1/2 , 0 )
E_Z_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (0  , 0  ,1/2)

D_X = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
D_Y = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
D_Z = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

D_X_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
D_Y_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
D_Z_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

%H and B fields are actually advanced by 1/2 step
H_X = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   % its really offset by (0   ,1/2,1/2)
H_Y = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
H_Z = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

B_X = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);   %current B
B_Y = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
B_Z = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

B_X_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z); %Previous B = n-1/2
B_Y_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
B_Z_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

B_X_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z); %Previous Previous B = n-3/2
B_Y_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
B_Z_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

H_X_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z); %Previous H = n-1/2
H_Y_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
H_Z_P = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

H_X_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z); %Previous Previous H = n-3/2
H_Y_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
H_Z_PP = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);

K_X = ones(Yee_N_X,Yee_N_Y,Yee_N_Z);    %PML Matrices
K_Y = ones(Yee_N_X,Yee_N_Y,Yee_N_Z);
K_Z = ones(Yee_N_X,Yee_N_Y,Yee_N_Z);
Om_X = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);  % Magnetic 'conductivity'
Om_Y = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
Om_Z = zeros(Yee_N_X,Yee_N_Y,Yee_N_Z);
%%%% Initialise Graphics Support and display
Str1 = '';
Str2 = '';
Str3 = '';
%for drawing plots
[x,y]=meshgrid(1:1:Yee_N_X);            % very basic , could use all 3 dims

Vrecord = squeeze(zeros(TimeSteps,1)) + squeeze(i*zeros(TimeSteps,1));
Irecord = squeeze(zeros(TimeSteps,1)) + squeeze(i*zeros(TimeSteps,1));
%-------------------------------------------------------------------------------------------------------------------------
%% Initialise Ferrite coefficients
%The following adapted from  Inan and marshall pg 281 onwards
C1=(mu_0*W_0*(W_0 + W_M)/2) + (mu_0*A_Loss*((2*W_0) + W_M)/(2*Delta_T)) + (mu_0*(1 + A_Loss^2)/(Delta_T^2));
C2=((W_0^2)/2) + (A_Loss*W_0/Delta_T) + ((1 + A_Loss^2)/Delta_T^2);
C3=((W_0^2)/2) - (A_Loss*W_0/Delta_T) + ((1 + A_Loss^2)/Delta_T^2);
C4= -2*(1+A_Loss^2)/Delta_T^2;
C5= 2*mu_0*(1+A_Loss^2)/Delta_T^2;
C6= -(mu_0*W_0*(W_0 + W_M)/2) + (mu_0*A_Loss*((2*W_0) + W_M)/(2*Delta_T)) - (mu_0*(1 + A_Loss^2)/Delta_T^2);
C7= mu_0*W_M/(2*Delta_T);
C8=C7;
G = C1 + ((C8^2)/C1);
C1P=C2/G;
C2P=C3/G;
C3P=C4/G;
C4P=C5/G;
C5P=(C6+ (C7*C8/C1))/G;
C6P=(C7- (C6*C8/C1))/G;
C7P=(C2*C8/C1)/G;
C8P=(C3*C8/C1)/G;
C9P=(C4*C8/C1)/G;
C10P=(C5*C8/C1)/G;
Fr_Co = [ C1 C2 C3 C4 C5 C6 C7 C8 G C1P C2P C3P C4P C5P C6P C7P C8P C9P C10P];
%-------------------------------------------------------------------------------------------------------------------------
%% Initialise PML coefficients
 % function InitPMLConst
% Absorbing Boundary Conditions UPML Inan and Marshall Chapter 9
% modelled as two largest planes extending whole space at X  = 1:thickness and X = end, then
% 2 for Y, then 2 for Z
if PML_Graded == false
    Om_X(1+end-PML_T:end,:,:)=testss;            %set magnetic conductivity for PML , it should be graded but use 1 for now
    Om_X(1:PML_T,:,:)        =testss;
    Om_Y(:,1+end-PML_T:end,:)=testss;
    Om_Y(:,1:PML_T,:)      =testss;
    Om_Z(:,:,1+end-PML_T:end)=testss;
    Om_Z(:,:,1:PML_T)        =testss;
else
 %PML_Sigma_Max = - ((PML_m +1) * log(10^-6))/(2 * PML_Eta * PML_T * Delta_X)
 PML_Sigma_Max = - ((PML_m +1) * log(10^-6))/(2 * PML_Eta * PML_T * Delta_X);

  for ii=1:PML_T
      G_Sigma=PML_Sigma_Max*(ii/PML_T)^PML_m;   %starts small and grows to max, but max on outer boundary so...
      Om_X(PML_T-ii+1,:,:)  =G_Sigma;
      Om_X(end-PML_T+ii,:,:)=G_Sigma;
      Om_Y(:,PML_T-ii+1,:)  =G_Sigma;
      Om_Y(:,end-PML_T+ii,:)=G_Sigma;
```

```
      Om_Z(:,:,PML_T-ii+1)  =G_Sigma;
      Om_Z(:,:,end-PML_T+ii)=G_Sigma;
  end
end

PMDX1 = ((2*E_0*K_Y)-(Delta_T*Om_Y)) ./ ((2*E_0*K_Y)+(Delta_T*Om_Y));       % X --- Ky and Oy
PMDX2 = (2*E_0*Delta_T) / ((2*E_0*K_Y)+(Delta_T*Om_Y));

PMDY1=  ((2*E_0*K_Z)-(Delta_T*Om_Z)) ./ ((2*E_0*K_Z)+(Delta_T*Om_Z));       % Y --- Kz and Oz
PMDY2 = (2*E_0*Delta_T) /  ((2*E_0*K_Z)+(Delta_T*Om_Z));

PMDZ1 = ((2*E_0*K_X)-(Delta_T*Om_X)) ./ ((2*E_0*K_X)+(Delta_T*Om_X));        % Z ---- Kx and Ox
PMDZ2 = (2*E_0*Delta_T) / ((2*E_0*K_X)+(Delta_T*Om_X));

PMEX1 = ((2*E_0*K_Z)-(Delta_T*Om_Z)) ./ ((2*E_0*K_Z)+(Delta_T*Om_Z));
PMEX2 = ((1/PML_E)*( (2*E_0*K_X) + (Delta_T*Om_X) ) ./((2*E_0*K_Z)+(Delta_T*Om_Z)));
PMEX3 = ((1/PML_E)*( (2*E_0*K_X) - (Delta_T*Om_X) ) ./((2*E_0*K_Z)+(Delta_T*Om_Z)));

PMEY1 = ((2*E_0*K_X)-(Delta_T*Om_X)) ./ ((2*E_0*K_X)+(Delta_T*Om_X));
PMEY2 = ((1/PML_E)*( (2*E_0*K_Y) + (Delta_T*Om_Y) ) ./((2*E_0*K_X)+(Delta_T*Om_X)));
PMEY3 = ((1/PML_E)*( (2*E_0*K_Y) - (Delta_T*Om_Y) ) ./((2*E_0*K_X)+(Delta_T*Om_X)));

PMEZ1 = ((2*E_0*K_Y)-(Delta_T*Om_Y)) ./ ((2*E_0*K_Y)+(Delta_T*Om_Y));
PMEZ2 = ((1/PML_E)*( (2*E_0*K_Z) + (Delta_T*Om_Z) ) ./((2*E_0*K_Y)+(Delta_T*Om_Y)));
PMEZ3 = ((1/PML_E)*( (2*E_0*K_Z) - (Delta_T*Om_Z) ) ./((2*E_0*K_Y)+(Delta_T*Om_Y)));

PMBX1 = ((2*E_0*K_Y)-(Delta_T*Om_Y)) ./ ((2*E_0*K_Y)+(Delta_T*Om_Y));
PMBX2 = (2*E_0*Delta_T) / ((2*E_0*K_Y)+(Delta_T*Om_Y));

PMBY1 = ((2*E_0*K_Z)-(Delta_T*Om_Z)) ./ ((2*E_0*K_Z)+(Delta_T*Om_Z));
PMBY2 = (2*E_0*Delta_T) / ((2*E_0*K_Z)+(Delta_T*Om_Z));

PMBZ1 = ((2*E_0*K_Z)-(Delta_T*Om_X)) ./ ((2*E_0*K_X)+(Delta_T*Om_X));
PMBZ2 = (2*E_0*Delta_T) / ((2*E_0*K_X)+(Delta_T*Om_X));

PMHX1 = (((2*E_0*K_Z)-  (Delta_T*Om_Z)) ./((2*E_0*K_Z)+(Delta_T*Om_Z)));
PMHX2 = ((1/PML_U)*( (2*E_0*K_X) + (Delta_T*Om_X)) ./((2*E_0*K_Z)+(Delta_T*Om_Z)));
PMHX3 = ((1/PML_U)*( (2*E_0*K_X) - (Delta_T*Om_X)) ./((2*E_0*K_Z)+(Delta_T*Om_Z)));

PMHY1 = (((2*E_0*K_X)- (Delta_T*Om_X)) ./ ((2*E_0*K_X)+(Delta_T*Om_X)));
PMHY2 = ((1/PML_U)*( (2*E_0*K_Y) + (Delta_T*Om_Y)) ./((2*E_0*K_X)+(Delta_T*Om_X)));
PMHY3 = ((1/PML_U)*( (2*E_0*K_Y) - (Delta_T*Om_Y)) ./((2*E_0*K_X)+(Delta_T*Om_X)));

