5-31-1992

Analysis of electric field on surface of an atomically sharp field emitter

Ziyan Cheng
New Jersey Institute of Technology

ABSTRACT
Analysis of
Electric Field on Surface of
an Atomically Sharp Field Emitter

by
Ziyan Cheng

In this thesis, exact calculation and computer simulation of an atomically sharp emitter are presented. In the theoretical analysis, the cathode-anode or cathode-gate complex is modeled for cone- and wedge-shaped emitters so that exact solution can be obtained. The exact solution reveals that the field on the emitter surface depends not only on the sharpness of the emitter, but also on the global geometry of the field emitter, gate, and anode complex. This finding is contrary to a widely accepted concept that field is predominantly determined only by local curvature of the tip.

The theoretical results are confirmed by computer simulation of the field distribution of realistic tip-gate-anode structures.. In our study the emitter is atomically sharp, with a radius as small as $1nm$, while the emitter-gate distance, the gate-anode distance, and the supporting structure of the emitter tip, are in the range of $2000 - 5000nm$.

To accommodate more than three orders of difference between the dimension of the tip area and that of the tip support height and tip-gate distance, the ANSYS Software which uses *finite element method* has variable mesh sizes. Our finite element simulation results agree with other simulation results published earlier in this field[4]. Since the mesh size can be as small as $1nm$ and by using virtual memory the element number can be as big as 4000, it is expected that our result is more accurate than the calculation presented in reference[4].

As a result of our calculation it was possible to give several design guidelines for the selection of emitter parameters such as the tip radius, the emitter-gate distance or gate diameter, and the emitter shaft's half angle. Furthermore, the effect of mesh

size on our simulation results are discussed.

Analysis of Electric Field on Surface of an Atomically Sharp Field Emitter

By

Ziyan Cheng

A Thesis
Submitted to the Faculty of New Jersey
Institute of Technology in
Partial Fulfillment of the Requirements for the Degree of
Master of Science
Department of Electrical and
Computer Engineering
May 1992

APPROVAL PAGE

Analysis of
Electric Field on Surface of
an Atomically Sharp Field Emitter

by
Ziyan Cheng

May 8/92

Dr. Ken K. Chin, Thesis Advisor
Associate Professor of Department of
Physics, NJIT

May 8, 1992

Dr. Roy Cornely, Committee Member
Professor of Department of Electrical and Computer
Engineering, NJIT

5/8/92

Dr. Durga Misra, Committee Member
Associate Professor of Department of Electrical and Computer
Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Ziyan Cheng

Degree: Master of Science in Electrical and Computer Engineering

Date: May 1992

Date of Birth:

Place of Birth:

Undergraduate and Graduate Education:

- Master of Science in Electrical and Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1992
- Master of Science in Electronic Science and Technology,
  East China Normal University, Shanghai, P.R.C., 1988
- Bachelor of Science in Electronic Science and Technology,
  East China Normal University, Shanghai, P.R.C., 1985

This thesis is dedicated to
Xu Cheng

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
Introduction

Since the first international conference on vacuum microelectronics, there has been renewed interest in cold cathode field emitter tips working as efficient electron sources. This interest was due to the advances in three dimensional semiconductor fabrication technology which enable tips with submicron radii to be made. These semiconductor electron emitters can be used as cathodes in micron-sized vacuum diodes and triodes mass produced within a piece of semiconductor material. These emitter tips can also be used as cathodes in devices aimed at glow discharge and electroluminesent display applications. Moreover, these semiconductor devices have several significant advantages over traditional ones, particularly as regards temperature and irradiation insensitivity.

The fundamental questions regarding the characteristic of emitters that device application engineers may ask are:

(a) How does emission depend on electric field intensity over the surface of the semiconductor tip? (b) What is the dependence of this electric field distribution on the geometry of the tip and the surrounding electrodes? (c) How are the emission characteristics improved by metallic emitter coatup? (d) Do phenomena such as ion bombardment heating effects destroy the emitter and are there any ways to protect against these effects by coating the tip? (e) What are the limits on current density?

This thesis is focused on the basic question concerning the effect of the shape of an emitter tip on the electric field distribution. The emitter geometries are guided by limits imposed by capabilities of various processing methods.

Various advanced structures have been suggested for vacuum microelectronic devices. Two basic structures have been proposed: one, a novel cathode anode structure; the other, a *"gated structure"*. A schematic diagram of the cathode anode structure and a thin film emission cathode (i.e. a "gated structure" is shown

1

in Fig 1.1 (below).
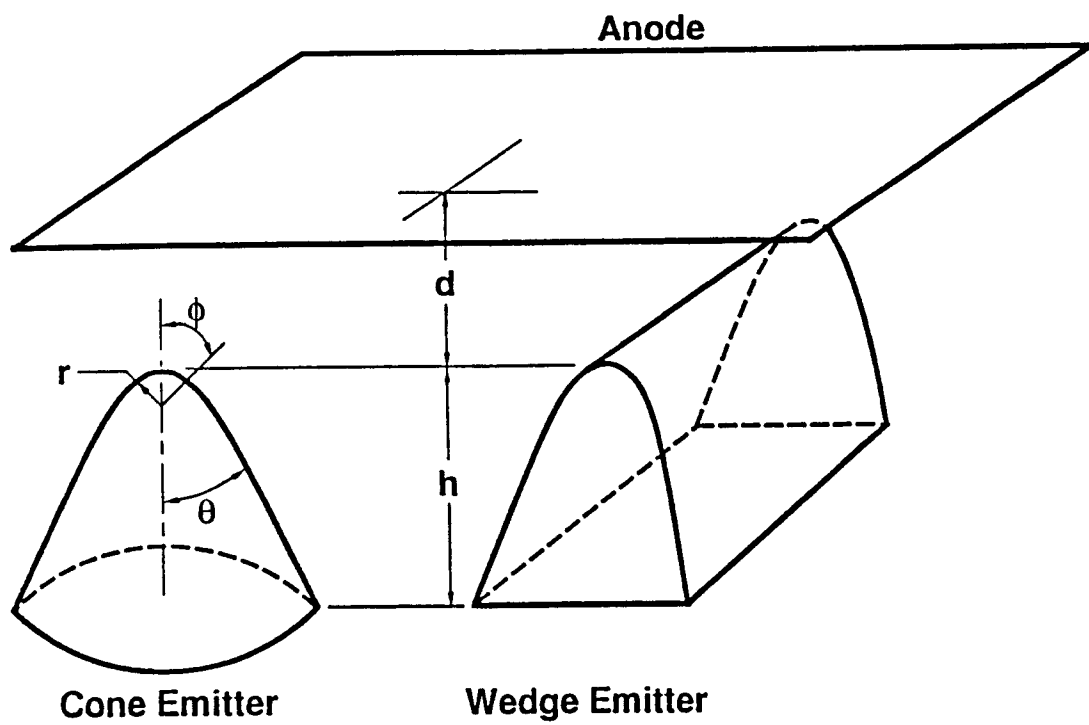
Fig. 1.1 (a) Schematic diagram of the cathode anode structure
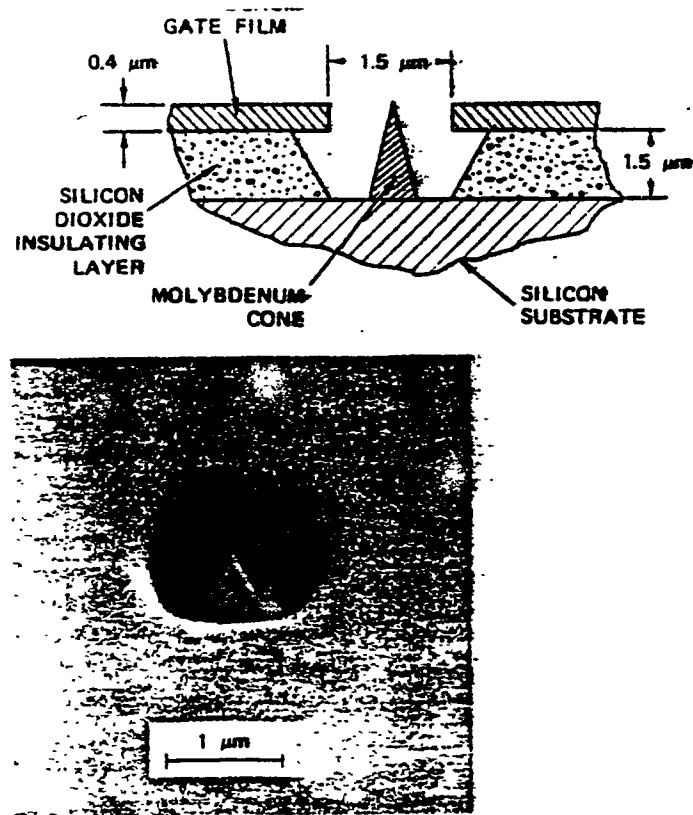
Fig.1.1 (b) Schematic diagram and scanning electron micrograph
of the gated field emission cathode

The emitter cathodes which previously have been proposed can be modeled for analysis of the field emitter as tiny spheres on either conical shanks(for the cone emitters) or on cylindrical shanks(for the wedge emitters).

Conical emitters are made by deposition through a tiny hole, made by isotropic etching and ion milling, while wedge-shaped emitters are made by deposition past a closely-shaped mask edge, by selective oxidation, or by selective epitaxial growth [1] [2] [3].

Several studies have been performed to date on optimizing the geometry of the cathode-anode structure. R. B. Marcus and K. K. Chin[4] performed a computer

simulation of the "cathode-anode" structure through use of the software package Simion; however, the software can handle but narrow range of geometrical differentiation of emitter.

W. P. Dyke and W. W. Dolan[5] did a theoretical study of field emission as a function of tip radius r, and as well, as a function of steepness of the emitter sidewalls. This analysis was valid only for a certain kind of cathode-anode structure. A general formula of field intensity with respect to the average curvature of equipotential surfaces was derived by Luo Enze[6]. This formula when supplemented by data measured from experiment using an analogue technique can be used to calculate the field intensity on the tip of the field emitter.

The electric field distribution in the vicinity of the cathode tip with radii of the order of $50nm$, together with reduction in the field strength due to emitter support was calculated by Spindt[7] using a relaxation method, this result was wildly quoted and accepted in the literature. This emission cathode was fabricated using thin film techniques and electron beam microlithography.

In several studies on *gated structure*, it is proposed that instead of using an emitter material with low work function, higher emissions can be obtained from a given operating voltage by sharpening an emitter to a sharp point or edge[8]. The emitter-gate geometry should have the following characteristics in order to get a uniform field:

- The gate opening should be small[9] and concentric with tip.
- The gate elevation should be roughly at the same level of the tip height[10][11]

Based on the assumption above, exact calculations and computer simulations of the electrical field of an atomically sharp emitter were carried out in this thesis. As for theoretical analysis, the cathode-anode or cathode-gate complex was modeled for wedge-shaped emitters. Using a conformal transformation the exact solutions were obtained by separation of variable in the new coordinate system.

The exact solutions revealed that the field on the emitter surface depends not

only on the sharpness of the tip, but also on the global geometry of the field emitter-that is gate, and anode structures. This finding is contrary to a widely accepted conventional concept that the field is predominantly determined only by local curvature of the tip.

The theoretical results are confirmed by computer simulation of the field distribution of realistic tip-gate-anode structures. For purpose of simulation the cone-shaped emitter was approximated as part of a spherical surface, and the wedge-shaped emitter was approximated as part of a cylindrical surface as shown in Fig 3.1 (chapter three, below). Thus a wedge shaped emitter is curved on one plane and flat on another, while a cone shape tip is curved on both. In comparing the performance of the wedge and cone tips, we assume that the two emitters have the same base area. The simulations were performed on ANSYS revision 4.4, a software package which was developed by *SWANSON ANALYSIS SYSTEMS, INC.* In our study the emitter is atomically sharp, with a radius as small as $1nm$, while the emitter-gate distance, the gate-anode distance, and the supporting structure of the emitter tip , are in the range of $2000 - 5000nm$.

The reason for using ANSYS is that it has a very good user interface. By selecting its menu, we can create any model we need. The result of ANSYS can be shown either in graphic or tabular form. By using zoom capability, any detailed electric field distribution on device boundaries can be obtained. This is particularly important when the radius of the cathode is very small compared to other dimensions such as emitter-gate distance, the gate-anode distance.

The ANSYS software package allows for variation of the mesh sizes used in the finite element analysis approach. This work used the student version of ANSYS 4.4 and mesh size of $1nm$ was employed. By using virtual memory capability, the mesh element capability was 4000.

The results of the simulation were compared with other published results, particularly the widely quoted results of Spindt. and there was good agreement with

results from others, that were based on completely different analysis approaches. In this thesis, the geometry of the structure was investigated in more detail than previous work. The present calculations indicate the possibility of providing several design guidelines for the selection of emitter parameters, such as tip radius, emitter-gate distance or gate diameter, and emitter support half angle.

Chapter 2
Background for Computer Simulation Method

The finite element method is the numerical method which is used in our computer simulation. This chapter introduces the basic idea and procedures of finite element method and detailed implementation of finite element method by a software package ANSYS.

2.1 Introduction of Finite Element Method

The finite element method is an analysis approach that involves dividing the solution domain of a problem into small subregions or finite elements. The method was initially developed for the solution of physical problems in the structural mechnics; however it was soon recognized that the method could be applied also to the solution of many other types of problems. In this thesis, it is used to solve electrostatic boundary problem.

As shown in Fig 2.1, the area has been divided into small finite elements, Associated with each element are several nodal points at which adjacent elements are effectively linked together. and at which the values of the relevant varibles are to be determined. An analysis based on physical arguments such as voltage is used to derive a set of simultaneous algebraic equations for these nodal point values. The resulting finite element formulations can readily accommodate the voltage boundary condition.