PMHZ1 = (((2*E_0*K_Y)-(Delta_T*Om_Y))  ./ ((2*E_0*K_Y)+(Delta_T*Om_Y)));
PMHZ2 = ((1/PML_U)*( (2*E_0*K_Z) + (Delta_T*Om_Z)) ./((2*E_0*K_Y)+(Delta_T*Om_Y)));
PMHZ3 = ((1/PML_U)*( (2*E_0*K_Z) - (Delta_T*Om_Z)) ./((2*E_0*K_Y)+(Delta_T*Om_Y)));
%% Main code
%put in 'blocks' on sides of cavity 2

if DontPlot == true;            %Stop all plots
Draw_Sim1 = false; Draw_Sim2 = false; Draw_Sim3 = false;
end

if Side_Blocks_PTFE == true   %lower X boundary is on PEC so offset by +1, Y ignored for now
Block_e = 1/(e_0*Block_re);
R_e_Y(Cavity2_Lower_Boundary_X+1:Cavity2_Lower_Boundary_X+1+(Block_Cells-
1),Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = Block_e;
R_e_Y(Cavity2_Upper_Boundary_X-(Block_Cells-
1):Cavity2_Upper_Boundary_X,Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary
_Z)= Block_e;
R_e_X=R_e_Y;
R_e_Z=R_e_Y;
end

if Side_Blocks_META == true   %lower X boundary is on PEC so offset by +1, Y ignored for now
Block_e = 1/(e_0*Block_re);
R_e_Y(Cavity2_Lower_Boundary_X+META_Offset,Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,Cavity2_Lower_Boundary_Z:Cavity2
_Upper_Boundary_Z) = Block_e;
R_e_Y(Cavity2_Upper_Boundary_X-
META_Offset,Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z)= Block_e;
R_e_X=R_e_Y;
R_e_Z=R_e_Y;

Number_Strips = round((Cavity2_Upper_Boundary_Y-Cavity2_Lower_Boundary_Y)/(Strip_Height_Cells+Strip_Gap_Cells))

Start_Y_Strip = Cavity2_Lower_Boundary_Y;           %Make map of outer strip E vectors to zero
 for Strip = 1:Number_Strips
    End_Y_Strip = Start_Y_Strip + Strip_Height_Cells;
    if Start_Y_Strip >= Cavity2_Upper_Boundary_Y    Start_Y_Strip = Cavity2_Upper_Boundary_Y; end;
    if End_Y_Strip   >= Cavity2_Upper_Boundary_Y    End_Y_Strip = Cavity2_Upper_Boundary_Y; end;
    %Left
    ZeroEY(Cavity2_Lower_Boundary_X+META_Offset,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    ZeroEZ(Cavity2_Lower_Boundary_X+META_Offset,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    %Right
    ZeroEY(Cavity2_Upper_Boundary_X-META_Offset,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    ZeroEZ(Cavity2_Upper_Boundary_X-META_Offset,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    Start_Y_Strip = Start_Y_Strip + Strip_Height_Cells + Strip_Gap_Cells;
 end

Start_Y_Strip = Cavity2_Lower_Boundary_Y + Strip_Offset_Cells;           %Make map of outer strip E vectors to zero
 for Strip = 1:Number_Strips
    End_Y_Strip = Start_Y_Strip + Strip_Height_Cells;
    if Start_Y_Strip >= Cavity2_Upper_Boundary_Y    Start_Y_Strip = Cavity2_Upper_Boundary_Y; end;
    if End_Y_Strip   >= Cavity2_Upper_Boundary_Y    End_Y_Strip = Cavity2_Upper_Boundary_Y; end;
    %Left
    ZeroEY(Cavity2_Lower_Boundary_X+META_Offset+1,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    ZeroEZ(Cavity2_Lower_Boundary_X+META_Offset+1,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    %Right
```

```
    ZeroEY(Cavity2_Upper_Boundary_X-META_Offset-1,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    ZeroEZ(Cavity2_Upper_Boundary_X-META_Offset-1,  Start_Y_Strip:End_Y_Strip
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = true;
    Start_Y_Strip = Start_Y_Strip + Strip_Height_Cells + Strip_Gap_Cells;
 end
end

if Dielectric_Antenna == true  %surround antenna with ptfe
Diel_e = 1/(e_0*Dielectric_e);
    for xx = 1:CellNumX
      for zz = 1:CellNumZ
       if   ((xx-Monopole_X_Coord)^2 + (zz-Monopole_Z_Coord)^2) < (Dielectric_Rad/Delta_Len)^2
          R_e_X(xx,Monopole_Y_Coord-1:Monopole_Y_Coord+Monopole_Y_Cells-1,zz) = Diel_e;
          R_e_Y(xx,Monopole_Y_Coord-1:Monopole_Y_Coord+Monopole_Y_Cells-1,zz) = Diel_e;
          R_e_Z(xx,Monopole_Y_Coord-1:Monopole_Y_Coord+Monopole_Y_Cells-1,zz) = Diel_e;
       end
      end
    end

   %but not in the gap 4 points around single cell - instability with 50ohm
   %source field due maybe to capacitance
   %R_e_X(Monopole_X_Coord:Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord:Monopole_Z_Coord+1)=1/(e_0);
   %R_e_Y(Monopole_X_Coord:Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord:Monopole_Z_Coord+1)=1/(e_0);
   %R_e_Z(Monopole_X_Coord:Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord:Monopole_Z_Coord+1)=1/(e_0);



end

wh = waitbar(0,'1','Name','Running Simulation',...
            'CreateCancelBtn',...
            'setappdata(gcbf,''cancelling'',1)');
ticID=tic;
for n=1:TimeSteps   %Make Movie
    Cycles_Done = Inject_F*n*Delta_T;

    Inject_Ramp = Cycles_Done/Ramp_Cycles;  %Make input amplitude ramp
    if Inject_Ramp > 1 Inject_Ramp = 1;
    end

    Source_Voltage = (Inject_Ramp*sin(Inject_W * Delta_T * n));

    Antenna_Voltage = Source_Voltage;

    Antenna_Voltage = Source_Voltage + (I_si*0);

    Antenna_E = Antenna_Voltage/ Delta_Len;



   if Animate == true %animate and maybe make movie of wave travel E_Y
    if rem(n,AnimationSteps)==0;
    %Fig9=figure(9+PlotNumber);
    %set(0,'CurrentFigure',9+PlotNumber);
     mesh(squeeze(E_Y(:,Centre_Y,:)));
    axis([1 CellNumZ 1 CellNumX -100.0 100.0 -100.0 100.0]);
    pbaspect([CellNumZ/CellNumX 1 1 ]);
    view(30,40);
    line( [Cavity1_Lower_Boundary_Z-1,Cavity1_Lower_Boundary_Z-1] , [Cavity1_Lower_Boundary_X-1,Cavity1_Upper_Boundary_X+1]);
    line( [Cavity1_Lower_Boundary_Z-1,Cavity1_Upper_Boundary_Z-1] , [Cavity1_Lower_Boundary_X-1,Cavity1_Lower_Boundary_X-1]);
    line( [Cavity1_Lower_Boundary_Z-1,Cavity1_Upper_Boundary_Z-1] , [Cavity1_Upper_Boundary_X+1,Cavity1_Upper_Boundary_X+1]);

    line( [Cavity2_Lower_Boundary_Z-1,Cavity2_Lower_Boundary_Z-1] , [Cavity2_Lower_Boundary_X-1,Cavity1_Lower_Boundary_X-1]);
    line( [Cavity2_Lower_Boundary_Z-1,Cavity2_Lower_Boundary_Z-1] , [Cavity1_Upper_Boundary_X+1,Cavity2_Upper_Boundary_X+1]);

    line( [Cavity2_Lower_Boundary_Z-1,Cavity2_Upper_Boundary_Z] , [Cavity2_Lower_Boundary_X-1,Cavity2_Lower_Boundary_X-1]);
    line( [Cavity2_Lower_Boundary_Z-1,Cavity2_Upper_Boundary_Z] , [Cavity2_Upper_Boundary_X+1,Cavity2_Upper_Boundary_X+1]);

    drawnow;
      if Make_Movie == false
       Animation(Astep)=getframe(gcf);
       Astep = Astep+1;
      else
          %ff=getframe(Fig9);
          writeVideo(writerObj,getframe(gcf));
          %aviObj = addframe(aviObj,Fig9);
      end
    end
   end

    %-------------------------------------------------------------------------------------------------------------
    if PML == true;
    %PML H --> D
    PMLCalcDfromH(2,PML_T,                    2,CellNumY,               2,CellNumZ)                    %PML Plane at X=1
    PMLCalcDfromH(CellNumX-PML_T,CellNumX,    2,CellNumY,               2,CellNumZ)                    %PML Plane at X=end

    PMLCalcDfromH(2,CellNumX,                 2,PML_T,                  2,CellNumZ)                    %PML Plane at Y=1
    PMLCalcDfromH(2,CellNumX,                 CellNumY-PML_T,CellNumY,  2,CellNumZ)                    %PML Plane at Y=end

    PMLCalcDfromH(2,CellNumX,                 2,CellNumY,               2,PML_T)                       %PML Plane at Z=1
    PMLCalcDfromH(2,CellNumX,                 2,CellNumY,               CellNumZ-PML_T,CellNumZ)       %PML Plane at X=end
    % -------------------
    % PML D --> E
    PMLCalcEfromD(2,PML_T,                    2,CellNumY,               2,CellNumZ)                    %PML Plane at X=1
    PMLCalcEfromD(CellNumX-PML_T,CellNumX,    2,CellNumY,               2,CellNumZ)                    %PML Plane at X=end

    PMLCalcEfromD(2,CellNumX,                2,PML_T,                   2,CellNumZ)                    %PML Plane at Y=1
    PMLCalcEfromD(2,CellNumX,                CellNumY-PML_T,CellNumY,   2,CellNumZ)                    %PML Plane at Y=end

    PMLCalcEfromD(2,CellNumX,                2,CellNumY,                2,PML_T)                       %PML Plane at Z=1
    PMLCalcEfromD(2,CellNumX,                2,CellNumY,                CellNumZ-PML_T,CellNumZ)       %PML Plane at Z=end
    end %of if PML
    %-------------------------------------------------------------------------------------------------------------
```

```
    % free space area
   FCalcDfromH(PML_T+1,CellNumX-PML_T-1,    PML_T+1,CellNumY-PML_T-1,    PML_T+1,CellNumZ-PML_T-1);   % from PML to end
   FCalcEfromD(PML_T+1,CellNumX-PML_T-1,    PML_T+1,CellNumY-PML_T-1,    PML_T+1,CellNumZ-PML_T-1);
  %-------------------------------------------------------------------------------------------------------------
    %ferrite
  if Ferrite_Side_Blocks_PMC==true
    E_X(Ferrite_Loc)=(1/(f_e))*D_X(Ferrite_Loc);
    E_Y(Ferrite_Loc)=(1/(f_e))*D_Y(Ferrite_Loc);
    E_Z(Ferrite_Loc)=(1/(f_e))*D_Z(Ferrite_Loc);
  end;
  %-------------------------------------------------------------------------------------------------------------
  % Source
   if Dim_1_Antenna == true
     E_Y(Monopole_X_Coord,Monopole_Y_Coord-1,Monopole_Z_Coord)= Antenna_E; % for use with 4 step free space
     E_Y(Monopole_X_Coord, Monopole_Y_Coord:Monopole_Y_Coord+Monopole_Y_Cells-1, Monopole_Z_Coord) = 0;
   else   %else pec handled by general pec function
     E_Y(Monopole_X_Coord:Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord:Monopole_Z_Coord+1)= Antenna_E; % for use
with 4 step free space
   end

    if Debug==true
        % lower dipole
        % E_Y(Monopole_X_Coord, Monopole_Y_Coord-2-(Monopole_Y_Cells-1):Monopole_Y_Coord-2, Monopole_Z_Coord) = 0;
    end
     %Calc_Source_Params;
  %-------------------------------------------------------------------------------------------------------------
    % do skin
    if Skin==true

    %override previous calced version of E_X by using E_X_P etc
    SCalcEfromH(Skin_Lower_Boundary_X,Skin_Upper_Boundary_X,  Skin_Lower_Boundary_Y,Skin_Upper_Boundary_Y,
Skin_Lower_Boundary_Z,Skin_Upper_Boundary_Z);
    end
  %-------------------------------------------------------------------------------------------------------------
    %Enforce PEC
       if Pec_Method_1 == true
         Cavity_Boundary;
        end

    E_X(ZeroEX)=0;          %DO pec stuff like strips, antenna
    E_Y(ZeroEY)=0;
    E_Z(ZeroEZ)=0;

    %-------------------------------------------------------------------------------------------------------------
    %Calc Magnetic fields from E
    %-------------------------------------------------------------------------------------------------------------
    % PML E --> B
    if PML == true
    PMLCalcBfromE(2,PML_T,                    2,CellNumY-1,               2,CellNumZ-1)               %PML Plane at X=1
    PMLCalcBfromE(CellNumX-PML_T,CellNumX-1,  2,CellNumY-1,               2,CellNumZ-1)               %PML Plane at X=end

    PMLCalcBfromE(2,CellNumX-1,               2,PML_T,                    2,CellNumZ-1)               %PML Plane at Y=1
    PMLCalcBfromE(2,CellNumX-1,               CellNumY-PML_T,CellNumY-1,  2,CellNumZ-1)               %PML Plane at Y=end

    PMLCalcBfromE(2,CellNumX-1,               2,CellNumY-1,               2,PML_T)                    %PML Plane at Z=1
    PMLCalcBfromE(2,CellNumX-1,               2,CellNumY-1,               CellNumZ-PML_T,CellNumZ-1)  %PML Plane at Z=end
    %-----------------------------------------------------
    % PML B --> H
    PMLCalcHfromB(2,PML_T,                    2,CellNumY-1,               2,CellNumZ-1)               %PML Plane at X=1
    PMLCalcHfromB(CellNumX-PML_T,CellNumX-1,  2,CellNumY-1,               2,CellNumZ-1)               %PML Plane at X=end

    PMLCalcHfromB(2,CellNumX-1,               2,PML_T,                    2,CellNumZ-1)               %PML Plane at Y=1
    PMLCalcHfromB(2,CellNumX-1,               CellNumY-PML_T,CellNumY-1,  2,CellNumZ-1)               %PML Plane at Y=end

    PMLCalcHfromB(2,CellNumX-1,               2,CellNumY-1,               2,PML_T)                    %PML Plane at Z=1
    PMLCalcHfromB(2,CellNumX-1,               2,CellNumY-1,               CellNumZ-PML_T,CellNumZ-1)  %PML Plane at Z=end
    end % of if PML
    %-------------------------------------------------------------------------------------------------------------
    %Free space
    FCalcBfromE(PML_T+1,CellNumX-PML_T-1,    PML_T+1,CellNumY-PML_T-1,    PML_T+1,CellNumZ-PML_T-1)          %E-->B
    FCalcHfromB(PML_T+1,CellNumX-PML_T-1,    PML_T+1,CellNumY-PML_T-1,    PML_T+1,CellNumZ-PML_T-1)          %B-->H
    %-------------------------------------------------------------------------------------------------------------
    %-------------------------------------------------------------------------------------------------------------
    if Simple_Side_Blocks_PMC == true

     %Cavity 1
     %Left
    B_Y(Cavity1_Lower_Boundary_X+1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = 0;
    B_Z(Cavity1_Lower_Boundary_X+1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = 0;
  %Right
    B_Y(Cavity1_Upper_Boundary_X-1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = 0;
    B_Z(Cavity1_Upper_Boundary_X-1,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y
,Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z) = 0;


    %Cavity2
    %Left
    B_Y(Cavity2_Lower_Boundary_X+1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = 0;
    B_Z(Cavity2_Lower_Boundary_X+1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = 0;
  %Right
    B_Y(Cavity2_Upper_Boundary_X-1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = 0;
    B_Z(Cavity2_Upper_Boundary_X-1,  Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y
,Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z) = 0;

    end
    %-------------------------------------------------------------------------------------------------------------
    if Ferrite_Side_Blocks_PMC==true
```

```
    H_Z(Ferrite_Loc) = (C1P*B_Z(Ferrite_Loc)) + (C2P*B_Z_PP(Ferrite_Loc)) + (C3P*B_Z_P(Ferrite_Loc)) +
(C4P*H_Z_P(Ferrite_Loc)) + (C5P*H_Z_PP(Ferrite_Loc)) + (C6P*H_Y_PP(Ferrite_Loc)) - (C7P*B_Y(Ferrite_Loc)) -
(C8P*B_Y_PP(Ferrite_Loc)) - (C9P*B_Y_P(Ferrite_Loc)) - (C10P*H_Y_P(Ferrite_Loc));
    H_Y(Ferrite_Loc) = (C1P*B_Y(Ferrite_Loc)) + (C2P*B_Y_PP(Ferrite_Loc)) + (C3P*B_Y_P(Ferrite_Loc)) +
(C4P*H_Y_P(Ferrite_Loc)) + (C5P*H_Y_PP(Ferrite_Loc)) - (C6P*H_Z_PP(Ferrite_Loc)) + (C7P*B_Z(Ferrite_Loc)) +
(C8P*B_Z_PP(Ferrite_Loc)) + (C9P*B_Z_P(Ferrite_Loc)) + (C10P*H_Z_P(Ferrite_Loc));
    H_X(Ferrite_Loc) = (1/(mu_0)) * B_X(Ferrite_Loc);
    end;
    %-------------------------------------------------------------------------------------------------------------------
    %keep time series
    % H_X etc is at n-1 = previous
    %H_X_PP = H_X_P;
    H_X_P = H_X;

    H_Y_PP = H_Y_P;
    H_Y_P = H_Y;

    H_Z_PP = H_Z_P;
    H_Z_P = H_Z;       % HZPP etc not used but kept in for symmetry

    %keep B time series
    %B_X_PP = B_X_P;
    B_X_P = B_X;

    B_Y_PP = B_Y_P;
    B_Y_P = B_Y;

    B_Z_PP = B_Z_P;

    B_Z_P = B_Z;
    D_X_P = D_X;
    D_Y_P = D_Y;
    D_Z_P = D_Z;
    E_X_P = E_X;
    E_Y_P = E_Y;
    E_Z_P = E_Z;
    % make intensity record
    if Cycles_Done>=Capture_Start_Cycles
        E2 = E2 + (E_X.^2) + (E_Y.^2) + (E_Z.^2);

    end
    Calc_Source_Params;
    nt=n;                               % keep a copy of number of steps done


   %-------------------------------------------------------------------------------------------------------------------

    if Draw_Sim2 == true
    DrawSim2;
    end
    if Draw_Sim3 == true         %
    DrawSim3;
    end


    if rem(n,2)==0;                 % Display waitbar
        elapsedT = toc(ticID);
        remainingT = (TimeSteps-n)*(elapsedT/n);
        hoursL = fix(remainingT/3600);
        minsL = fix( (remainingT - (hoursL * 3600))/60);
        totalT = TimeSteps*(elapsedT/n);
        hoursT = fix(totalT/3600);
        minsT = fix( (totalT - (hoursT * 3600))/60);
        msgstr = ['Time left = ' num2str(hoursL,'%10.0f') ':' num2str(minsL,'%02.0f') ' of ' num2str(hoursT,'%10.0f') ':'
num2str(minsT,'%02.0f') ' Completed =' num2str(100*n/TimeSteps,'%10.2f') '%'];
        waitbar(n/TimeSteps,wh,msgstr);
        if getappdata(wh,'cancelling');
        delete(wh)
        break
        end
    end

end   %of sim loop
%memory;
    if Draw_Sim1 == true
        DrawSim1;                   % plot progress
    end

    elapsedTime = toc(ticID);
E_s
I_s
ZZ
    if ishandle(wh)             % if waitbar then delete it
        delete(wh)
%keyboard;
    end
    F = findall(0,'type','figure','tag','TMWWaitbar'); delete(F);   %and any other ones

    if Animate == true
      if Make_Movie == true
        close(writerObj);
        %aviObj=close(aviObj);
      end
    end

    if DontPlot == false
    DoFinalPlots1;                      %various depths of penetration/intensity
    end

return ;
%% Make display strings of simulation parameters
%% Maxwell Functions
%-------------------------------------------------------------------------------------------------------------------
```