The main advantage that the finite element method has over other analysis techniques is its large generality. Normally, as was pointed out, it seems possible, by using many elements, to virtually approximate any continum with complex boundary and loading conditions to such a degree that an accurate analysis can be carried out. In practice, however, obvious engineering limitations arise, a most important one being the cost of the analysis. As the number of elements used is
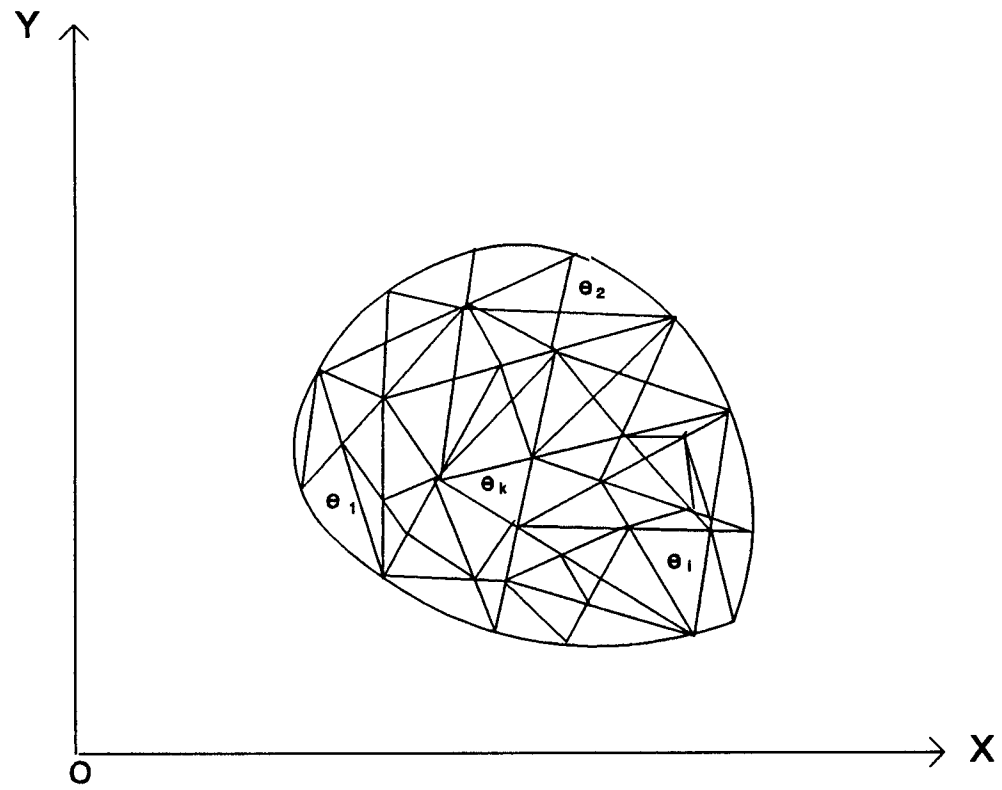
Fig 2.1 Region subdivided into I finite elements

increased, the manpower required to prepare the relevant data and to interpret the results increase; also a larger amount of computer time is needed for the analysis. Furthermore, the limitations of the program and computer may prevent the use of a large number of finite elements to idealize the continuum. The analysis process can be understood to consist of essentially three phases: a. Calculation of structure matrices K, M, C, and R whichever are apllicable; b. Solution of equilibrium equation; c. Evaluation of element.

2.2 Implementation of the finite element method.
2.2.1 The Electrostatic-Potential Distribution Problem.

Our problem is electro-static field problem, the solution of which usually consists in seeking a two- or three-dimensional function which satisfies an elliptic differential equation together with certain boundary conditions. In a two-dimensional problem, for a homongenous and isotropic medium, a function $U(x,y)$ is sought which satisfies Possion's equation in a given region G:

$$-\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) = f(x,y) \qquad (2.1a)$$

or

$$-\nabla^2 U = f \qquad (2.1b))$$

where $f(x,y)$ is a given two-dimensional function in G . On the boundary C of the region G, the function $U(x,y)$ must satisfy certain boundary conditions. In general the values of U are specified on the segment C1 of the boundary, so that $U(s) = \Phi(s)$ on C1 (2.1c), where s is the boundary length and $\Phi(s)$ is a given function. For the above problem, the corresponding related variational integral is[12]:

$$I(u) = \frac{1}{2} \int \int_G \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} - 2fU\right) dx dy \qquad (2.2a)$$

For f=0, $I(u)$ is given by:

$$I(u) = \frac{1}{2} \int \int_G \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} \right) dx\,dy \qquad (2.2b)$$

It can be seen that the solution $U(x,y)$ satisfying Eqs (2.1) is identical to that function which minimizes the variational integral Eq (2.2)[13].

Various procedures for solving equation (2.1) or (2.2) are presented in the literatures by various authors[14],[15],[16],[17], these procedures are summarized in the following paragraphs and presented as six setps for obtaining the solution to Eq. (2.1) or (2.2).

1) Discretizing the continuum. The problem is discretized by dividing the total space domain into simple subdomains: the elements. In a two-dimensional field problem, the basic region G is divided into triangles, parallelograms, curved-sided triangles or quadrilaterals. With only straight-sided elements, a fairly good approximation of the basic region can be achieved with an appropriately fine discretization. Curved elements obviously produce a better approximation of the basic region. This type of discretization is very flexible and is adjusted to suit the region of the problem.

For three-dimensional problems, the region is discretized into tetrahedral elements, cuboidal elements or elements with curved surfaces. As shown in fig 2.1, suppose the region is subdivided into l elements, the total number of nodes is denoted by n. The subdivision of the region and the required continuity continuity of the solution allow Eq. (2.2) to be written as

$$I = \sum_{\imath=1}^{l} I^{e\imath} \qquad (2.3)$$

where element contribution $I^{e\imath}$ is defined as

$$I^{e\imath} = \frac{1}{2} \int \int_{e\imath} \left((\frac{\partial U^{e\imath}}{\partial x})^2 + (\frac{\partial U^{e\imath}}{\partial y})^2 \right) dx\,dy \qquad (2.4)$$

2) Selecting interpolation functions. For each of the elements a suitable approximation to the function sought, or more generally, to the functions which describe

the problem, has to be chosen. For two-dimensional problems, linear, and higher -order polynomials of the quadratic forms

$$U(x,y) = C_1 + C_2 x + C_3 y$$

$$U(x,y) = C_1 + C_2 x + C_3 y + C_4 x^2 + C_5 xy + C_6 y^2$$

or bilinear approximation

$$U(x,y) = C_1 + C_2 x + C_3 y + C_4 xy$$

are appropriate.
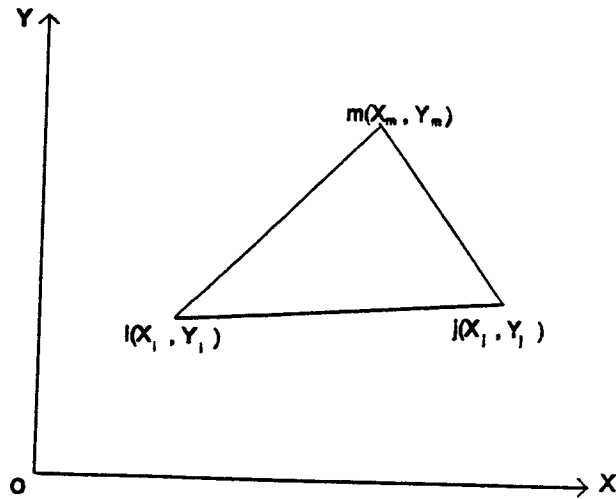
Fig 2.2 A typical triangular element e_1

This form of approximation depends on the shape of the element and also on the type of problem being treated. If we choose triangular elements, such as the element e_1 shown in Fig 2.2, the node identifiers i,j and m can be assigned to the nodes in a counterclockwise sense. For an arbitary element e_1,

$$U^{e_1}(x,y) = C_1 + C_2 x + C_3 y \qquad (2.5)$$

has been chosen for the three node of each element. An advantage of using triangular elements is that they can be made to fit any shape of the domain boundary, provided that the boundary can be represented with sufficient accuracy by a set of equations

$$U_i = c_1 + c_2 x_i + c_3 y_i \qquad (2.6a)$$

$$U_j = c_1 + c_2 x_j + c_3 y_j \qquad (2.6b)$$

$$U_m = c_1 + c_2 x_m + c_3 y_m \qquad (2.6c)$$

Where U_i, U_j, U_m are values of U at node i,j, and m, respectively, and superscripts e_i have been dropped for convienence. The constants c_1, c_2 and c_3 can be written by matrix theory as

$$c_1 = \frac{1}{2\triangle}(a_i U_i + a_j U_j + a_m U_m), \qquad (2.7a)$$

$$c_2 = \frac{1}{2\triangle}(b_i U_i + b_j U_j + b_m U_m), \qquad (2.7b)$$

$$c_3 = \frac{1}{2\triangle}(c_i U_i + c_j U_j + c_m U_m), \qquad (2.7c)$$

where

$$a_i = x_j y_m - x_m y_j, \quad b_i = y_j - y_m, \quad c_i = x_m - x_j \qquad (2.8)$$

and where $a_j, a_m, b_j, b_m, c_j, c_m$ can be obtained by cyclic permutation of the indices. The element area of the triangle $\triangle$ is

$$\triangle = \frac{1}{2}\begin{vmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_m & y_m \end{vmatrix} \neq 0$$

provided that the area of the triangle is not equal to zero. The shape function representation can now be written as

$$U(x,y) = \frac{1}{2\triangle}\left[(a_i + b_i x + c_i y)U_i + (a_j + b_j x + c_j y)U_j + (a_m + b_m x + c_m y)U_m\right] \qquad (2.9a)$$

or in a more concise form

$$U^{e_i} = N_i U_i + N_j U_j + N_m U_m = N^e U^e \qquad (2.9b)$$

where the shape function matrix is defined as

$$N^e = [\, N_i \quad N_j \quad N_m \,] \qquad (2.10a)$$

and the element nodal vector as

$$U^e = \begin{bmatrix} U_i \\ U_j \\ U_m \end{bmatrix} \qquad (2.10b)$$

T he required derivatives can be obtained from (Eq 2.9a) as

$$\frac{\partial U^{e_i}}{\partial x} = \frac{1}{2\Delta} \left[b_i U_i + b_j U_j + b_m U_m \right] \qquad (2.11a)$$

$$\frac{\partial U^{e_i}}{\partial y} = \frac{1}{2\Delta} \left[c_i U_i + c_j U_j + c_m U_m \right] \qquad (2.11b)$$

3) Finding the element properties. Once the finite element model has been established(that is ,once the elements and their interpolation functions have been selected), we are ready to determine the matrix equations expressing the properties of individual elements. We can use one of four approachs: the direct approach, the variational approach, the weighted residual approach, or the energy balance approach. Here we use the variational approach. Substitution of Eqs (2.11) into element contribution [Eq(2.4)] yields

$$I^{e_i} = \frac{1}{8\Delta^2} \int \int_{e_i} \left[\left(b_i U_i + b_j U_j + b_m U_m \right)^2 + \left(c_i U_i + c_j U_j + c_m U_m \right)^2 \right] dx\, dy \qquad (2.12)$$

since Eq. (2.12) is independent of the varibles x and y Eq (2.12) may be written as

$$I^{e_i} = \frac{1}{8\Delta} \left[\left(b_i U_i + b_j U_j + b_m U_m \right)^2 + \left(c_i U_i + c_j U_j + c_m U_m \right)^2 \right] \qquad (2.13)$$

substituting all these element contributions into Eq. (2.3) transforms the variational integral into a function of all nodal values U_1, U_2, U_n, that is

$$I = I(U_1, U_2, \cdots, U_n) \qquad (2.14)$$

Now the variational parameters U_i are to be chosen such that $I(U_1, U_2, \cdots, U_n)$ is to be minimum. Thus

$$\frac{\partial I}{\partial U_p} = 0 \qquad p = 1, 2, \ldots, n \qquad (2.15)$$

Substitution of Eq. (2.3) into Eq. (2.15) allows this equation to be written as

$$\frac{\partial I}{\partial U_p} = \sum_{i=1}^{l} \frac{\partial I^{e_i}}{\partial U_p} = 0 \qquad p = 1, 2, \ldots, n \qquad (2.16)$$

Thus if the node identifiers i, j, and m of element e_i refer to system node numbers p, q, and r, respectively, differentiation of Eq. (2.13) with respect to U_p yields

$$\frac{\partial I^{e_i}}{\partial U_p} = \frac{1}{4\Delta} \left[b_p \left(b_p U_p + b_q U_q + b_r U_r \right) + \left(c_p U_p + c_q U_q + c_r U_r \right) \right] \qquad (2.17)$$

The general form of element matrix equation is

$$\begin{bmatrix} \frac{\partial I^{e_i}}{\partial U_i} \\ \frac{\partial I^{e_i}}{\partial U_j} \\ \frac{\partial I^{e_i}}{\partial U_m} \end{bmatrix} = \begin{bmatrix} k_{ii}^e & k_{ij}^e & k_{im}^e \\ k_{ji}^e & k_{jj}^e & k_{jm}^e \\ k_{mi}^e & k_{mj}^e & k_{mm}^e \end{bmatrix} \begin{bmatrix} U_i \\ U_j \\ U_m \end{bmatrix} \qquad (2.18a)$$

This form can be expressed more concisely by

$$\frac{\partial I^{e_i}}{\partial U_e} = K^e T^e \qquad (2.18b)$$

where

$$K^e = \begin{bmatrix} k_{ii}^e & k_{ij}^e & k_{im}^e \\ k_{ji}^e & k_{jj}^e & k_{jm}^e \\ k_{mi}^e & k_{mj}^e & k_{mm}^e \end{bmatrix} \qquad (2.19a)$$

$$k_{xy} = \frac{1}{4\triangle}(b_x b_y + c_x c_y) and \qquad x = 1,2,3, y = 1,2,3. \qquad (2.19b)$$

where $b's$ & $c's$ are defined by Eqs(2.8) K is the stiffness matrix for element e and U_e is the element nodal vector defined in Eq. (2.10b) 4) Assembling The Element Properties to Obtain the System Equations. To find the properties of the overall system models by network of elements, the matrix equation of the entire solution region has been formed by combining the matrix expressions. The element stiffness Matrix for the system nodal vector can be written as

$$U = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_n \end{bmatrix} \qquad (2.20)$$

For each nodal parameter in the system, an equation similar to Eq. (2.20) can be obtained. Assembling these equations into a single matrix equation gives

$$\frac{\partial I}{\partial U} = \begin{bmatrix} \frac{\partial I}{\partial U_1} \\ \frac{\partial I}{\partial U_2} \\ \vdots \\ \frac{\partial I}{\partial U_n} \end{bmatrix} = KT = 0 \qquad (2.21)$$

5) Solving the system equations. The assembly process of the preceding step gives a set of simultaneous equations that we need to solve to obtain the unknown nodel values of the field variable.

Assuming that the nodal value (assuming voltage at the point)voltage from $U_{p+1} - U_n$ is known(Dirichlet boundary condition), the matrix expression of (2.21) can be rewritted as

$$\begin{bmatrix} K_{pp} & K_{pp'} \\ K_{p'p} & K_{p'p'} \end{bmatrix} \begin{bmatrix} U_1 \\ \vdots \\ U_p \\ U_{p+1} \\ \vdots \\ U_n \end{bmatrix} = 0 \qquad (2.22)$$

$$[K_{pp}] \begin{bmatrix} U_1 \\ \vdots \\ U_p \end{bmatrix} = -[K_{pp'}] \begin{bmatrix} U_{p+1} \\ \vdots \\ U_n \end{bmatrix} \qquad (2.23)$$

where $[K_{pp}]$ and $[K_{pp'}]$ are known from (2.19b), solve (2.23) the system solution $U_1, U_2, \cdots, U_p$ van be obtained.

6) Making additional computations if desired. Once we get system nodal voltage, any voltage within the element can be solved by linear interpolation (2.11a) other important parameter(e.g. electrical field density) can also be obtained by simple calculation.

2.3 Computer Approach Using ANSYS Software

The detail of the implementation of the finite element method with the software package called ANSYS is described.

2.3.1 General Description of the Software Package

The ANSYS[18] program is a large, sophisticated finite element analysis program used in a wide variety of fields. The ANSYS program is organized into a Begin Level and Processor Level that is connected to the begin level. The user must enter ANSYS via the Begin Level, and move in and out of that level to the different processors, in which most of the analysis is done. This is illustrated in Fig. 3.1. A typical ANSYS analysis involves three distinct steps:

(1) Preprocessing, where the user provides data such as the geometry, materials and boundary conditions to the program;

(2) Solution, where the program computes the solution using the finite element method; and

(3) Postprocessing, where the results of the analysis are reviewed by the user through graphics displays and tabular listing.