```
% Main functions for calculating fields
    function Calc_Source_Params
         if Dim_1_Antenna == true
    I_siv = Delta_Len*( + H_X(Monopole_X_Coord,:,(Monopole_Z_Coord)-1) + H_Z(Monopole_X_Coord,:,Monopole_Z_Coord) -
H_X(Monopole_X_Coord,:,Monopole_Z_Coord) - H_Z((Monopole_X_Coord)-1,:,Monopole_Z_Coord)) ; %old
    E_si = E_Y(Monopole_X_Coord,Monopole_Y_Coord-1,Monopole_Z_Coord);
         else                %Else its solid block
    I_si_v = ( + H_X(Monopole_X_Coord,:, Monopole_Z_Coord-1) ...
               + H_X(Monopole_X_Coord+1,:, Monopole_Z_Coord-1) ...
               + H_Z(Monopole_X_Coord+1,:, Monopole_Z_Coord  ) ...
               + H_Z(Monopole_X_Coord+1,:, Monopole_Z_Coord+1) ...
               - H_X(Monopole_X_Coord+1,:, Monopole_Z_Coord+1) ...   %was H_X(Monopole_X_Coord+1,:, Monopole_Z_Coord-1)
               - H_X(Monopole_X_Coord  ,:, Monopole_Z_Coord+1) ...   %was H_X(Monopole_X_Coord  ,:, Monopole_Z_Coord-1)
               - H_Z(Monopole_X_Coord-1,:, Monopole_Z_Coord+1) ...
               - H_Z(Monopole_X_Coord-1,:, Monopole_Z_Coord  )     )*Delta_Len;

     % for E, take average of 4 points -- they should be the same?
      E_si =(+ E_Y(Monopole_X_Coord  ,Monopole_Y_Coord-1,Monopole_Z_Coord  ) ...
             + E_Y(Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord  ) ...
             + E_Y(Monopole_X_Coord  ,Monopole_Y_Coord-1,Monopole_Z_Coord+1) ...
             + E_Y(Monopole_X_Coord+1,Monopole_Y_Coord-1,Monopole_Z_Coord+1) )/4 ;
         end

    I_si = I_si_v(Monopole_Y_Coord+1);
    Vrecord(n) = E_si*Delta_Len;
    Irecord(n) = I_si;

    if Cycles_Done>=Capture_Start_Cycles
     E_s = E_s + (E_si* cos(Inject_W * n * Delta_T)) - (i * E_si * (sin(Inject_W * n * Delta_T)));
     I_s = I_s + (I_si* cos(Inject_W * n * Delta_T)) - (i * I_si * (sin(Inject_W * n * Delta_T)));
     ZZ = (E_s*Delta_Len)/I_s;
    end


    end
    function PMLCalcDfromH(Xs,Xe,Ys,Ye,Zs,Ze)       %PML Xstart,Xend,Ystart etc
    % For simplicity, plane at x,y,z = 1 not calculated
    D_X(Xs:Xe,Ys:Ye,Zs:Ze)= (PMDX1(Xs:Xe,Ys:Ye,Zs:Ze).* D_X_P(Xs:Xe,Ys:Ye,Zs:Ze))+ (PMDX2(Xs:Xe,Ys:Ye,Zs:Ze).*
((diff(H_Z(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y) - (diff(H_Y(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z)));
    D_Y(Xs:Xe,Ys:Ye,Zs:Ze)= (PMDY1(Xs:Xe,Ys:Ye,Zs:Ze).* D_Y_P(Xs:Xe,Ys:Ye,Zs:Ze))+ (PMDY2(Xs:Xe,Ys:Ye,Zs:Ze).*
((diff(H_X(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z) - (diff(H_Z(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X)));
    D_Z(Xs:Xe,Ys:Ye,Zs:Ze)= (PMDZ1(Xs:Xe,Ys:Ye,Zs:Ze).* D_Z_P(Xs:Xe,Ys:Ye,Zs:Ze))+ (PMDZ2(Xs:Xe,Ys:Ye,Zs:Ze).* ((diff(H_Y(Xs-
1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X) - (diff(H_X(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y)));
    end
    function PMLCalcEfromD(Xs,Xe,Ys,Ye,Zs,Ze)       %PML
    E_X(Xs:Xe,Ys:Ye,Zs:Ze)  =   (PMEX1(Xs:Xe,Ys:Ye,Zs:Ze) .* E_X_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMEX2(Xs:Xe,Ys:Ye,Zs:Ze) .*
D_X(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMEX3(Xs:Xe,Ys:Ye,Zs:Ze) .* D_X_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;
    E_Y(Xs:Xe,Ys:Ye,Zs:Ze)  =   (PMEY1(Xs:Xe,Ys:Ye,Zs:Ze) .* E_Y_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMEY2(Xs:Xe,Ys:Ye,Zs:Ze) .*
D_Y(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMEY3(Xs:Xe,Ys:Ye,Zs:Ze) .* D_Y_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;
    E_Z(Xs:Xe,Ys:Ye,Zs:Ze)  =   (PMEZ1(Xs:Xe,Ys:Ye,Zs:Ze) .* E_Z_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMEZ2(Xs:Xe,Ys:Ye,Zs:Ze) .*
D_Z(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMEZ3(Xs:Xe,Ys:Ye,Zs:Ze) .* D_Z_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;
    end
    function FCalcEfromH(Xs,Xe,Ys,Ye,Zs,Ze)         %Free space
    E_X(Xs:Xe,Ys:Ye,Zs:Ze) = E_X(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T/Fe_0)* ((diff(H_Z(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y) -
diff(H_Y(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z));
    E_Y(Xs:Xe,Ys:Ye,Zs:Ze) = E_Y(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T/Fe_0)* ((diff(H_X(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z) -
diff(H_Z(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X));
    E_Z(Xs:Xe,Ys:Ye,Zs:Ze) = E_Z(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T/Fe_0)* ((diff(H_Y(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X) -
diff(H_X(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y));
    end
     function SCalcEfromH(Xs,Xe,Ys,Ye,Zs,Ze);       %NOTE USES E_X_P, E_Y_P etc to override free space calc
     E_X(Xs:Xe,Ys:Ye,Zs:Ze) = SC1*E_X_P(Xs:Xe,Ys:Ye,Zs:Ze) + (SC2* ((diff(H_Z(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y) -
diff(H_Y(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z));
     E_Y(Xs:Xe,Ys:Ye,Zs:Ze) = SC1*E_Y_P(Xs:Xe,Ys:Ye,Zs:Ze) + (SC2* ((diff(H_X(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z) -
diff(H_Z(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X));
     E_Z(Xs:Xe,Ys:Ye,Zs:Ze) = SC1*E_Z_P(Xs:Xe,Ys:Ye,Zs:Ze) + (SC2* ((diff(H_Y(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X) -
diff(H_X(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y));
     end
    function FCalcDfromH(Xs,Xe,Ys,Ye,Zs,Ze)         %Free space
    D_X(Xs:Xe,Ys:Ye,Zs:Ze) = D_X(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T)* ((diff(H_Z(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y) -
diff(H_Y(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z));
    D_Y(Xs:Xe,Ys:Ye,Zs:Ze) = D_Y(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T)* ((diff(H_X(Xs:Xe,  Ys:Ye,Zs-1:Ze),1,3)/Delta_Z) -
diff(H_Z(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X));
    D_Z(Xs:Xe,Ys:Ye,Zs:Ze) = D_Z(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T)* ((diff(H_Y(Xs-1:Xe,Ys:Ye,  Zs:Ze),1,1)/Delta_X) -
diff(H_X(Xs:Xe,Ys-1:Ye,  Zs:Ze),1,2)/Delta_Y));
    end
    function FCalcEfromD(Xs,Xe,Ys,Ye,Zs,Ze)         %Free space
    %E_X(Xs:Xe,Ys:Ye,Zs:Ze) = (1/Fe_0)*D_X(Xs:Xe,Ys:Ye,Zs:Ze) ;
    %E_Y(Xs:Xe,Ys:Ye,Zs:Ze) = (1/Fe_0)*D_Y(Xs:Xe,Ys:Ye,Zs:Ze) ;
    %E_Z(Xs:Xe,Ys:Ye,Zs:Ze) = (1/Fe_0)*D_Z(Xs:Xe,Ys:Ye,Zs:Ze) ;

    E_X(Xs:Xe,Ys:Ye,Zs:Ze) = R_e_X(Xs:Xe,Ys:Ye,Zs:Ze).*D_X(Xs:Xe,Ys:Ye,Zs:Ze) ;    %use for different permittivity per cell
    E_Y(Xs:Xe,Ys:Ye,Zs:Ze) = R_e_Y(Xs:Xe,Ys:Ye,Zs:Ze).*D_Y(Xs:Xe,Ys:Ye,Zs:Ze) ;
    E_Z(Xs:Xe,Ys:Ye,Zs:Ze) = R_e_Z(Xs:Xe,Ys:Ye,Zs:Ze).*D_Z(Xs:Xe,Ys:Ye,Zs:Ze) ;

    end
    function PMLCalcBfromE(Xs,Xe,Ys,Ye,Zs,Ze)
% SIGN INSIDE BRACKETS FOR CURL SEEMS WRONG WAY ROUND SEE ININ AND MARSHALL pg 224 9.83
    B_X(Xs:Xe,Ys:Ye,Zs:Ze)  = (PMBX1(Xs:Xe,Ys:Ye,Zs:Ze) .* B_X_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMBX2(Xs:Xe,Ys:Ye,Zs:Ze) .*
((diff(E_Y(Xs:Xe,Ys:Ye,Zs:Ze+1),1,3)/Delta_Z) - (diff(E_Z(Xs:Xe,Ys:Ye+1,Zs:Ze),1,2)/Delta_Y)));
    B_Y(Xs:Xe,Ys:Ye,Zs:Ze)  = (PMBY1(Xs:Xe,Ys:Ye,Zs:Ze) .* B_Y_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMBY2(Xs:Xe,Ys:Ye,Zs:Ze) .*
((diff(E_Z(Xs:Xe+1,Ys:Ye,Zs:Ze),1,1)/Delta_X) - (diff(E_X(Xs:Xe,Ys:Ye,Zs:Ze+1),1,3)/Delta_Z)));
    B_Z(Xs:Xe,Ys:Ye,Zs:Ze)  = (PMBZ1(Xs:Xe,Ys:Ye,Zs:Ze) .* B_Z_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMBZ2(Xs:Xe,Ys:Ye,Zs:Ze) .*
((diff(E_X(Xs:Xe,Ys:Ye+1,Zs:Ze),1,2)/Delta_Y) - (diff(E_Y(Xs:Xe+1,Ys:Ye,Zs:Ze),1,1)/Delta_X)));
    end
    function PMLCalcHfromB(Xs,Xe,Ys,Ye,Zs,Ze)
    H_X(Xs:Xe,Ys:Ye,Zs:Ze)     = (PMHX1(Xs:Xe,Ys:Ye,Zs:Ze)  .* H_X_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMHX2(Xs:Xe,Ys:Ye,Zs:Ze)  .*
B_X(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMHX3(Xs:Xe,Ys:Ye,Zs:Ze)  .* B_X_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;          %B_X_P = previous
    H_Y(Xs:Xe,Ys:Ye,Zs:Ze)     = (PMHY1(Xs:Xe,Ys:Ye,Zs:Ze)  .* H_Y_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMHY2(Xs:Xe,Ys:Ye,Zs:Ze)  .*
B_Y(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMHY3(Xs:Xe,Ys:Ye,Zs:Ze)  .* B_Y_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;
    H_Z(Xs:Xe,Ys:Ye,Zs:Ze)     = (PMHZ1(Xs:Xe,Ys:Ye,Zs:Ze)  .* H_Z_P(Xs:Xe,Ys:Ye,Zs:Ze)) + (PMHZ2(Xs:Xe,Ys:Ye,Zs:Ze)  .*
B_Z(Xs:Xe,Ys:Ye,Zs:Ze)) - (PMHZ3(Xs:Xe,Ys:Ye,Zs:Ze)  .* B_Z_P(Xs:Xe,Ys:Ye,Zs:Ze)) ;
    end
    function FCalcBfromE(Xs,Xe,Ys,Ye,Zs,Ze)         %free space
```

```
    B_X(Xs:Xe,Ys:Ye,Zs:Ze) = B_X(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T) * ( (diff(E_Y(Xs:Xe  ,Ys:Ye  ,Zs:Ze+1),1,3)/Delta_Z) -
(diff(E_Z(Xs:Xe  ,Ys:Ye+1,Zs:Ze  ),1,2)/Delta_Y)));
    B_Y(Xs:Xe,Ys:Ye,Zs:Ze) = B_Y(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T) * ( (diff(E_Z(Xs:Xe+1,Ys:Ye  ,Zs:Ze  ),1,1)/Delta_X) -
(diff(E_X(Xs:Xe  ,Ys:Ye  ,Zs:Ze+1),1,3)/Delta_Z)));
    B_Z(Xs:Xe,Ys:Ye,Zs:Ze) = B_Z(Xs:Xe,Ys:Ye,Zs:Ze) + ((Delta_T) * ( (diff(E_X(Xs:Xe  ,Ys:Ye+1,Zs:Ze  ),1,2)/Delta_Y) -
(diff(E_Y(Xs:Xe+1,Ys:Ye  ,Zs:Ze  ),1,1)/Delta_X)));
    end
    function FCalcHfromB(Xs,Xe,Ys,Ye,Zs,Ze)
    H_X(Xs:Xe,Ys:Ye,Zs:Ze)=(1/Fmu_0)*B_X(Xs:Xe,Ys:Ye,Zs:Ze);
    H_Y(Xs:Xe,Ys:Ye,Zs:Ze)=(1/Fmu_0)*B_Y(Xs:Xe,Ys:Ye,Zs:Ze);
    H_Z(Xs:Xe,Ys:Ye,Zs:Ze)=(1/Fmu_0)*B_Z(Xs:Xe,Ys:Ye,Zs:Ze);
    end
    function Cavity_Boundary                        %Do E boundary conditions of cavity
    % do boundary cavity = air, around cavity = pec

    %Cavity 1

    %bottom
    E_X(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y-1,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y-1,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;

    %top
    E_X(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Upper_Boundary_Y+1,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Upper_Boundary_Y+1,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;

    %left
    E_Y(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;

    %right
    E_Y(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z:Cavity1_Upper_Boundary_Z-1)=0;

    %rear
    E_X(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity1_Lower_Boundary_Z-1)=0;


    %Bot rear horiz corner
    E_X(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Lower_Boundary_Z-1)=0;

    %Top rear horiz corner
    E_X(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1:Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;

    %Left rear vert corner
    E_X(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;

    %Right rear vert corner
    E_X(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Y(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;
    E_Z(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y-1:Cavity1_Upper_Boundary_Y+1,  Cavity1_Lower_Boundary_Z-1)=0;


    %top Front corner of cavity1
    E_X(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Upper_Boundary_Z)=0;

    %bot Front corner of cavity1
    E_X(Cavity1_Lower_Boundary_X:Cavity1_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Upper_Boundary_Z)=0;

    %Left vert front corner cavity1
    E_Y(Cavity1_Lower_Boundary_X-1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,  Cavity1_Upper_Boundary_Z)=0;

    %Right vert front corner cavity1
    E_Y(Cavity1_Upper_Boundary_X+1,   Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,  Cavity1_Upper_Boundary_Z)=0;


    %Front corners
    %Bot left
    E_X(Cavity1_Lower_Boundary_X-1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Upper_Boundary_Z)=0;
    E_Y(Cavity1_Lower_Boundary_X-1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Upper_Boundary_Z)=0;
    %Bot right
    E_X(Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Upper_Boundary_Z)=0;
    E_Y(Cavity1_Upper_Boundary_X+1,  Cavity1_Lower_Boundary_Y-1,  Cavity1_Upper_Boundary_Z)=0;
    %Top left
    E_X(Cavity1_Lower_Boundary_X-1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Upper_Boundary_Z)=0;
    E_Y(Cavity1_Lower_Boundary_X-1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Upper_Boundary_Z)=0;
    %Top right
    E_X(Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Upper_Boundary_Z)=0;
    E_Y(Cavity1_Upper_Boundary_X+1,  Cavity1_Upper_Boundary_Y+1,  Cavity1_Upper_Boundary_Z)=0;


    %CAVITY1 OUTSIDE CORNERS IN Z DIR (Some overlap with rear but ok)
    %Bot left
    E_X(Cavity1_Lower_Boundary_X-1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X-1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;

    %Bot right
```

```
    E_X(Cavity1_Upper_Boundary_X+1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Y(Cavity1_Upper_Boundary_X+1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Upper_Boundary_X+1, Cavity1_Lower_Boundary_Y-1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;

    %top left
    E_X(Cavity1_Lower_Boundary_X-1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Y(Cavity1_Lower_Boundary_X-1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Lower_Boundary_X-1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;

    %top right
    E_X(Cavity1_Upper_Boundary_X+1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Y(Cavity1_Upper_Boundary_X+1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    E_Z(Cavity1_Upper_Boundary_X+1, Cavity1_Upper_Boundary_Y+1, Cavity1_Lower_Boundary_Z-1:Cavity1_Upper_Boundary_Z-1)=0;
    %-------------------------------------------------------------------------------------------------------------------------------
---
    %-------------------------------------------------------------------------------------------------------------------------------
---
    %Cavity 2
    %bottom
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Lower_Boundary_Y-1,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Lower_Boundary_Y-1,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

    %top
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Upper_Boundary_Y+1,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Upper_Boundary_Y+1,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

    %left
    E_Y(Cavity2_Lower_Boundary_X-1,   Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1,   Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

    %right
    E_Y(Cavity2_Upper_Boundary_X+1,   Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Upper_Boundary_X+1,   Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

     %corners already done in cavity 1
     %Rear bottom
     E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Lower_Boundary_Y:Cavity1_Lower_Boundary_Y-2,
Cavity2_Lower_Boundary_Z-1)=0;
     E_Y(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity2_Lower_Boundary_Y:Cavity1_Lower_Boundary_Y-2,
Cavity2_Lower_Boundary_Z-1)=0;

     %Rear top
     E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity1_Upper_Boundary_Y+2:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;
     E_Y(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X,  Cavity1_Upper_Boundary_Y+2:Cavity2_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;

     %Rear left
     E_X(Cavity2_Lower_Boundary_X:Cavity1_Lower_Boundary_X-2,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;
     E_Y(Cavity2_Lower_Boundary_X:Cavity1_Lower_Boundary_X-2,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;

     %Rear right
     E_X(Cavity1_Upper_Boundary_X+2:Cavity2_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;
     E_Y(Cavity1_Upper_Boundary_X+2:Cavity2_Upper_Boundary_X,  Cavity1_Lower_Boundary_Y:Cavity1_Upper_Boundary_Y,
Cavity2_Lower_Boundary_Z-1)=0;

    %-----------------------------------------------------------------------------------------------------------
    %rear outer corner of cavity2 - outside of cavity
    %Bottom
    E_X(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Y(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z-1)=0;

    %Top
    E_X(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Y(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;

    %Left
    E_X(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Y(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;

    %Right
    E_X(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Y(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    E_Z(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1:Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z-1)=0;
    %-----------------------------------------------------------------------------------------------------------
    %Cavity2 - corners in Z direction
    %Bottom Left
    E_X(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Y(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