As a general purpose program, ANSYS can be used to solve different physical problems in heat transfer, mechanics and electromagnetic field. The notation is as follow: *static* (KAN=0), *buckling* (KAN =1), *modal* (KAN=2), *full harmonic re-*

sponse (KAN=3), *nonlinear transient dynamic* (KAN=4), *linear transient dynamic* (KAN=5), *reduced harmonic response* (KAN = 6), *substructure* (KAN=7), and *thermal* (KAN =-1) analysis.

In the version of ANSYS we have access to, there is no direct analysis to solve the steady state electric field problem. However, since both the steady-state heat transfer and steady-state electric field problem abide by the same Possion equation, the solution of static electric field problem can be solved by using the analogy between the thermal and electrostatic parameters since ANSYS calculate directly the solution of heat transfer problem. If we get one parameter in the thermal analysis, we know its counterpart of the other. The relationships between the electric field parameters and thermal parameters are as follows:

Temperature $\Longleftrightarrow$ Voltage

Thermal gradient $\Longleftrightarrow$ Electric field

Heat flow $\Longleftrightarrow$ Current

The differential equation that governs the phenomenon of heat conduction also governs other phenomena in many areas of physical sciences and engineering. Other applications include are shaft torsion, pressurized membranes, seepage through porous media, incompressible, irrotational motion of ideal fluids, and diffusion in addition to electrostatics.

2.3.2 Application of ANSYS to Laplace Equations

Although ANSYS is capable of both steady-state and transient analysis, for this thesis only the steady-state analysis capability was required. The steady-state thermal analysis solves the following equation:

$$[K][T] = [Q] \qquad (3.1)$$

where

[K] is the conductivity matrix,

[T] the nodal temperature vector, and

```
                          ┌──────────────────┐
                          │   Entry to ANSYS │
                          └──────────────────┘
                                   │
                                   ▼
┌────────────────────────────────────────────────────────────────────────┐
│                              Begin Level                                 │
└────────────────────────────────────────────────────────────────────────┘
      │             │             │             │              │
      ▼             ▼             ▼             ▼              ▼
┌──────────┐ ┌──────────┐ ┌──────────┐ ┌──────────┐  ┌──────────┐
│ General  │ │Time-Histroy│ │          │ │ General  │  │Time-Histroy│
│Preprocessor│ │Preprocessor│ │ Solution │ │Postprocessor│ │Postprocessor│
│ (PERP7)  │ │ (PREP6)  │ │          │ │ (POSTI)  │  │ (POST26) │
└──────────┘ └──────────┘ └──────────┘ └──────────┘  └──────────┘
```
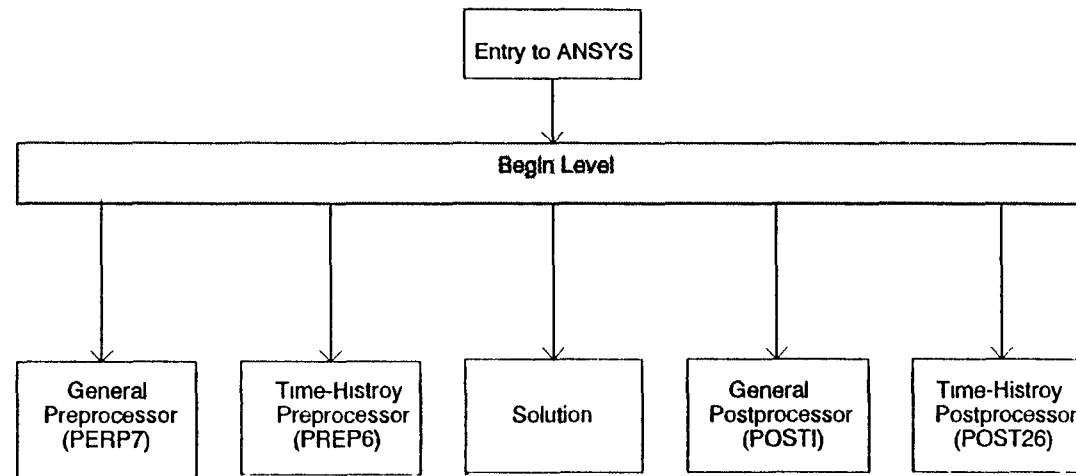
Fig 2.3   Relation of various ANSYS processor

[Q] the nodal heat flow rate vector.

The "conductivity" matrix [K] includes not only terms involving conduction, but also those involving convection and radiation. [Q] includes convection as well as internal heat generation rate terms.

The general form of Possion equation in the heat transfer domain is:

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial U}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial U}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z \frac{\partial U}{\partial z}\right) = C \qquad (3.3)$$

where $U(x, y, z)$ is the temperature variable to be determined; and K is a material property, with the subscripts representing the three orthogonal directions; and C is a constant. By substituting K_i with a constant, and C with 0, equation (3.3) is reduced to the Laplace equation of electrostatic potential problem as in (3.4)

$$\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} = 0 \qquad (3.4)$$

2.3.3 Details on Program Usage

An outline of the procedure used to obtain the simulation result in the chapter 3 is given below for steady-state thermal analysis.

Preprocessing: phase is where all relevant data, such as material properties, geometry and boundary condition, are entered into the database in preparation for solution.

1. Set Up: To define the analysis type, analysis options, element types, element geometric properties and material properties.

Analysis type (KAN,-1).

Analysis Options (KAY(3)) .

Element types (ET).

Geometric Properties (R, RMODIF. etc.).

Material properties (MP).

2. Model: To build the finite element (nodes and elements) either by solid modeling

or by direct generation.

> Solid Modeling (K, L, A, V, etc.). or
>
> Direct Generation (N, NGEN, E, EGEN. etc.).
>
> Reordering (WSORT, WAVES).

3. Load Data: To specify boundary conditions (constraints and loads) and load options.

> Boundary Conditions
>
> on the solid model (KNT, KHFLOW, LCVSF, etc.), or
>
> on nodes and elements (NT, HFLOW, CVSF, etc.).
>
> Load options (ITER, etc.)
>
> Load Steps (LWRITE, etc.).

4. Wrap Up: To write all appropriate information in a form suitable for the solution phase of the analysis. Write analysis file (AFWRITE).

Solution: The solution phase is the heart of a finite element program. In the solution phase, the solver is used to solve the basic equation of the analysis type and to compute the results. The solution phase requires little user interaction.

1. Divert output to a file if desired (/OUTPUT).

2. Submit analysis file (File27) for solution (/INPUT,27).

Postprocessing: The postprocessing phase is where you review the results of the analysis by obtaining graphics displays and tabular reports.

1. Read results data from the postdata file (File12) into the POST1 database (SET).

2. Perform desired operations – contour displays, results tabulation of post item such as etc.

> The analysis data flowchart is shown in Fig. 2.4
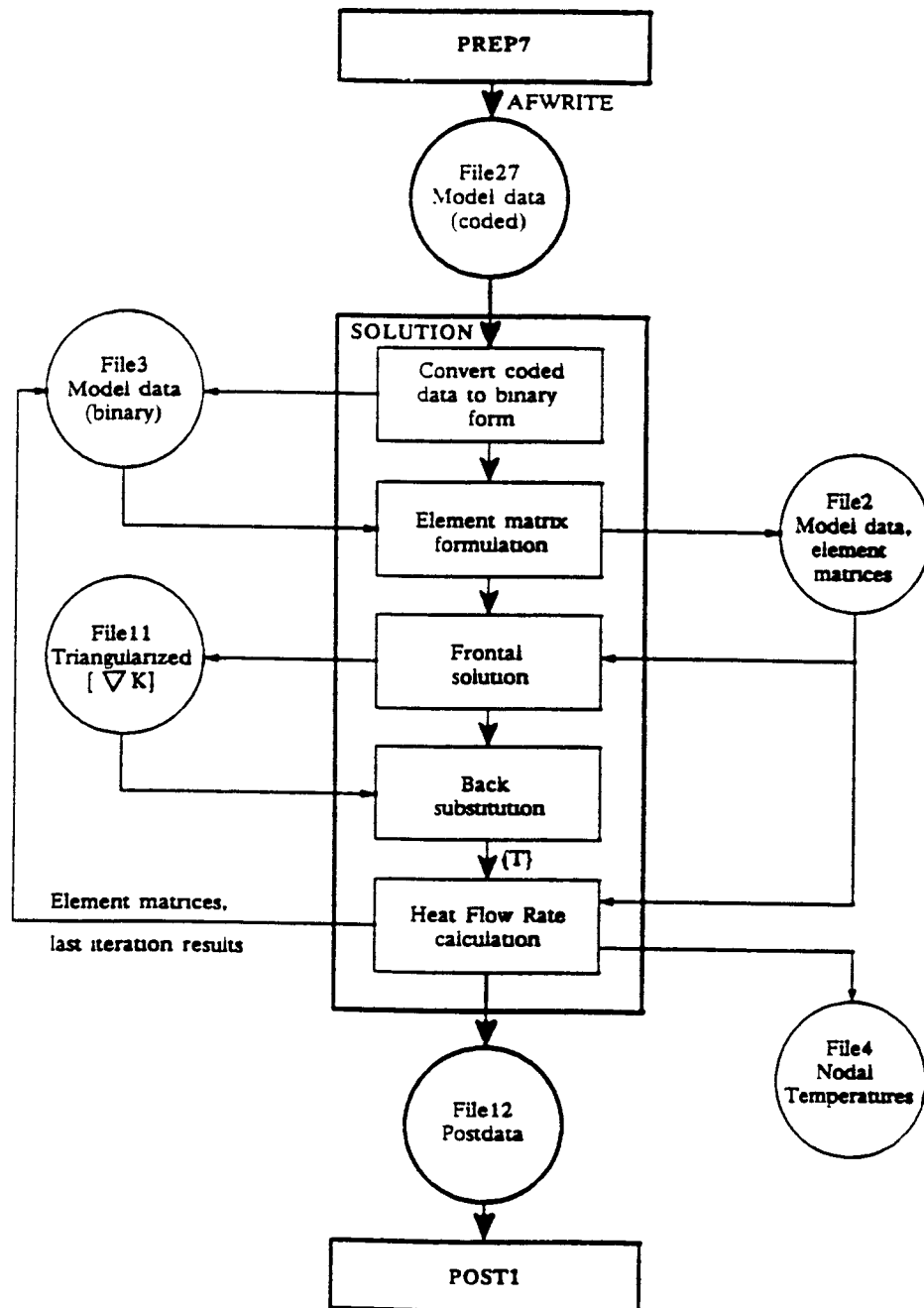
2.4 ANSYS Steady State Analysis Data Flow

Chapter 3
Analysis by Computer simulation method

In this chapter results obtained using numerical analysis to solve the emitter field problem are presented. The numerical analysis approach was based on applying the finite element method using ANSYS as described in Chapter 3. The ANSYS program is implemented on a Sun Sparc-I workstation. The two dimensional Laplace equation was used for analyzing wedge geometry, and three dimensional Laplace equations was used for analyzing the cone geometry, with appropriate boundary conditions for each case given as inputs for the computer analysis.

3.1 Device Geometry Specification

As shown in Fig 3.1, there are two cathode structures for a field emitter. One consists of a partial sphare representing a conical system, while the other consists of a partial cylinder , representing a wedge system. In both structures the emitter is supported. In the simulation, the ANSYS is applied to both structures.
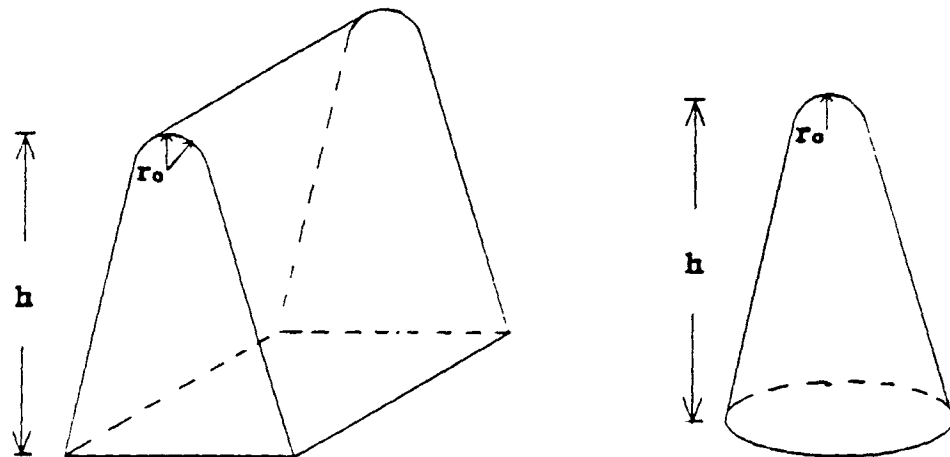
Fig. 3.1 Two basic cathode structures

3.2 Description of Geometry Model

To set up a model for simulation, let's first take a look at vertical triode with a cone cathode geometry, as shown in Fig 3.2.
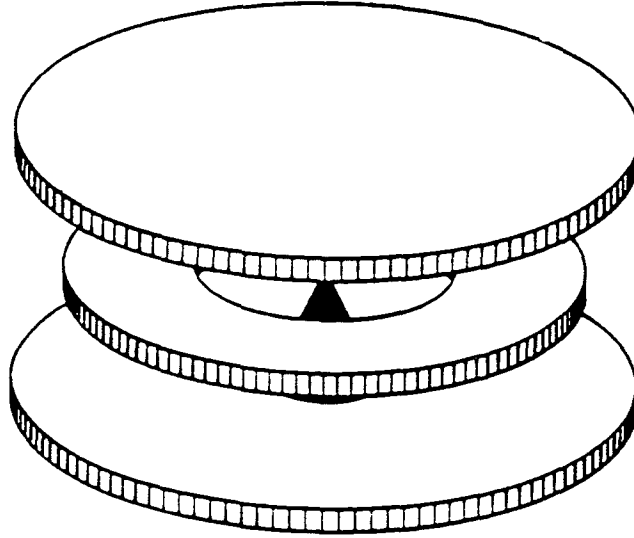
Fig. 3.2 Geometry of Vertical Triode with Cone cathode

Fig. 3.3 shows a model for geometric construction of the thin film field emission cathode proposed by Spindt. The main design parameters are: t, thickness of gate electrode; R, radius of hole in gate electrode; d, thickness of insulting layer; h, cone height; r, tip radius; and θ: the cone or wedge half angle. These parameters can be controlled during processing to various extents. In addition, the angle ϕ which represents the deviation from the normal at the emitting surface(with the origin at the tip) is also an important parameter in calculating the spatial distribution.