     %Top Left
    E_X(Cavity2_Lower_Boundary_X-1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Y(Cavity2_Lower_Boundary_X-1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Lower_Boundary_X-1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;

    %Bottom Right
    E_X(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Y(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Upper_Boundary_X+1,  Cavity2_Lower_Boundary_Y-1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
```

```
    %Top Right
    E_X(Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Y(Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;
    E_Z(Cavity2_Upper_Boundary_X+1,  Cavity2_Upper_Boundary_Y+1,   Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1)=0;


    %Corners at end of cavity 2
    %Bottom
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Lower_Boundary_Y-1, Cavity2_Upper_Boundary_Z) = 0;

    %Top
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+1, Cavity2_Upper_Boundary_Z) = 0;

    %Left
    E_Y(Cavity2_Lower_Boundary_X-1, Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y, Cavity2_Upper_Boundary_Z) = 0;

    %Right
    E_Y(Cavity2_Upper_Boundary_X+1, Cavity2_Lower_Boundary_Y:Cavity2_Upper_Boundary_Y, Cavity2_Upper_Boundary_Z) = 0;

    %Bot left
    E_X(Cavity2_Lower_Boundary_X-1, Cavity2_Lower_Boundary_Y-1, Cavity2_Upper_Boundary_Z) = 0;
    E_Y(Cavity2_Lower_Boundary_X-1, Cavity2_Lower_Boundary_Y-1, Cavity2_Upper_Boundary_Z) = 0;
    %Top left
    E_X(Cavity2_Lower_Boundary_X-1, Cavity2_Upper_Boundary_Y+1, Cavity2_Upper_Boundary_Z) = 0;
    E_Y(Cavity2_Lower_Boundary_X-1, Cavity2_Upper_Boundary_Y+1, Cavity2_Upper_Boundary_Z) = 0;
    %Bot right
    E_X(Cavity2_Upper_Boundary_X+1, Cavity2_Lower_Boundary_Y-1, Cavity2_Upper_Boundary_Z) = 0;
    E_Y(Cavity2_Upper_Boundary_X+1, Cavity2_Lower_Boundary_Y-1, Cavity2_Upper_Boundary_Z) = 0;
    %Top Right
    E_X(Cavity2_Upper_Boundary_X+1, Cavity2_Upper_Boundary_Y+1, Cavity2_Upper_Boundary_Z) = 0;
    E_Y(Cavity2_Upper_Boundary_X+1, Cavity2_Upper_Boundary_Y+1, Cavity2_Upper_Boundary_Z) = 0;

    %TO DO front flange
    %Bottom
    E_X(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Cavity2_Lower_Boundary_Y-2,
Cavity2_Upper_Boundary_Z)=0;
    E_Y(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Cavity2_Lower_Boundary_Y-2,
Cavity2_Upper_Boundary_Z)=0;
    %Top
    E_X(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+2:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;
    E_Y(Front_Lower_Boundary_X:Front_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+2:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;
    %Left
    E_X(Front_Lower_Boundary_X:Cavity2_Lower_Boundary_X-2, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;
    E_Y(Front_Lower_Boundary_X:Cavity2_Lower_Boundary_X-2, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;
    %Right
    E_X(Cavity2_Upper_Boundary_X+2:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;
    E_Y(Cavity2_Upper_Boundary_X+2:Front_Upper_Boundary_X, Front_Lower_Boundary_Y:Front_Upper_Boundary_Y,
Cavity2_Upper_Boundary_Z)=0;


    %TO DO metal outside
    %Outside top cavity 2
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;
    E_Y(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;
    E_Z(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Upper_Boundary_Y+2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;

    %Outside bottom cavity 2
    E_X(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Lower_Boundary_Y-2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;
    E_Y(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Lower_Boundary_Y-2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;
    E_Z(Cavity2_Lower_Boundary_X:Cavity2_Upper_Boundary_X, Cavity2_Lower_Boundary_Y-2,
Cavity2_Lower_Boundary_Z:Cavity2_Upper_Boundary_Z-1) = 0;



    end
    function Ground_Plane_Boundary                  % Simple ground plane under monopole
     E_X(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Monopole_Y_Coord-1, Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z+1)=0;
     E_Z(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Monopole_Y_Coord-1, Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z+1)=0;

     E_X(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Monopole_Y_Coord-2, Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z+1)=0;
     E_Y(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Monopole_Y_Coord-2, Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z+1)=0;
     E_Z(Cavity2_Lower_Boundary_X-1:Cavity2_Upper_Boundary_X+1, Monopole_Y_Coord-2, Cavity1_Lower_Boundary_Z-
1:Cavity1_Upper_Boundary_Z+1)=0;
     end
%% Draw functions
%-----------------------------------------------------------------------------------------------------------
function DrawSim1;   %draw horizontal cross section
    figure(1+PlotNumber);
    test = squeeze(E_X(:,Centre_Y,:));
    contour3(test,60);

    figure(2+PlotNumber);
    test = squeeze(E_Y(:,Centre_Y,:));
    contour3(test,60);

    figure(3+PlotNumber);
    test = squeeze(E_Z(:,Centre_Y,:));
    contour3(test,60);
```

```
    if Cavity2_Upper_Boundary_Z+2 < CellNumZ    %debug smaller outside dimensions means cavity2 not in sim space
    figure(4+PlotNumber)
    test2 = squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+2));
    view(-91,90);
    contour3(test2,60);

    figure(5+PlotNumber)
    test2 = squeeze(E_Y(:,:,Cavity2_Upper_Boundary_Z+2));
    contour3(test2,60);
    end

    drawnow;


end
function DrawSim2
    drawnow;
end
function DrawSim3

    drawnow;
end
function DoFinalPlots1;     % Plot several depths of skin penetration, Z scale set to 1st plot
 if Cavity2_Upper_Boundary_Z+2 < CellNumZ    %debug smaller outside dimensions means cavity2 not in sim space
    scrsz = get(0,'ScreenSize');
    hh=figure(8+PlotNumber);
    set(hh,'OuterPosition',[200 80 scrsz(3)/1.8 scrsz(4)/1.1],'Name','Intensity at various skin depths');
    Az=35,El=15;
    MAmp=0.05;

    subplot(3,2,1);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+1))',60);
    view(Az,El);
    drawnow;
    test=zlim;
    title('Intensity at Skin Surface')

    subplot(3,2,2);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+2))',60);
     view(Az,El);
    zlim(test);
    title('Intensity at Skin Surface+1')

    subplot(3,2,3);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+3))',60);
     view(Az,El);
    zlim(test);
    title('Intensity at Skin Surface+2')

    subplot(3,2,4);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+4))',60);
    view(Az,El);
    zlim(test);
    title('Intensity at Skin Surface+3')

    subplot(3,2,5);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+5))',60);
    view(Az,El);
    zlim(test);
    title('Intensity at Skin Surface+4')

    subplot(3,2,6);
    contour3(squeeze(E2(:,:,Cavity2_Upper_Boundary_Z+6))',60);
    view(Az,El);
    zlim(test);
    title('Intensity at Skin Surface+5')

 end


%String_Sim_Params;                          % make strings of parameters
%annotation('textbox','Position',[0 0.83 0.17 0.16], 'LineStyle', 'none',
'FitBoxToText','on','fontweight','b','String',char(Str1,Str2,Str3,'Ferrite Surface'));

%annotation('textbox','Position',[0 0.83 0.17 0.16], 'LineStyle', 'none',
'FitBoxToText','on','fontweight','b','String',char(Str1,Str2,Str3,'PML - 2'));
end

%-------------------------------------------------------------
%Misc functions
function [ Impedance ] = Dipole_Params( Freq, Length, Radius )
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here
Z0 = sqrt(mu_0/e_0);
EulerN = 0.57721;
Lambda = c/Freq;
WaveNum = 2*pi/Lambda;
WL = WaveNum * Length;

%From wikipedia 'Dipole Antenna' general formulae

R_Rad = (Z0/(2 * pi * (sin(WL/2))^2 )) * ...
        (( EulerN + log(WL) - cosint(WL) ) + ...
           (0.5 * sin(WL) * ( sinint(2*WL) - (2*sinint(WL)) ) ) +...
         ( (0.5 * cos(WL) ) * (EulerN + log(WL/2) + cosint(2*WL) - (2*cosint(WL)) ) )...
        );


X_Rad = (Z0/(4 * pi * (sin(WL/2))^2 )) * ...
        (( 2 * sinint(WL)  ) ...
         + ( cos(WL) * ( (2*sinint(WL)) - (sinint(2*WL)) ) ) ...
         - ( sin(WL)* ( (2*cosint(WL)) - (cosint(2*WL)) - cosint( (2*WaveNum*(Radius^2))/Length)) ) ...
        );
```

```
Impedance = R_Rad + i*X_Rad
end
%-------------------------------------------------------------

end %simulation
```

Appendix 3A Thermal Feedback Control Program

Main Program Heimann1.m

Disclaimer: This code has been created for the control of power based on thermal feedback only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the control of power based on thermal feedback only  and has not been verified
as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
% PH 2014
% Program to set up and read Heimann 8X8 sensor, Log temperature, and
% control on/off microwave power
fclose(s1);

clear all;
close all;
 global S8X  A B C D E F t FrameStack TempBuf Recs Debug1;
 global Ticks MaxTicks;
 global WinRowStart WinRowEnd WinColStart WinColEnd TempAverage TempNow WinPixels
 global MaxTemp MinTemp Power
 global TimerState
 global StartTime HeatTime CoolTime
 global Retries MaxRetries PowerBuf
 global s1

 StartTime = 5; HeatTime = 35; CoolTime = 5;
 MaxTemp = 60;          %Dont want more than this
 MinTemp = 59;          %Want more than this
 Power = 0;             %Current power state 0=off
 A = zeros(144);
 Debug1=0;
 Retries = 0;MaxRetries = 5;
F=0;
FrameStack = 32;
Ticks=1;
MaxTicks = StartTime+HeatTime+CoolTime;
TempBuf=zeros(MaxTicks,8,8);
PowerBuf=zeros(MaxTicks,1);
Recs = 1;
WinRowStart= 5; WinRowEnd=6; WinColStart=3; WinColEnd = 7;
WinPixels = (WinRowEnd-WinRowStart+1) * (WinColEnd-WinColStart+1);
TempAverage = zeros(MaxTicks,1);

HeimannUtils;
```