Fig 3.4 shows a geometry for the Spindt device with the geometries defined in a way that finite element method can be used. Since the geometry is symmetrical, only boundaries on the right side of symmetrical axis need to be considered. For the wedge geometry, plane symmetry is used. For the cone, axial symmetry is used.
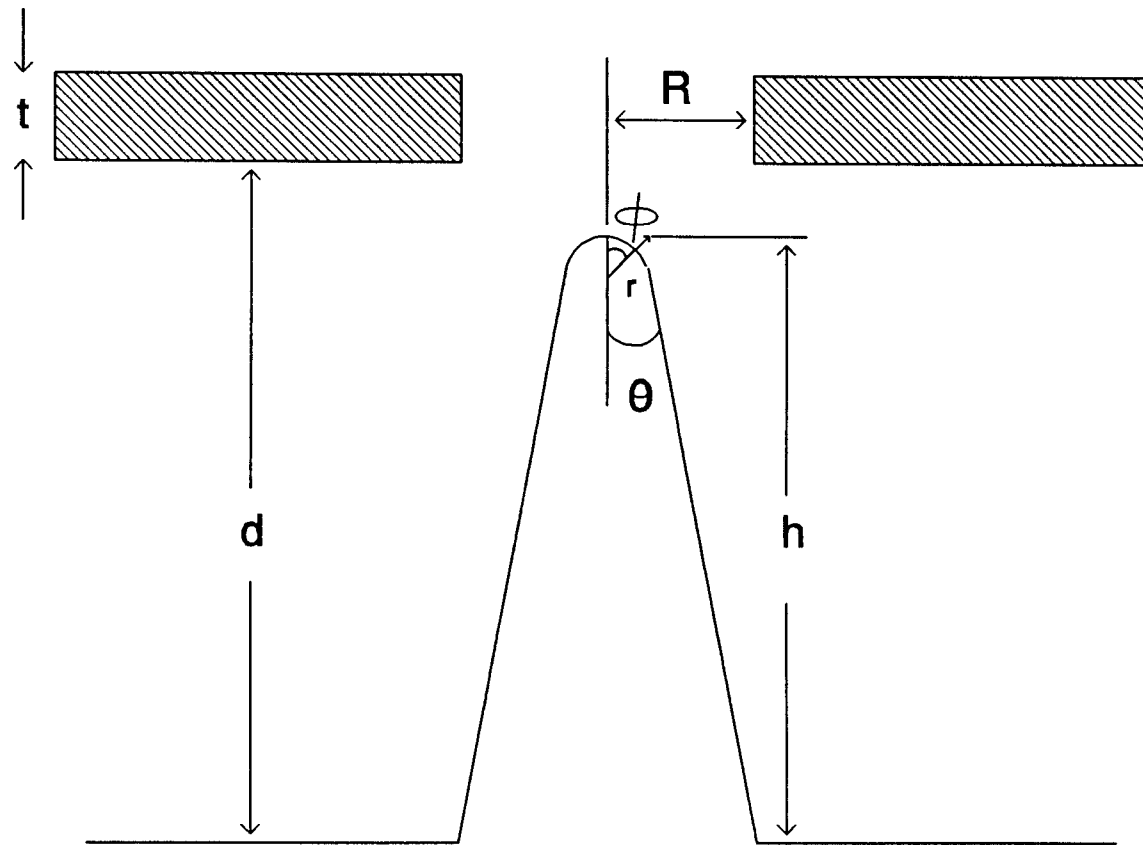
Fig 3.3   Emitter Simulation Model
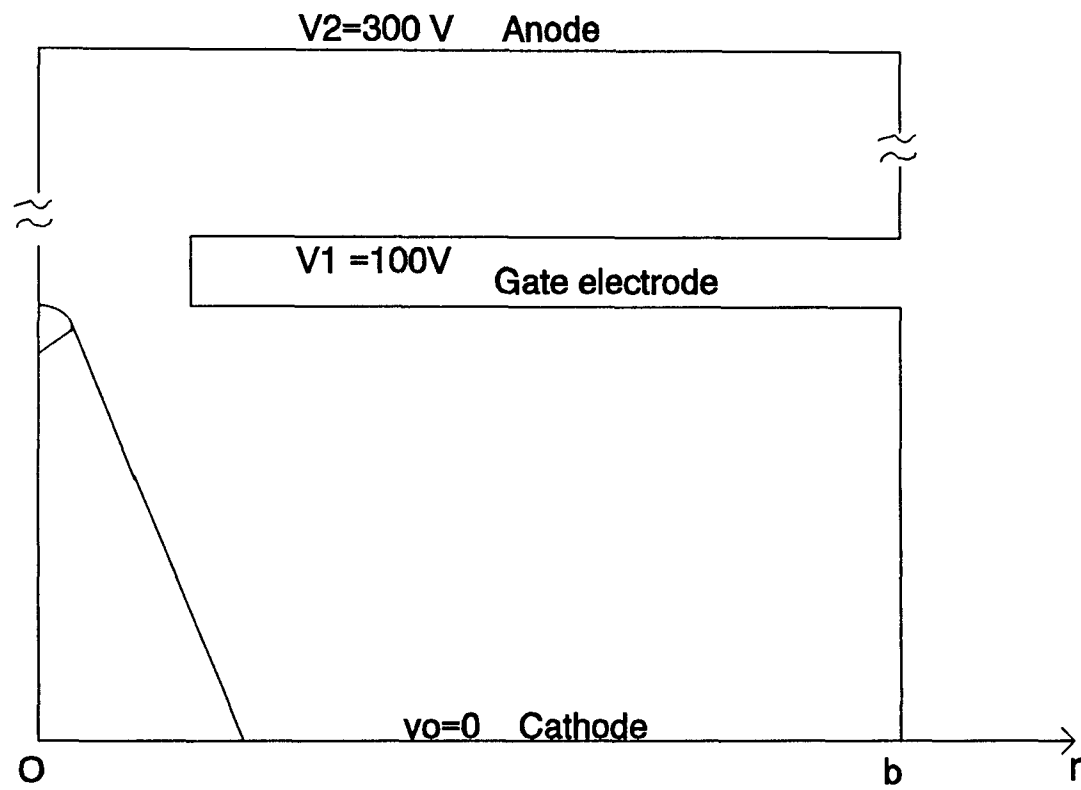
Fig. 3.4 The Geometry and Boundary Condition of the
Spindt's Device for Computer Simulation

The reason for choosing symmetrical boundaries is that from previous research[6][7], the gate opening should be concentric with the tip in order to get a uniform field.

3.3 Simulation Results and Comparison with Other Methods

A representative set of dimensional values that were used in the ANSYS simulation is shown in table 3.1.

TABLE 3.1

Spindt device dimensions for simulation

Parameter	Typical Value
t	0.4 μm
R	0.65 μm
d	1.4 μm
h	1.4 μm
r	0.05 μm
θ	20°

Fig. 3.5 shows the ANSYS model with node and the element mesh. The region of the emitter which requires high accuracy was given a dense mesh, as can be seen from observation of the figure.

These dimensions were used to compute the electric potential and electric field distribution in the vicinity of the tip. Fig 3.6 shows the computed isopotential lines for the dimensions above. The calculated field at the tip with $100V$ applied at the gate, and $110V$ applied at the anode, was $1.297 \times 10^7 \frac{V}{cm}$. Fig. 3.7 shows the electric field intensity distribution. The electric field is displayed as a vector with arrows scaled proportionally to the magnitude of the field. The electric field is shown as a single arrow at the center of each element.

The electric field E can be defined as

$$E = \alpha \frac{V}{r} = \beta V$$
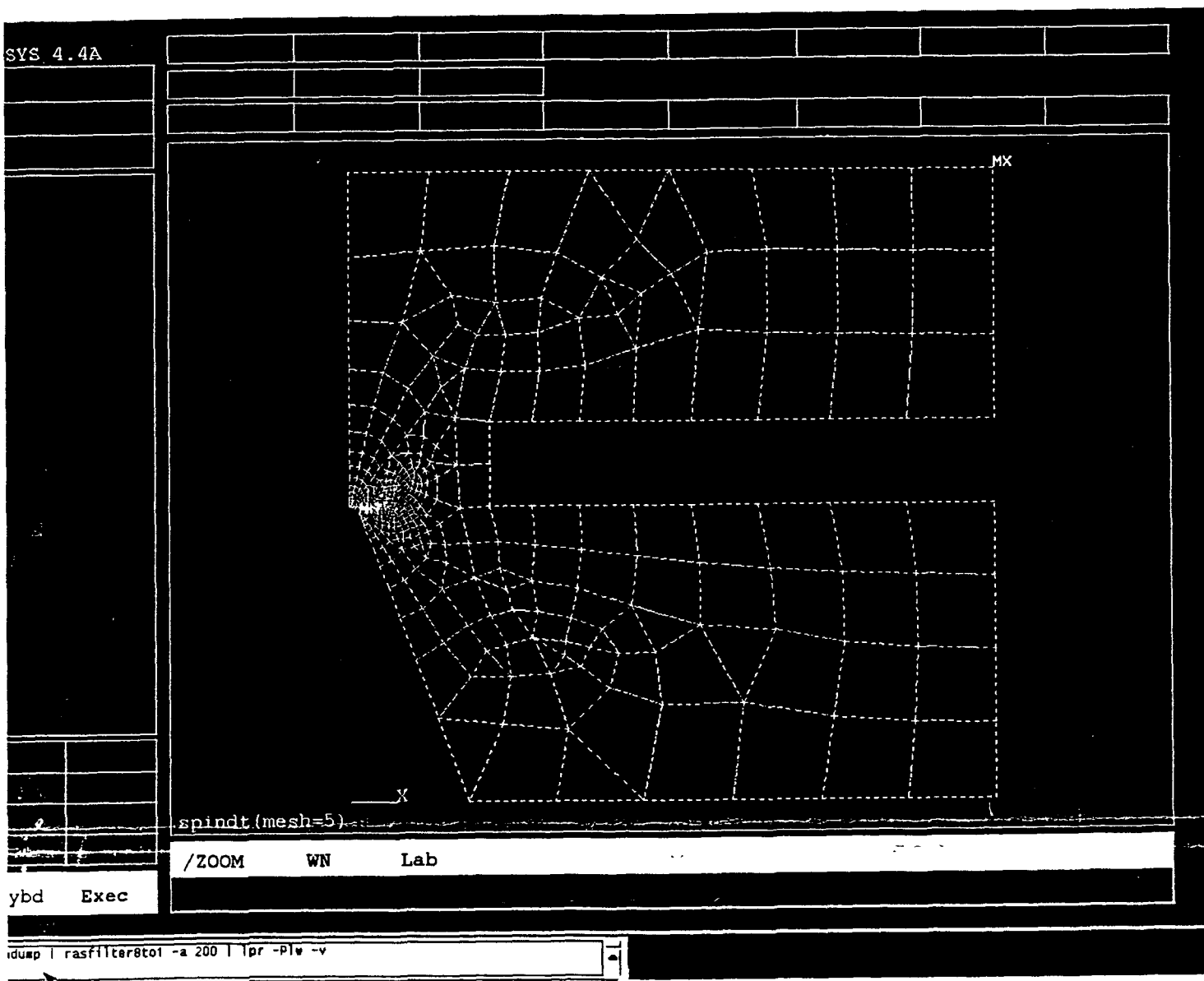
Fig. 3.5 the ANSYS model with node and the element mesh.
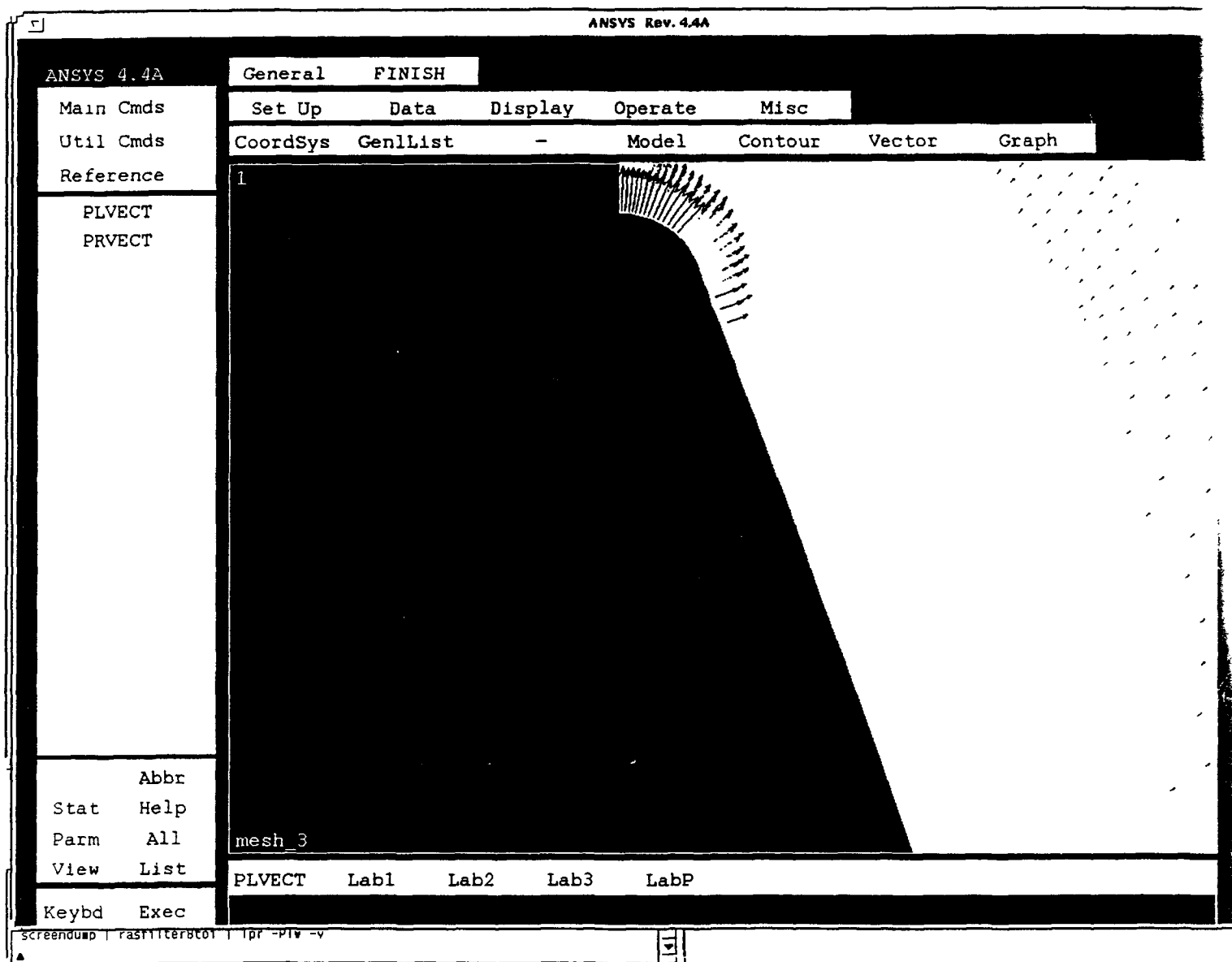
Fig. 3.7 the electric field intensity distribution.

with $\beta = \frac{E}{V}$ as $1.297 \times 10^5 cm^{-1}$. This value of β was compare with a value for an isolated sphere with a radius $r = 0.05 \mu m$ surrounded by a concentric sphere anode of radius $0.65 \mu m$. For this isolated sphere, the electric field E_i is calculated to be[7]:

$$E_i = \frac{V}{r}\frac{d+r}{d} \simeq \frac{V}{r} = \frac{100v}{0.05 \times 10^{-6}} = 2 \times 10^7 \qquad V/cm$$

The corresponding value for the isolated sphere, β_i is calculated to be:

$$\beta_i = \frac{E_i}{V} = \frac{2 \times 10^7 V/cm}{100} = 2 \times 10^5 cm^{-1}$$

Therefore the ratio of the conversion factors, defined as the reduction factor D, is

$$D = \frac{\beta_i}{\beta} = \frac{2 \times 10^5}{1.297 \times 10^5} = 1.54$$

The electric field at the realistic tip with the same dimensions mentioned above which was calculated using relaxation method by Spindt[7] was $1.250 \times 10^7 \frac{V}{cm}$. His calculated reduction factor is $D = 1.7$ compared to the value of 1.54 calculated in this thesis. In another research did by Luo[6], a value E of $1.290 \times 10^7 \frac{V}{cm}$ and a reduction factor of 1.4357 is obtained by using a derived formula.

The calculated value of electric field for a wedge emitter, with the same dimensions, is $0.993 \times 10^7 \frac{V}{cm}$. Table 3.2 summarizes calculation results of electric field, and β, α using different method of spindt device

TABLE 3.2

Results for Analysis of Spindt device

Geometry	Method	$E(V/cm)$	$\alpha(D^{-1})$	$\beta(cm^{-1})$
Cone	Spindt's	1.250×10^7	0.6250	1.250×10^5
Cone	ANSYS	1.296×10^7	0.6480	1.297×10^5
Cone	Luo Enze's	2.580×10^7	0.6965	1.290×10^5
Wedge	ANSYS	0.994×10^7	0.4969	0.993×10^5
Wedge	Analytic	0.785×10^7	0.2782	0.785×10^5

3.3.1 Discussion of simulation result

From the calculation above, it can be seen that the reduction factor calculated from ANSYS differs by 3.5 % from Spindt's result, and by 7.5 % from Luo's result. Since three entirely different methods give value for reduction coefficient by no more than 7.5% difference, we can confident that actual experimental value should be in this range. This result also meet expected result which electric field of wedge emitter should be lower than that of cone emitter because of the less curvature in one dimension.

3.4 Effects of Cathode Dimension on Performance

From the simulation results above, it can be seen that the result is in a close agreement with Spindt's result which was widely accepted. In the section, geometry of structure was investigated in depth.

3.4.1 Dependence on mesh size of tip

Like all numerical methods, the accuracy of calculation is very important and has to be taken into consideration. The accuracy of calculation mainly depends on the mesh size. Usually the smaller the mesh size, the greater the accuracy of the calculation. When a smaller mesh size is used in ANSYS, the number of elements will be increased. Owing to wavefront limitation of ANSYS, certain moderate mesh size has been selected. Fig 3.8 compares the effects of different mesh size on the emission.
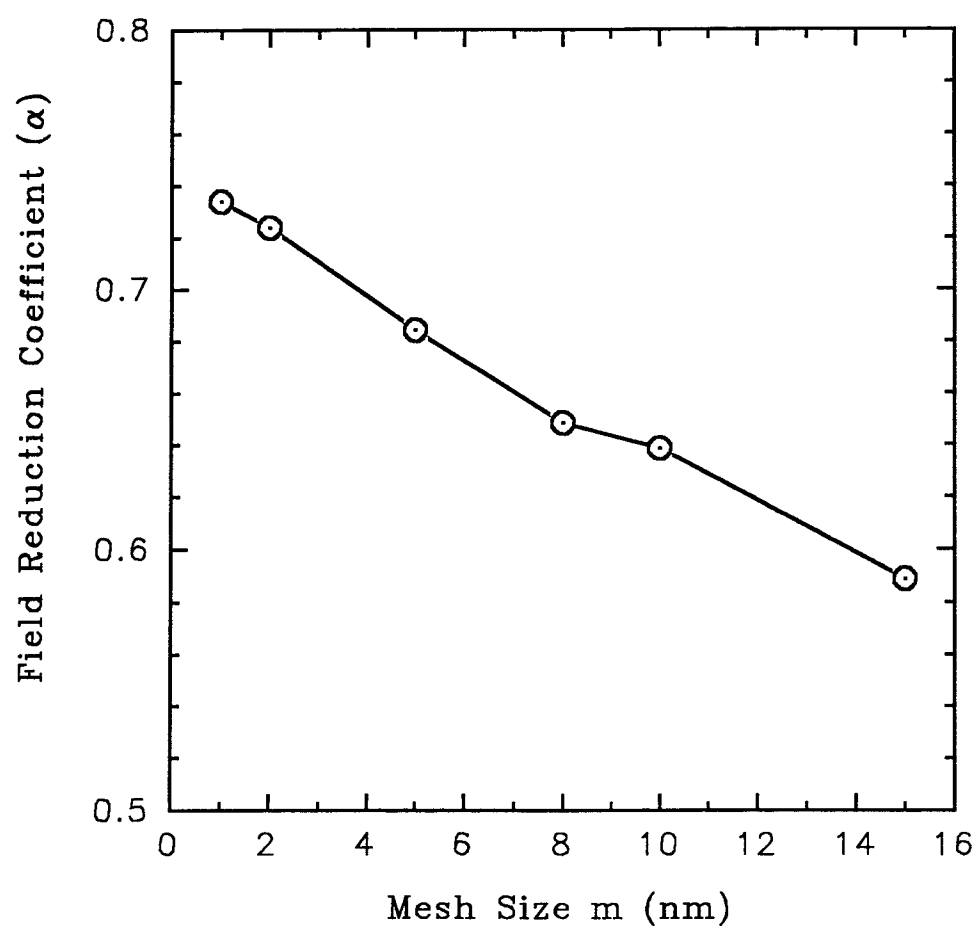
Fig. 3.8 Field reduction coefficient α as a function of mesh size

3.4.2 Dependence on the gate electrode length

The gated emitter model doesn't specify width of the gate (refer to Fig. 3.4). However in the finite element calculation, we need a closed boundary to perform the calculation. To solve this problem, we have to make a closed boundary by cutting the gate electrode to a finite length, provided that it doesn't affect the calculation results. Owing to the wavefront limit of ANSYS, a moderate value of boundary b is selected. Fig. 3.9 shows the field reduction coefficient verses gate electrode length.
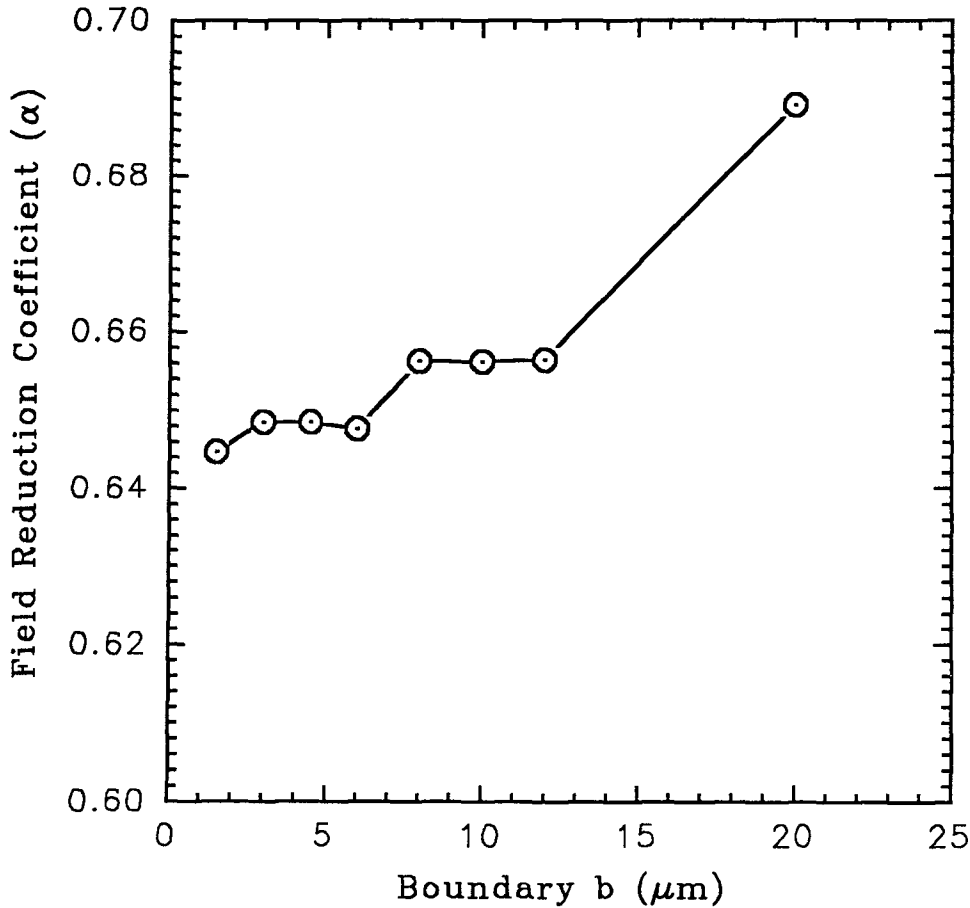
Fig. 3.9 Field reduction coefficient α as a function of b

3.4.3 Dependence on spatial angle ϕ

Sometimes when we calculate total emission current, the emission area needs to be known. Most of calculations just assume the emission area. If the relation between emission and spatial angle is known, then how much of area is contributed to the emission is known. Fig. 3.10 shows the emission factor α as a function of ϕ. From Fig 3.10, we can see that maximum emission happens in the direction of tip normal, with emission falling off as one moves in a direction over the curved surfaces (spherical for conical tips, cylindrical for wedge tips) towards side walls.
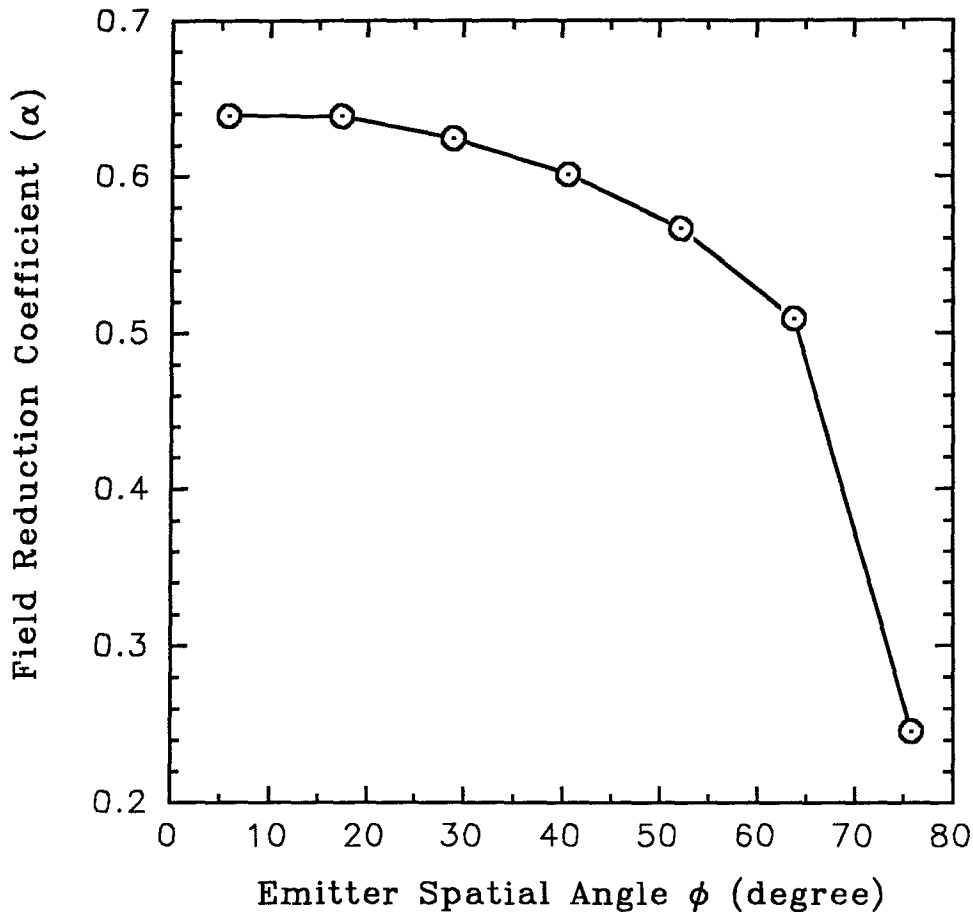
Fig. 3.10 Field reduction coefficient α as a function of ϕ

3.4.4 Dependance on emitter half angle θ

One of the factors which controls the magnitude of electric field E at the emitting surface is the proximity of neighboring potential elements. Thus a broad cone or wedge (large half-angle θ) presents a large potential surface closer to the gate with a consequent decrease the maximum E field and the electron emission. Fig. 3.11 illustrates the effect of the variation of half angle θ from 10° to 50° on field reduction
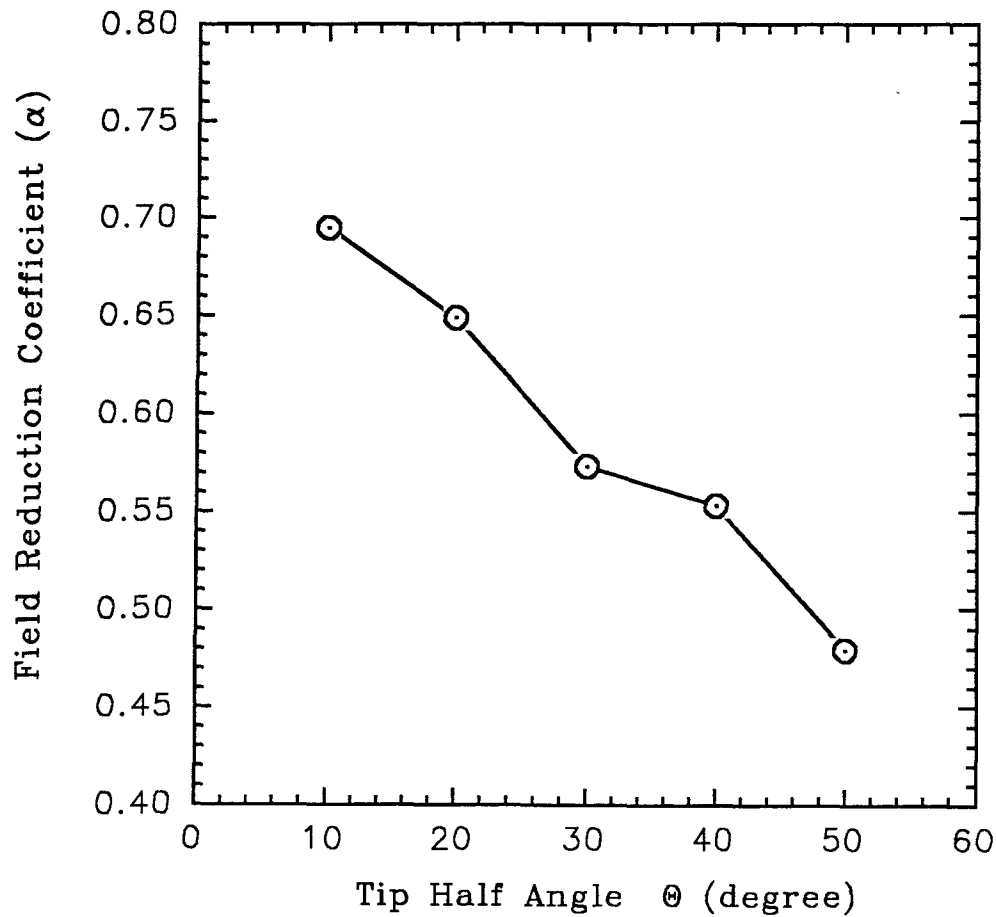
Fig. 3.11 Field reduction coefficient α as a function of θ