Support functions HeimanUtils.m

Disclaimer: This code has been created for the control of power based on thermal feedback only and has not been verified as to its correct basis, operation, results or fitness for any purpose. Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

```
% DISCLAIMER: This code has been created for the control of power based on thermal feedback only  and has not been verified
as to its correct basis, operation, results or fitness for any purpose.
% Any person or third party user of the code does so entirely at their own risk. No warranty is given and no liability is
assumed for direct or consequential loss however caused.
%
function [ ] = HeimannUtils( )
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here
% dummy function
% PH 2014
 global S8X A B C D E F t FrameStack TempBuf;
 global Ticks MaxTicks;
 global MaxTemp MinTemp Power
global TimerState
global s1
s1=serial('COM7');
fopen(s1); %turn it off

 PowerOff;
 S8X = udp('192.168.1.130',30444,'Timeout',5,'InputBufferSize',1440) %make udp object

S8X.ErrorFcn=@Errorcode;
S8X.BytesAvailableFcnCount=144;
%S8X.BytesAvailableFcn=@IntRead;

fopen(S8X);
out=query(S8X,'Calling HTPA series devices')
out=query(S8X,'x Release HTPA series device')
out=query(S8X,'Bind HTPA series device')

S8X.DatagramReceivedFcn=@IntRead;


B=zeros(1,64);
D=zeros(8,8);


if 1==0
for ii=1:10
    D = ReadTemp(S8X);
        figure(1);
        image(D);
        drawnow;


        figure(2);
        hold on;
        plot(E(1,:));
        drawnow;

        figure(3);
        hold on;
        plot(E(2,:));
        hold off;
       % colorbar;
        drawnow;
end
end

TimerState=0
t = timer('TimerFcn',@TimerTick,'ExecutionMode', 'fixedRate', 'Period', 1.0);

start(t);
fwrite(S8X,'K');


end


function [ Temp_Array ] = ReadTemp( UdpObj )
%UNTITLED2 Summary of this function goes here
%   Detailed explanation goes here
 global S8X  A B C D E;


 if S8X.BytesAvailable ~= 0
     flushinput(S8X);
 end

 D=zeros(8,8);

 for ii=1:2
 %
 end

 for ii=1:10
 fwrite(S8X,'k');
```

```
 A = fread(S8X,144,'uint8');
 zz=size(A)

 B(1:64)=A(1:2:127)+ (256*A(2:2:128));
 B = (B/10)-273;

  C=reshape(B,8,8);
  D=D+C';
 end

 D=D/32

 %S8X.BytesAvailable

 Temp_Array = D;

 E=D(4:5,3:7)

end


function IntRead(Udp,Event)
global S8X  A B C D E F FrameStack TempBuf Recs Debug1;
global WinRowStart WinRowEnd WinColStart WinColEnd TempAverage TempNow WinPixels
global MaxTemp MinTemp Power
global Ticks MaxTicks;
global Retries MaxRetries PowerBuf

if strcmp(S8X.status, 'open')

A = fread(S8X,144,'uint8');
% zzz=size(A)

if size(A,1)==144
    if F==0
    D=zeros(8,8);
    end
    if F<FrameStack
        B(1:64)=((A(1:2:127)+ (256*A(2:2:128)))/10)-273;
        %B = (B/10)-273;
        C=reshape(B,8,8);
        D=D+C';
        F=F+1;
    end
  if F==FrameStack   %Temp Frame available
    F=F+1
    TempBuf(Ticks,:,:) = D/FrameStack;
    TempNow = sum(sum(TempBuf(Ticks,WinRowStart:WinRowEnd,WinColStart:WinColEnd)))/WinPixels
    TempAverage(Ticks) = TempNow;
    g(:,:)=TempBuf(Ticks,:,:); image(g);  drawnow;
    Recs=Recs+1;

  end   %of F== framestack
    if F>FrameStack
    %Debug1=Debug1+1
    end

else    %Else size not temp buff
    char(A)
    zzz=size(A)
    beep;
  end

end %of if open

end %of function
%-------------------------------------------------------------------------------
function TimerTick(t,Event)

global S8X A B C D E F FrameStack Ticks MaxTicks TempBuf Recs;
global WinRowStart WinRowEnd WinColStart WinColEnd TempAverage TempNow WinPixels
global MaxTemp MinTemp Power
global TimerState
global Retries MaxRetries PowerBuf
global StartTime HeatTime CoolTime

if F >= FrameStack    %Means a temp frame is available
 Retries = 0;
else Retries = Retries+1
   if Retries >= MaxRetries   %error
      TimerState =  4;
   end

end

switch TimerState
    %--------------------------
    case 0     %Start up
        PowerOff;
        if Ticks >= StartTime
            TimerState = 1;
        end
   %--------------------------
    case 1     %Heating
        if Ticks > StartTime+HeatTime
            PowerOff;
            TimerState = 2;
        else
           if TempNow <=MinTemp    %if too cold turn it on
            PowerOn
           elseif TempNow >=MaxTemp %if too hot turn it off
            PowerOff
           else                     %else refresh power setting, midband gives hysteresis
            if Power==0
```

```
                PowerOff;
            else
                PowerOn;
            end
           end

        end    %of if Ticks>

   %---------------------------
    case 2     %cooling and delay
      PowerOff;
        if Ticks > StartTime+HeatTime+CoolTime
            TimerState = 10;

        else


        end
    %-------------------------
    case 3     %shutdown and switch off not used
    PowerOff;
    TimerState = 10;
    %-------------------------
    case 4      % Error state
    disp 'Aborting - no temperature data received'
    PowerOff
    fwrite(S8X,'X');
    TimerState=10;
    %------------------------
    case 10     %stop and clean up
    stop(t)
    fwrite(S8X,'X');
    fwrite(S8X,'X');
    fwrite(S8X,'X');
    fwrite(S8X,'X');
    PowerOff;

    figure;
    hold on
    plot(TempAverage);
    plot((PowerBuf*2.5)+10,'r','LineWidth',2);
    axis([1 MaxTicks 10 80]);
    grid on;
    drawnow;
    hold off

    figure;
    mesh(squeeze(TempBuf(:,WinRowStart,WinColStart:WinColEnd)));
    axis([1 5 1 MaxTicks 10 80])
    drawnow;

    figure;
    mesh(squeeze(TempBuf(:,WinRowEnd,WinColStart:WinColEnd)));
    axis([1 5 1 MaxTicks 10 80])
    drawnow;
    %-----------------------
    otherwise  %error really
    TimerState = 10;
    PowerOff;
end  %of switch

%----------------------------------------
PowerBuf(Ticks)=Power;  %Record power state
Ticks=Ticks+1;
if Ticks~=MaxTicks
 F=0;   %Restart temp acquisition
else
 %Ticks
end


end


function Errorcode(Udp,Event)
udp
Event
end



function [] = PowerOn()
global MaxTemp MinTemp Power s1
disp 'ON'
Power = 1;
s1.DataTerminalReady = 'off';
end

function [] = PowerOff()
global MaxTemp MinTemp Power s1
disp 'OFF'
Power = 0;
s1.DataTerminalReady = 'on';
end
```

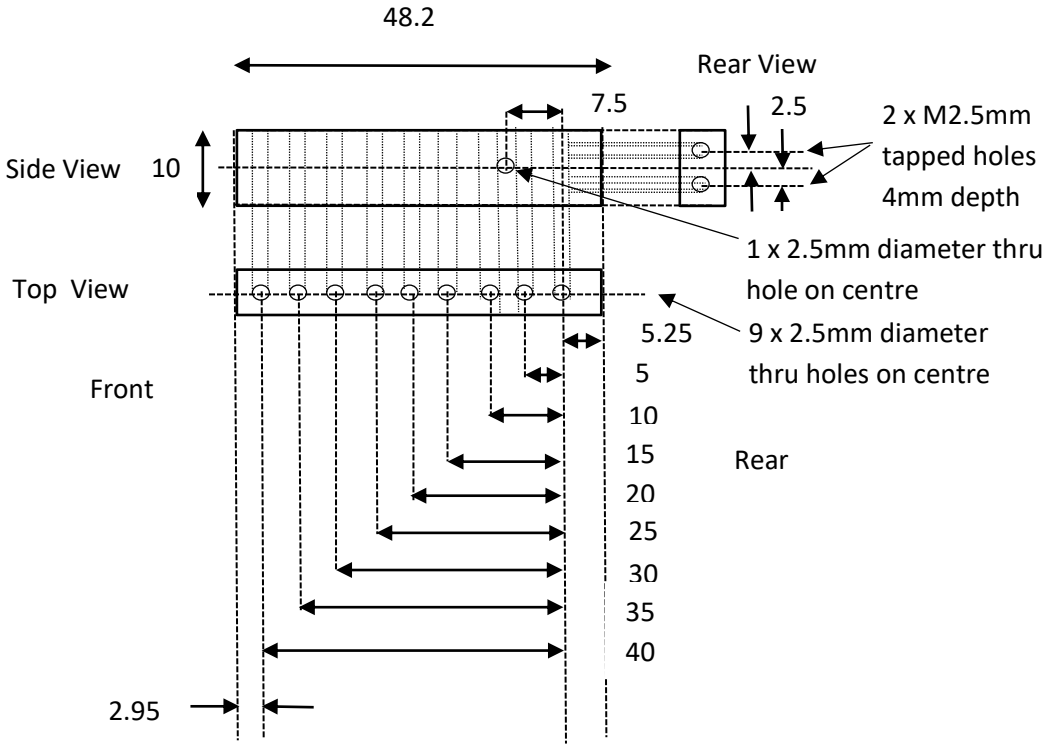
Appendix 3B New Applicator Component parts

Disclaimer: The design detailed here is included for documentation purposes only and has not been verified as to its correct basis, operation, results, safety, spurious or other emissions or fitness for any purpose. Any person or third party user of the design does so entirely at their own risk. No warranty is given and no liability is assumed for direct or consequential loss however caused.