3.4.5 Dependence on emitter radius r:

Fig 3.12 shows the field conversion factor β as a function of tip radius r. The radius of $1nm$ to $70nm$ is simulated.
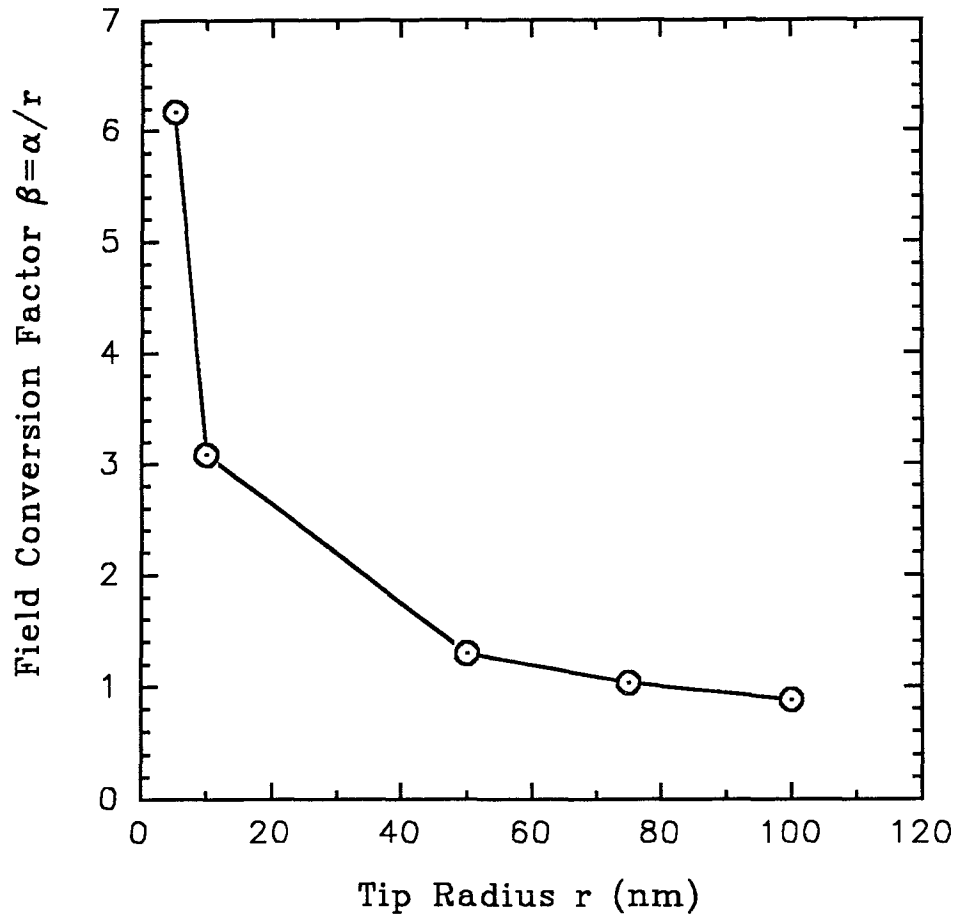
Fig. 3.12 Field conversion factor β as a function of r

From the figure, we can see that the emission depends greatly on the tip. Fig. 3.13 and Fig. 3.14 show the isopotential lines and electric field distribution along the cathode with cathode radius as $1nm$.
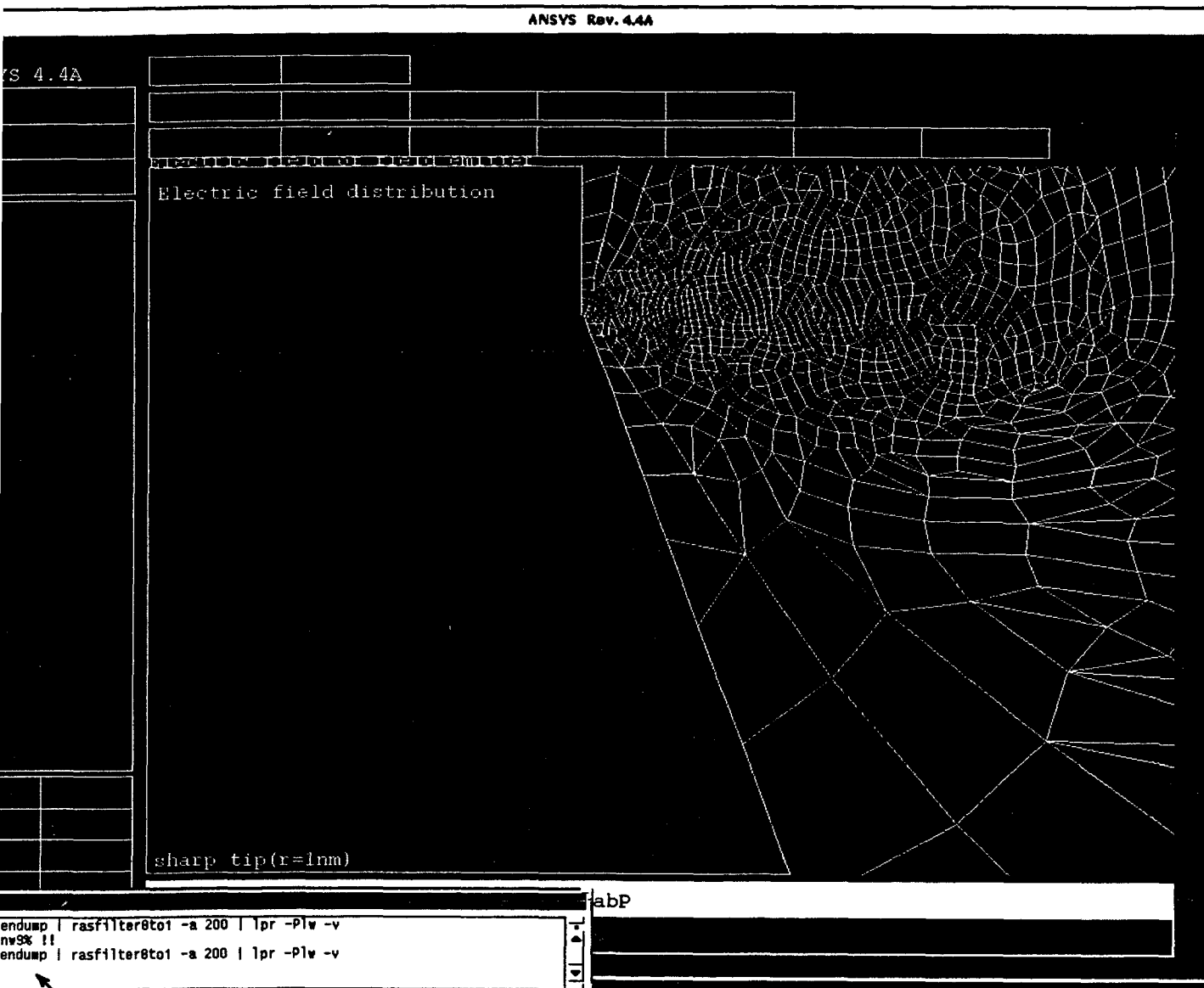
Fig. 3.14 Electric Field of Sharp Field Emitter

3.4.6 Dependence on emitter support height h:

Emitter support height is a very important parameter which influences the electric field on the surface of the emitter. With a same tip radius r and half angle θ, the higher the emitter support, the stronger the electric field. Simulations were made to show the effect of changes in support height by using the same value of essential parameters as given above. The support and base surface are assumed to be the same potential $(0V)$ as the emitter tip. Results are shown in Fig. 3.15
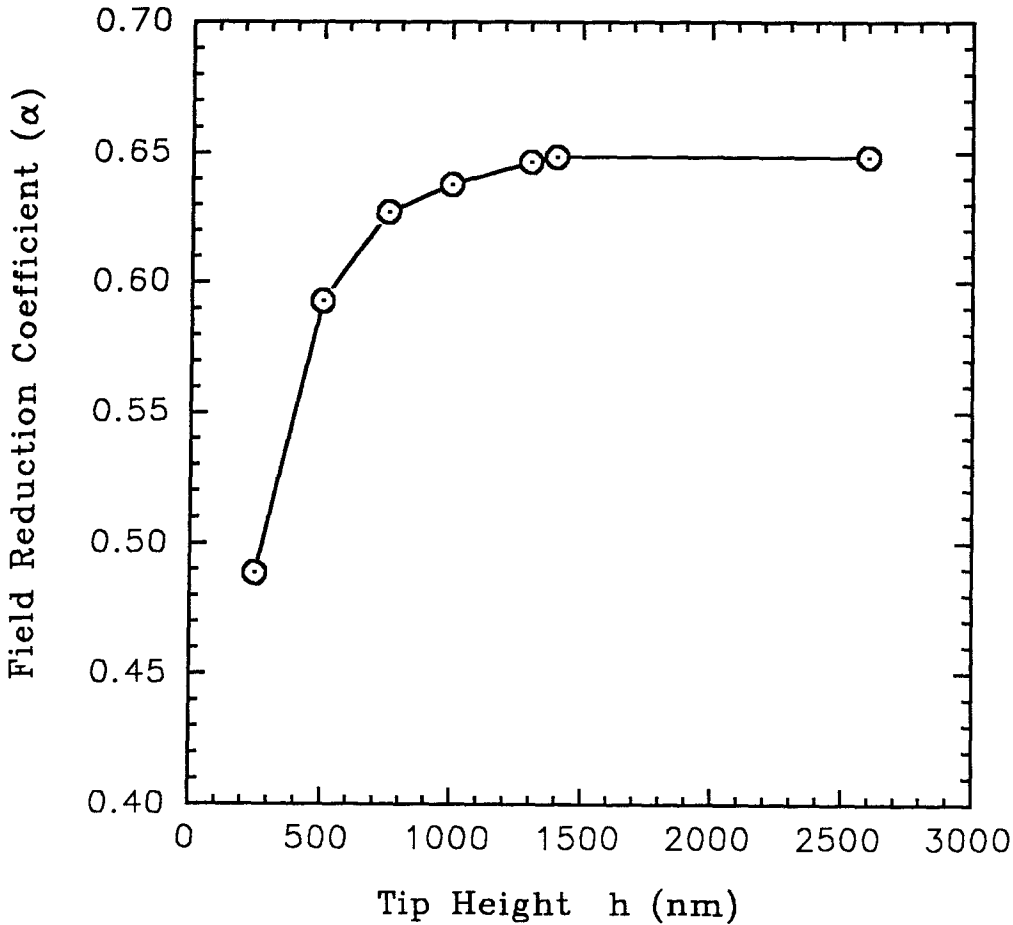
Fig. 3.15 Field reduction coefficient α as a function of h

3.4.7. Dependence on gate opening of emitter

From other result[5], it is suggested that gate opening should be as small as possible. How gate opening affects the emission is discussed. Fig. 3.16 dispatches field reduction coefficient as a function of gate opening R.
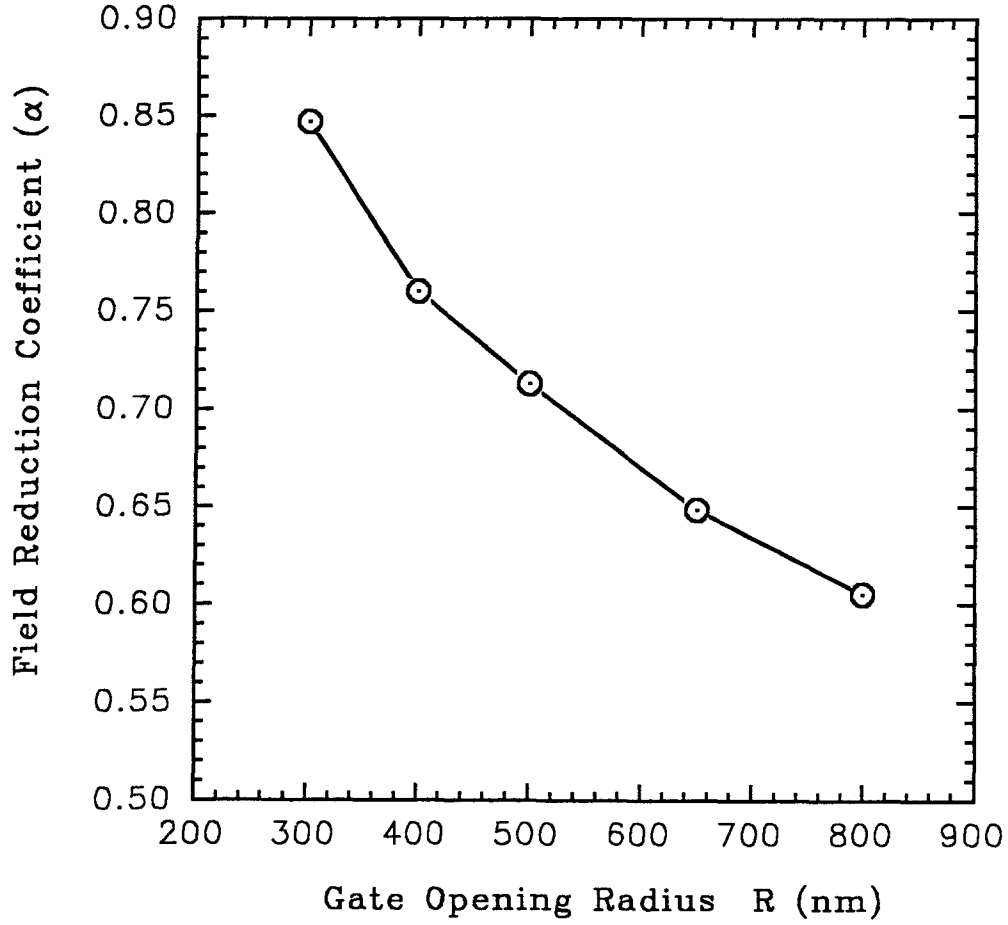
Fig. 3.16 Field reduction coefficient α as a function of R

3.4.8 Discussion of results obtained by simulation

From the above simulation results, it can be seen, that field reduction coefficient changes with mesh size with a slope of 9.68×10^{-3} (nm^{-1}) over the entire range of mesh size changes of $1 - 15nm$. therefore we can predict that, when the mesh size goes to zero, the field reduction coefficient is to be 0.745. If mesh size used in the simulation is $1nm$, the percentage error from mesh size of 0 which is supposed to be a ideal calculation is 1.4%.

The field reduction coefficient as a function of gate electrode length increases at a rate of 6.45×10^{-6} (nm^{-1}) over the entire range of gate length of $20,000nm$.

The electric field drops off about 10% when moving away from the center by 50°, therefore, if we define the effective emitting area by an angle where the field has dropped off 10%, the spatial angle is then 50°. The corresponding emitting area can be calculated in Section 3.5.

The field reduction coefficient shows a monotonic decrease of 0.0537 per 10°. So with a half angle change of 10°, the change of electric filed or reduction coefficient is about 5%.

The dependence of electric field on the cathode curvature is described by the relationship between β and radius r. The electric field with radius of $1nm$ is 6 times of electric field with radius of $100nm$.

As shown in Fig. 3.15 the dependence of emission factor on the support height is saturate when the height is greater than $1500nm$, thus if the tip made higher, the electric field will not change. Therefore a moderate height of tip should be chosen.

Fig. 3.16 shows a linear drop of reduction coefficient as gate opening increases. The slope of change of electric field intensity is 4.8×10^{-3} (nm^{-1}). The result indicates that protruding cones, with R as small as possible, should be used.

3.5 Calculation of Emission Current
3.5.1 Background for emission current

It is well known that the emission of electrons varies exponentially with field strength. Because of the high concentration of electric lines of force at a small surface area of a fine point or emitter tip, a high voltage gradient, or electric field, is produced[17]. The equation which describes the theory of field emission was first derived by Fowler and Nordheim in 1928[19]. The Fowler Nordheim equation which gives the current density J emitted from a clean metal surface, is

$$J = \frac{A \cdot E^2}{\phi \cdot t^2(y)} exp\left(\frac{-B\phi^{\frac{3}{2}}V(y)}{E}\right) \qquad A/cm^2 \qquad (3.5)$$

where E is the surface electric field in V/cm, ϕ is the work function in electron volts,

$$A = 1.54 \times 10^{-6}$$

$$B = 6.87 \times 10^7$$

$$y = 3.79 \times 10^{-4}\sqrt{E}/\phi$$

$$\phi = 4.5ev$$

where y is the factor to account for Schottky lowering of the work-function barrier. The function V(y) is the expression of the image potential for electrons near the emitter surface. The functions V(y) and t(y) can be closely approximated by the following equation over the operating range of most cathodes.

$$V(y) = 0.95 - y^2$$

$$t^2(y) \simeq 1.1$$

By substituting all these parameter and E found from the computer analysis into equation (4.27), the current density J can be determined at each point on the emitter surface. Once we get the J, the total emission current can be determined. Assuming the total emission area is divided into several small regions, the electric

field is constant within each small region. Then the area of each small region can be given by :

$$\sigma_i = 2\pi \cdot r \cdot sin\psi_i \cdot \delta l \qquad (3.6)$$

where r is the tip radius, and δl and ψ_i are as shown in Fig 3.17.

To obtain the total emission current, the current density are integrated over the total tip area:

$$I = \sum_k \sigma_i \cdot J_i \qquad (3.7)$$

3.5.2 Result of emission current

Now we have already had the electric filed from the simulation for both cone and wedge structure. Using the above equations (3.5 - 3.7), the current density in each small region and total emittion current can be summarized in table below.

TABLE 3.3

Electric field and current density
and total emission current of Spindt device

Angle(ϕ)	$E(V/cm)$(Cone)	$J(A/cm^2)$(Cone)	$E(V/cm)$(Wedge)	$J(A/cm^2)$(Wedge)
5.8°	1.297×10^7	7.634×10^{-12}	0.9934×10^7	1.946×10^{-18}
17.4°	1.296×10^7	7.345×10^{-12}	0.9815×10^7	8.595×10^{-19}
29.4°	1.281×10^7	4.168×10^{-12}	0.9634×10^7	2.519×10^{-19}
40.6°	1.253×10^7	1.357×10^{-12}	0.9355×10^7	3.453×10^{-20}
52.2°	1.206×10^7	1.823×10^{-13}	0.8933×10^7	1.357×10^{-21}
63.8°	1.132×10^7	5.294×10^{-15}	0.8308×10^7	6.183×10^{-24}
Total Current	$3.829 \times 10^{-22}(A)$		$6.262 \times 10^{-31}(A)$	
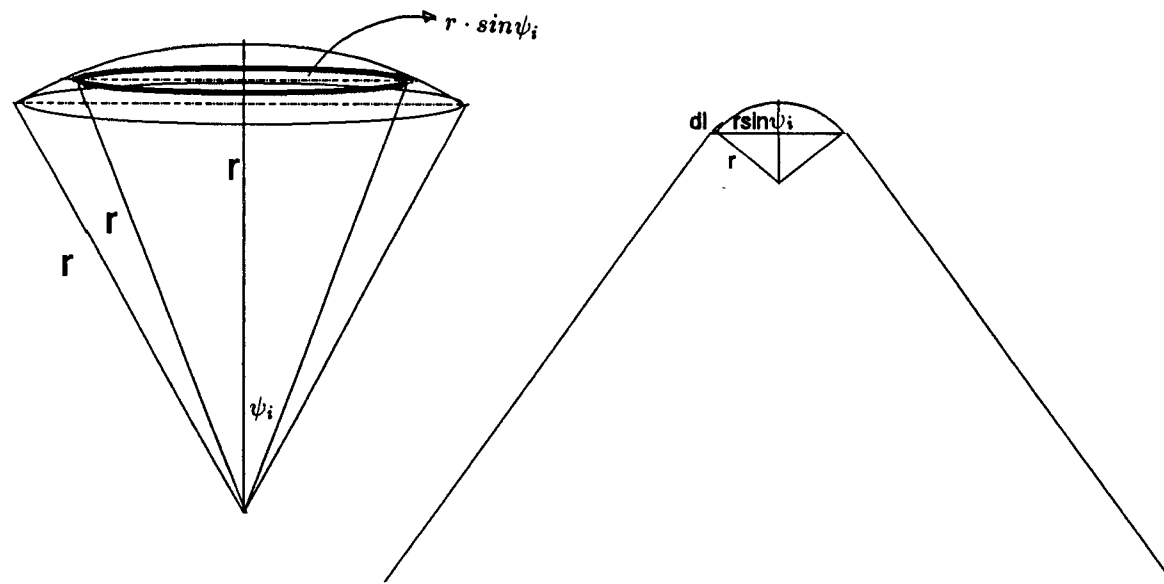
Fig3.17 The Area of Small Region Used in Current Calculation

Chapter 4
Theoretical Analysis

A numerical method was used to find approximate solution of Laplace equation for realistic cathode structures as described in chapter 3. However, an analytical solution which is more concise and accurate is required. Unfortunately, it was difficult to derive a closed expression for a real cathode structure. It is necessary to assume a very simplistic model for a wedge shaped emitter having an anode as a large circle surrounding a cathode at its center. A closed form solution of Laplace equation could be obtained by using a conformal transform for this model. The solution can be used as a close approximation for the actual situation. The basic background for analyzing the simplistic model is presented in section 4.1. The analysis of electric field and reduction coefficient for wedge emitter structure using this simplistic model, are described in 4.2, 4.3 and 4.4.

4.1 Theoretical Model and General Background

As mentioned above, in this thesis work only the wedge cathode structure has been studied to obtain expression for the E field in a closed form.

For the wedge model shown in Fig 4.1, a is the inner radius of the tip, R is the radius of the anode and θ_0 defines the support boundary. Also C_1 represents the boundary of the anode, C_2 represents the boundary of the cathode, L_1 and L_2 correspond to the boundary of the support. The boundary conditions for voltage are $V(R) = V$ and $V(a) = 0$.
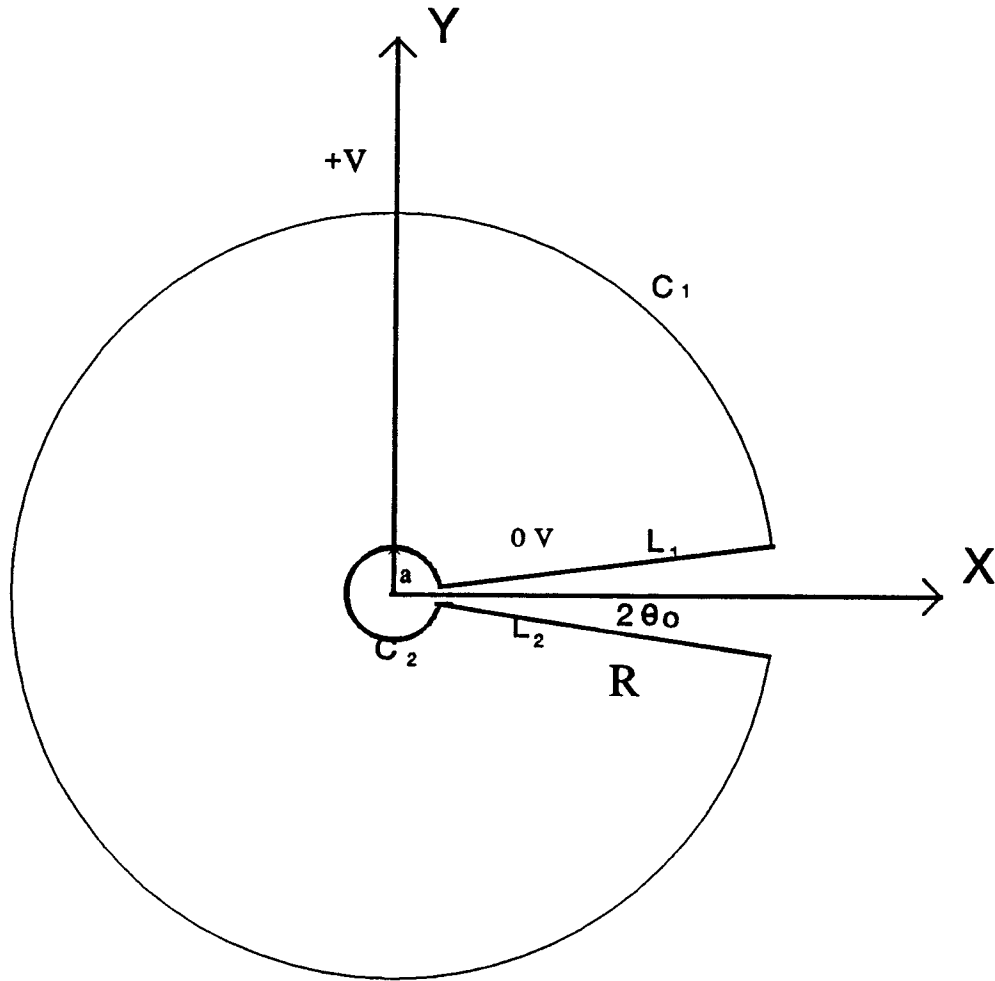
Fig 4.1 Wedge theoretical Model and Boundary condition

For the system shown above, no variation with z is expected except near the ends. Since the wedge is supported, the potential must be a function of θ_0 as well as θ. With these conditions, the Laplace equation in the cylindrical coordinate degenerates to a second order differential equation that can be rewritten as :

$$\nabla^2 \Phi = \frac{1}{r} \cdot \frac{\partial}{\partial r}\left(\frac{r \cdot \partial \Phi}{\partial r}\right) + \frac{1}{r^2} \cdot \frac{\partial^2 \Phi}{\partial \theta^2} = 0 \qquad (4.1a)$$

where Φ is the potential of point P and r is the distance between the origin O and

point P. Equation (4.1a) is identical to:

$$-\left(\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2}\right) = 0 \qquad (4.1b)$$

So the problem transfers to a rectangular coordinate problem with equation (4.1b) and boundaries as following:

$$C_1 \qquad \rho = R \qquad (\theta_0 < \theta < 2\pi - \theta_0) \qquad (4.2a);$$

$$C_2 \qquad \rho = a \qquad (\theta_0 < \theta < 2\pi - \theta_0) \qquad (4.2b);$$

$$L_1 \qquad \theta = \theta_0 \qquad (a < \rho < R) \qquad (4.2c);$$

and

$$L_2 \qquad \theta = 2\pi - \theta_0 \qquad (a < \rho < R) \qquad (4.2d);$$

The partial differential equations of mathematical physics, which are often referred as boundary-value problems, are often solved conveniently by *separation of variables*. In the process, one often generates orthogonal sets of functions which are useful in their own right. Equations involving the three-dimensional Laplacian operator are known to be separable in eleven different coordinate systems[19]. The equation to be solved is the Laplace equation in the rectangular coordinates(referring (4.1b), so equation can be separable. However the boundary conditions is not separable in the rectangular coordinate system. The equation (4.1b) together with the boundary condition must be transfer to a coordinate system in which both boundary condition and equation can be separable. As we know, if the two-dimensional orthogonal coordinate system is obtained by analytic transformation, then Laplace equation can be separable in such coordinate system[16]. Assuming that there is an arbitrary analytical function $W = f(z)$, that is a function of the complex variable $z = x + jy$, W can be divided into real and imaginary parts:

$$W = f(z) = u(x,y) + iv(x,y) = \xi + i\eta$$

From the Cauchy-Riemann theorem, if a conformal transformation $W = f(z) = \ln z$ is performed, then u and v are perpendicular.

$$W = \xi + i\eta = \ln z = \ln(\rho e^{i\theta}) = \ln\rho + i\theta \qquad (4.3)$$

Thus in the ξ, η coordinate system, Laplace Eq.(4.1b) becomes

$$-\left(\frac{\partial^2\Phi}{\partial\xi^2} + \frac{\partial^2\Phi}{\partial\eta^2}\right) f(\xi, \eta) = 0 \qquad (4.4)$$

or

$$-\left(\frac{\partial^2\Phi}{\partial\xi^2} + \frac{\partial^2\Phi}{\partial\eta^2}\right) = 0 \qquad (4.5)$$

It is easy to see that boundaries in the new coordinate system become:

$$C_2 \qquad \xi = \ln R \qquad \theta_0 < \eta < 2\pi - \alpha \qquad (4.6a)$$

$$C_1 \qquad \xi = \ln a \qquad \theta_0 < \eta < 2\pi - \theta_0 \qquad (4.6b)$$

$$L_1 \qquad \eta = \theta_0 \qquad (\ln a < \xi < \ln R) \qquad (4.6c)$$

$$L_2 \qquad \eta = 2\pi - \theta_0 \qquad (\ln a < \xi < \ln R) \qquad (4.6d)$$
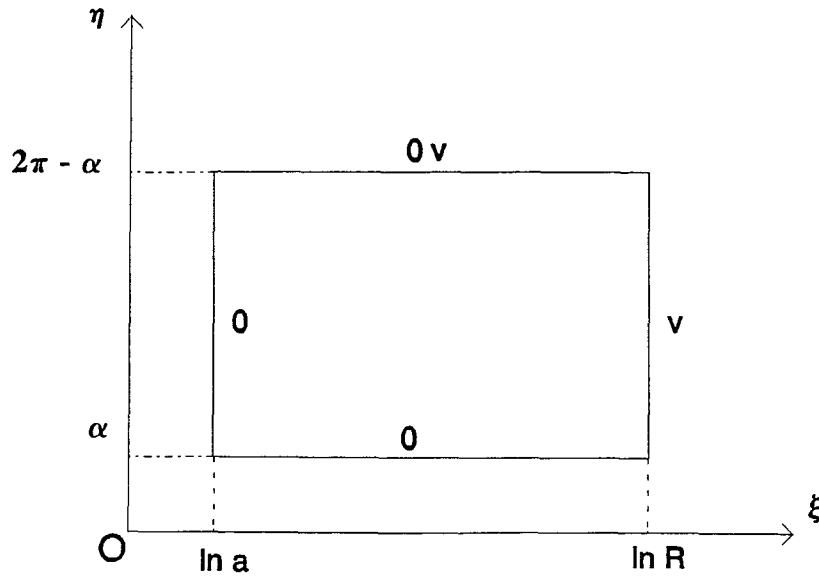
Fig. 4.3 Boundary of wedge after conformal transform