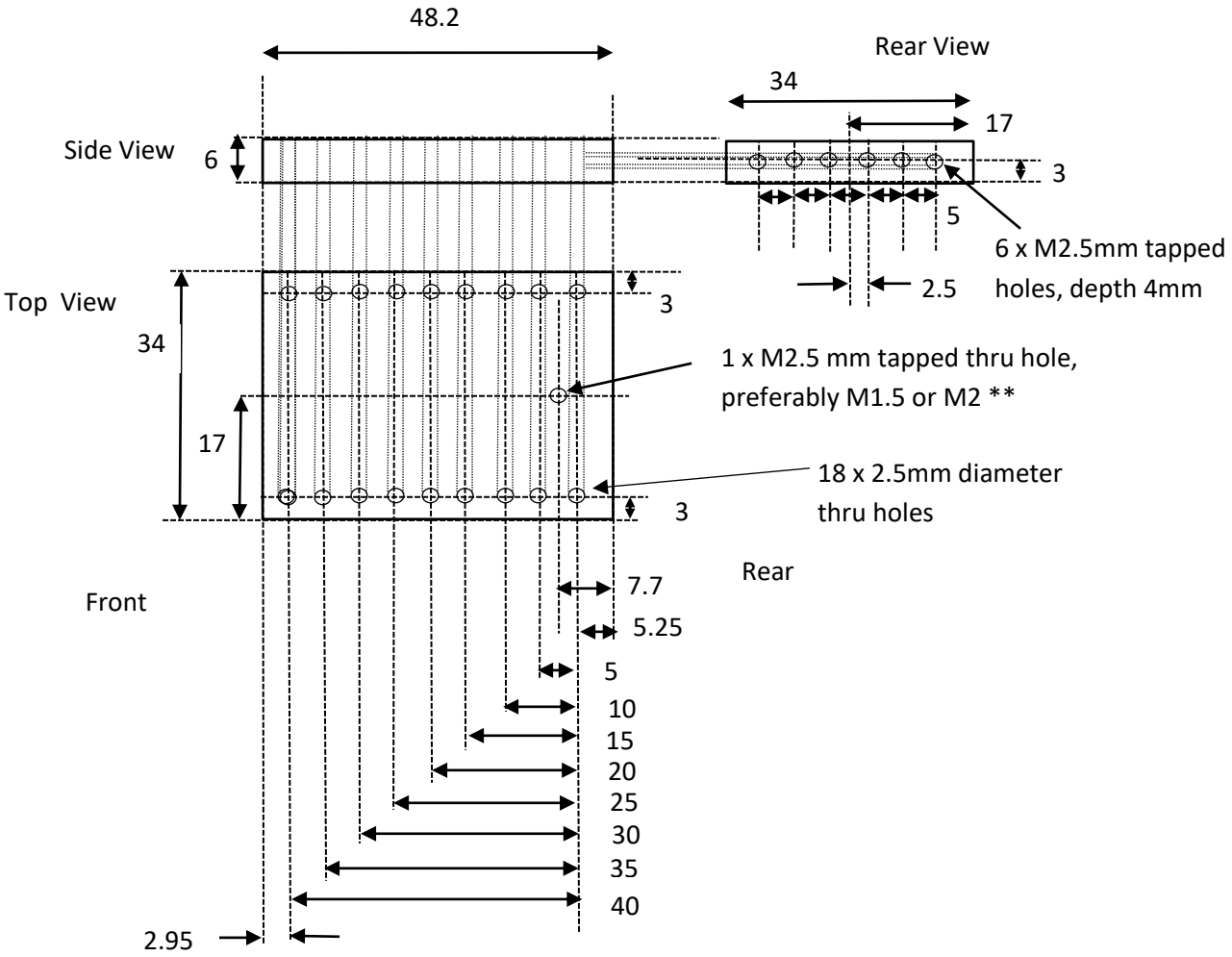
Skin Applicator #1.1v (Based on Simulation #4.1) 28/3/2014

All Dimensions in mm

Side Pieces Qty – 2, Material 6mm Aluminium

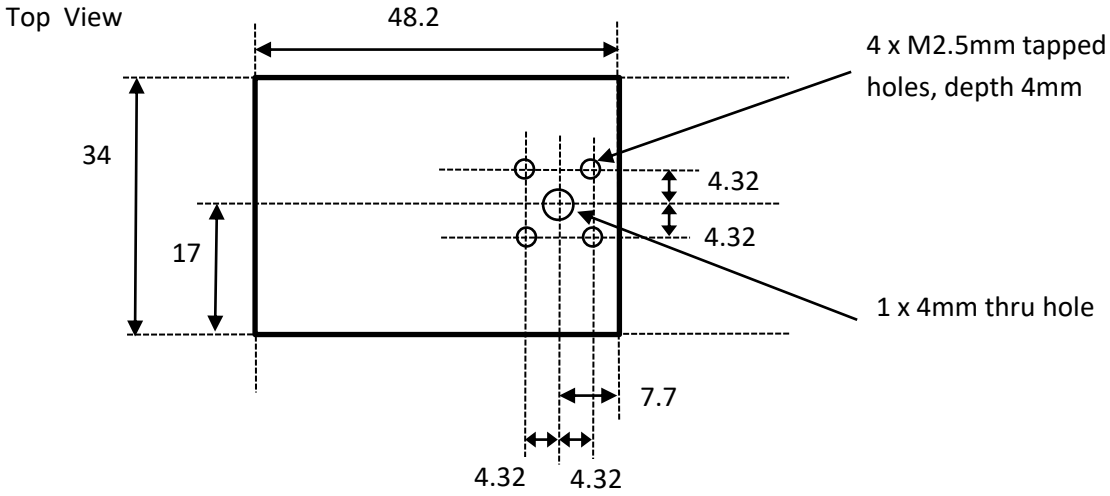


Top Piece Qty – 1 , Material 6mm aluminium

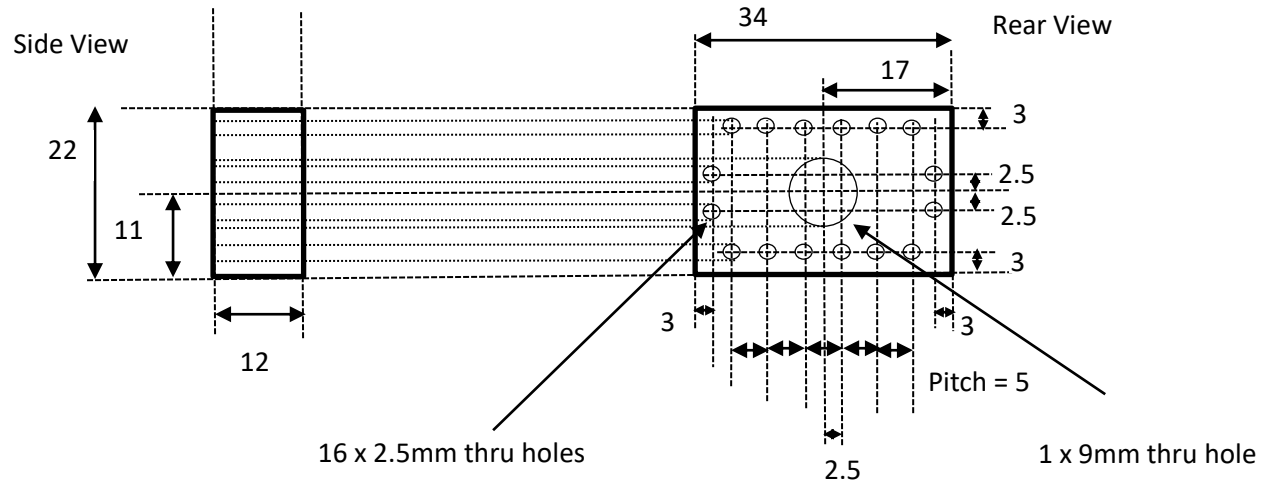


Bottom Piece Qty – 1, material 6mm aluminium

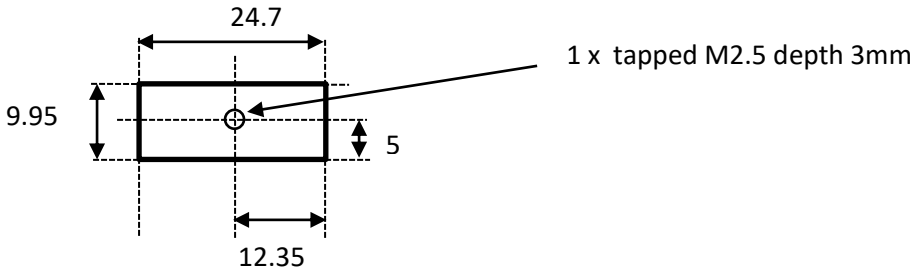
As per top piece but with additional holes and removal of tapped M2.5 (or M2/M1.5) ** hole



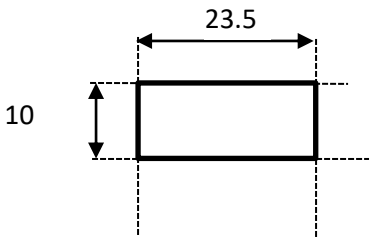
End Piece Qty – 1, material aluminium



Inside Piece type 1 Qty-2 Material: Brass or Aluminium, Thickness 4.1mm



Inside Piece type 2 Qty-2 Material: Teflon, Thickness 4.2mm



Inside Piece type 2 Qty-2 Material: Brass or aluminium rod, radius 2mm