As shown in Fig 4.3, the boundary conditions are as following:

$$\begin{cases} \Phi|_{\xi=lna} = 0 \\[2mm] \Phi|_{\xi=lnR} = V \\[2mm] \Phi|_{\eta=\theta_0} = 0 \\[2mm] \Phi|_{\eta=2\pi-\theta_0} = 0 \end{cases} \qquad (4.7)$$

4.2 Derivation of Field Equation for the Emitter Structure

Using the above equations, the solution in the new coordinate system is the same as in the rectangular system. A solution of this partial differential equation can be found in terms of two ordinary differential equations, all of the same form, by the assumption that the potential can be represented by a product of two functions, one for each coordinate:

$$\Phi(\xi,\eta) = \xi(\xi)\eta(\eta) \qquad (4.8)$$

Substitute the above equation into (4.5) an division of the result by (4.8) yields

$$\frac{1}{\xi(\xi)}\frac{d^2\xi}{d\xi^2} + \frac{1}{\eta(\eta)}\frac{d^2\eta}{d\eta^2} = 0 \qquad (4.9)$$

while total derivatives have replaced partial derivatives, since each term involves a function of one variable only. If (4.9) is to hold for arbitrary values of the independent coordinates, each of the two terms must be separately constant:

$$\frac{1}{\xi}\frac{d^2\xi}{d\xi^2} = -\mu^2 \qquad (4.10a)$$

$$\frac{1}{\eta}\frac{d^2\eta}{d\eta^2} = \nu^2 \qquad (4.10b)$$

$$-\mu^2 + \nu^2 = 0$$

If μ^2, ν^2 are arbitrarily chosen to be positive, and letting $\mu^2 = k^2$, then the solutions of the two ordinary differential equations can be written as:

$$\eta(\eta) = A_1 sink\eta + B_1 cosk\eta \qquad (4.11a)$$

$$\xi(\xi) = A_2 e^{k\xi} + B_2 e^{-k\xi} \qquad (4.11b)$$

where $\qquad k^2 = \mu^2 = \nu^2 \qquad (4.12)$

The potential (4.8) can thus be built up from the product solutions:

$$\Phi(\xi, \eta) = (A_2 e^{k\xi} + B_2 e^{-k\xi})(A_1 \sin k\eta + B_1 \cos k\eta) \qquad (4.13)$$

At this stage μ or ν are completely arbitrary. Consequently (4.13), by linear super-position, represents a very large class of solutions to the Laplace equation. It's not easy to find constant A_1, A_2, B_1, B_2 and k, if boundary conditions are applied. After performing coordinate translation:

$$\begin{cases} \xi' = \xi - \ln a \\ \\ \eta' = \eta - \theta_0 \end{cases} \qquad (4.14)$$

as shown in Fig 4.4, the equations (4.15a) and (4.15b) are obtained.

$$\eta(\eta') = A_1' \sin k\eta' + B_1' \cos k\eta' \qquad (4.15a)$$

$$\xi(\xi') = A_2' e^{k\xi'} + B_2' e^{-k\xi'} \qquad (4.15b)$$
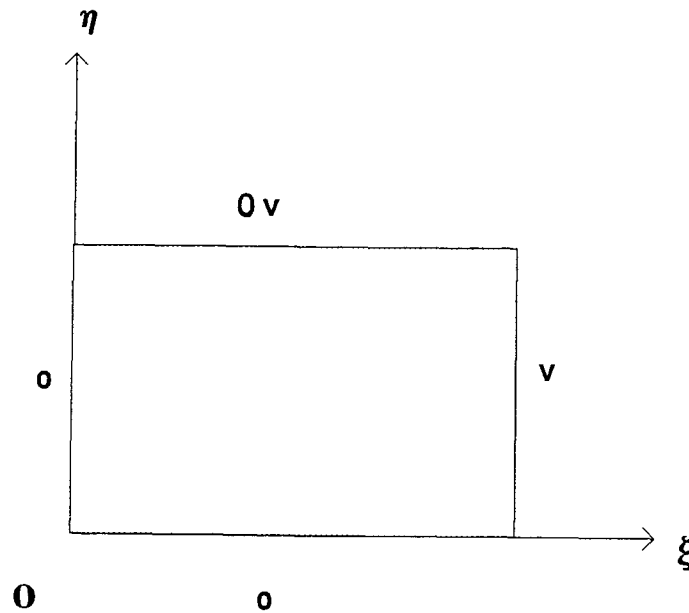
Fig. 4.4 Boundary of Wedge after Coordinate Transfom

Thus the boundary condition changes to

$$\begin{cases} \Phi|_{\xi'=0} = 0 \\[2mm] \Phi|_{\xi'=ln\frac{R}{a}} = V \\[2mm] \Phi|_{\eta'=\theta_0} = 0 \\[2mm] \Phi|_{\eta'=2\pi-\theta_0} = 0 \end{cases} \qquad (4.16)$$

To determine μ or ν, it is necessary to impose specific boundary conditions on the potential. Substituting the boundary conditions (4.16) into Eqs (4.15), we have

$$B_1' = 0, A_2' = -B_2'$$

$$k_n = \mu_n = \frac{n\pi}{2(\pi - \theta_0)}$$

$$\eta(\eta') = A_1' sin k_n \eta' \qquad (4.16a)$$

$$\xi(\xi') = A_2'(e^{k_n\xi'} - e^{-k_n\xi'}) \qquad (4.16b)$$

$$\Phi_n = (e^{k_n\xi'} - e^{-k_n\xi'}) sin k_n \eta' \qquad (4.17)$$

The potential can be expanded in terms of these k_n with initially arbitrary coefficients, chosen to satisfy the final boundary conditions(4.14):

$$\Phi(\xi',\eta') = \sum_{\substack{n=1 \\ n=odd}}^{\infty} A_n(e^{k\xi'} - e^{-k\xi'}) sin k\eta' \qquad (4.18)$$

There remains only the boundary condition $\xi' = ln\frac{R}{a}, \Phi = V$, i.e. $\Phi(ln\frac{R}{a},\eta') = V$ that must be satisfied.

Equation (4.18) is a Fourier series for the function Φ. Consequently the coefficients A_n are given by:

$$A_n = \frac{2}{2(\pi - \theta_0)(e^{k_n ln\frac{R}{a}} - e^{-k_n ln\frac{R}{a}})} \int_0^{2(\pi-\theta_0)} V sin(k_n\eta')d\eta'$$

$$= \frac{4V}{n\pi(e^{k_n \ln \frac{R}{a}} - e^{-k_n \ln \frac{R}{a}})} \qquad n = 1, 3, 5, \ldots \qquad (4.19)$$

Thus we have

$$\Phi(\xi', \eta') = \sum_{\substack{n=1 \\ n=odd}}^{\infty} \frac{4V}{n\pi(e^{k_n \ln \frac{R}{a}} - e^{-k_n \ln \frac{R}{a}})} (e^{k_n \xi'} - e^{-k_n \xi'}) \sin k_n \eta'$$

$$= \sum_{\substack{n=1 \\ n=odd}}^{\infty} \frac{4V}{n\pi} \left(\frac{\sinh(k_n \ln \xi')}{\sinh(k_n \ln \frac{R}{a})} \right) \sin k_n \eta' \qquad (4.20)$$

Since $\xi' = \xi - \ln a = \ln \frac{\rho}{a}$ and $\qquad \eta' = \eta - \alpha = \theta - \theta_0,$ $\qquad \Phi$ can be rewritten

as

$$\Phi(\rho, \theta) = \sum_{n=1}^{\infty} \frac{4V}{n\pi} \left(\frac{\sinh(k_n \ln \frac{\rho}{a})}{\sinh(k_n \ln \frac{R}{a})} \right) \sin k_n(\theta - \theta_0) \qquad (n = 1, 3, 5, \ldots) \qquad (4.21)$$

Therefore, the electric field $\vec{E} = \vec{E}_\rho + \vec{E}_\theta$

$$\overline{E}_\rho = -\frac{\partial \Phi}{\partial \rho} \hat{\rho} = - \sum_{\substack{n=1 \\ n=odd}}^{\infty} \frac{2V}{\rho(\pi - \theta_0)} \left(\frac{\cosh(k_n \ln \frac{\rho}{a})}{\sinh(k_n \ln \frac{R}{a})} \right) \sin k_n(\theta - \theta_0) \qquad (4.22a)$$

$$\overline{E}_\theta = -\frac{1}{\rho}\frac{\partial \Phi}{\partial \theta} \hat{\theta} = - \sum_{\substack{n=1 \\ n=odd}}^{\infty} \frac{2V}{\rho(\pi - \theta_0)} \left(\frac{\sinh(k_n \ln \frac{\rho}{a})}{\sinh(k_n \ln \frac{R}{a})} \right) \cos k_n(\theta - \theta_0) \qquad (4.22b)$$

$$E = \sqrt{E_\rho^2 + \rho^2 \cdot E_\theta^2} \qquad (4.23)$$

4.3 Derivation of Field Reduction Coefficient

Having derived three equations which voltage and electric field can be calculated

for wedge emitter, it is necessary to derive an expression for the reduction in E due

to possible geometry change that we are interested in, particularly the reduction

coefficient α due to the support.

(a)

(b)

Fig 4.5 Concentric Cylinder Emitter Model

Now consider the simple case for a concentric cylinder. Laplace equation in the cylindrical coordinates is expressed in (4.1a). As shown in Fig. 4.5, there is no variation with z for the analysis of the wedge field emitter. Furthermore, because of cylindrical symmetry, the potential is independent of θ. Accordingly, the equation (4.1a) becomes

$$\frac{1}{r} \cdot \frac{\partial}{\partial r}\left(\frac{r \cdot \partial \Phi}{\partial r}\right) = 0$$

By solving this second order differential equation, the potential between two cylinders is obtained as:

$$\Phi = \frac{V}{ln(a/R)} \cdot ln(r/R) \qquad (4.24)$$

so that

$$\vec{E}_0(r = a) = -\frac{\partial \Phi}{\partial r} = \frac{V}{a \cdot ln\frac{R}{a}} \qquad (4.25)$$

while with a supporting structure for the emitter tip, a reduction in the field strength is expected and this is accounted for using a reduction coefficient α on the top of inner cylinder.

$$\vec{E}(\rho = a, \theta = 180°) = \alpha\vec{E}_0 = \frac{\alpha V}{a \cdot ln\frac{R}{a}} \qquad (4.26)$$

where $\vec{E}(\rho = a, \theta = 180°)$ can be calculated from Eq(4.22).

Thus the field reduction coefficient due to the support structure can be expressed as

$$\alpha(\rho = a, \theta = 180°) = \frac{\vec{E}}{V} \cdot a \cdot ln\frac{R}{a} \qquad (4.27)$$

4.4 Calculated Result of Wedge Emitter

From the Eq. 4.21, we can draw the relationship between the voltage Φ and θ, θ_0, ρ. Fig 4.7 is the result of the calculation of the dependance of voltage on the ρ, θ, θ_0. From the Fig 4.7, we see that a wedge emitter with half support angle (small θ_0) has higher voltage at the same observation point.

From Eq(4.21) the equalpotential line can be drawn on the tip of emitter. Once we get the electric field on the surface. The emission current can be calculated. Fig. 4.8 is the isopotential line on the wedge surface, from the Fig 4.8, we know the isopotential line is densest on the center top of the wedge of the emitter, so we can expect highest electric field on the top.

Fig. 4.10 shows the electric field on the tip surface for different support angle as defined in Fig 4.9. From the Fig. 4.10, it can be seen that highest electric field is located at the center of emitter surface (where $\theta = 180°$). Also it is seen that electric field decreases fairly rapidly as θ decreases from 130° toward 0°. The electric field depends greatly on angular shape of emitter support. For smaller emitter support angle, the charge would distribute over a relatively smaller surface area, and thus the corresponding electric flux would be denser in this area, and thus a higher

electric field would be obtained.

Fig. 4.11 shows the dependence of electric field on the spatial angle for different a/R ratios. From the results, it can be seen that a higher electric field at the tip is obtained when a is smaller (Here we keep anode radius r unchanged).

The following computing results (list in Appendice A) show the reduction coefficient α calculated using Eqs (4.20-4.22) under different geometrical condition.

From the above table, we can see that with large a/R ratio, the electric field of the wedge with a support is approximate closer to the electric field without a support. On the other hand, when the cathode radius a/R is very small, the electric field of emitter with support is greatly less than that of emitter without support. We get the conclusion that the support has great effect on the emitter with small a/R ratio, and that the bigger the support half angle, the larger the effect. The above table is depicted in Fig. 4.12(a) and 4.12(b)
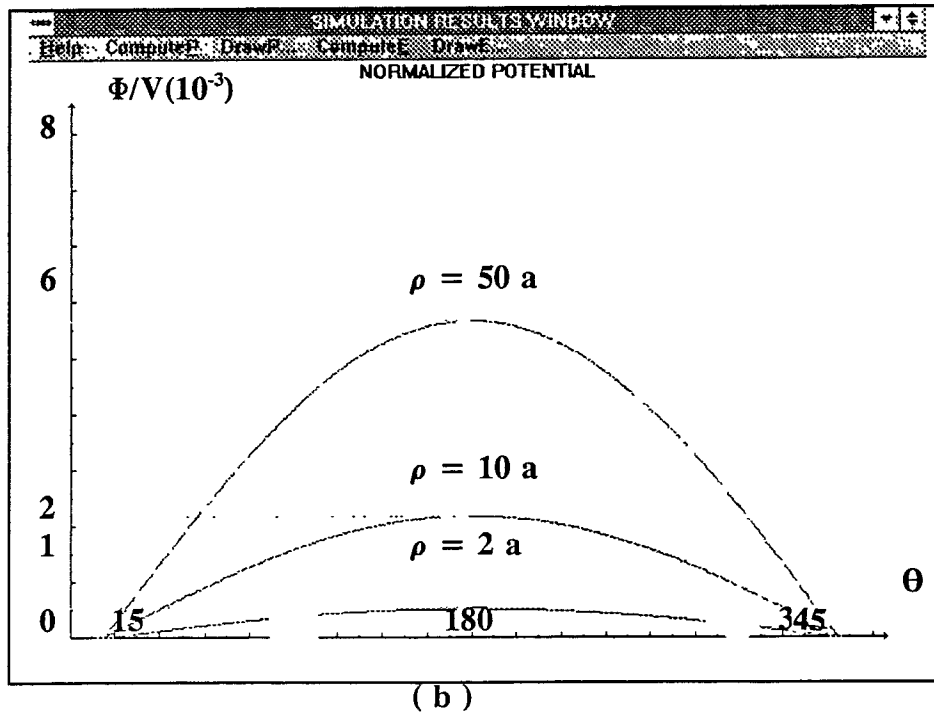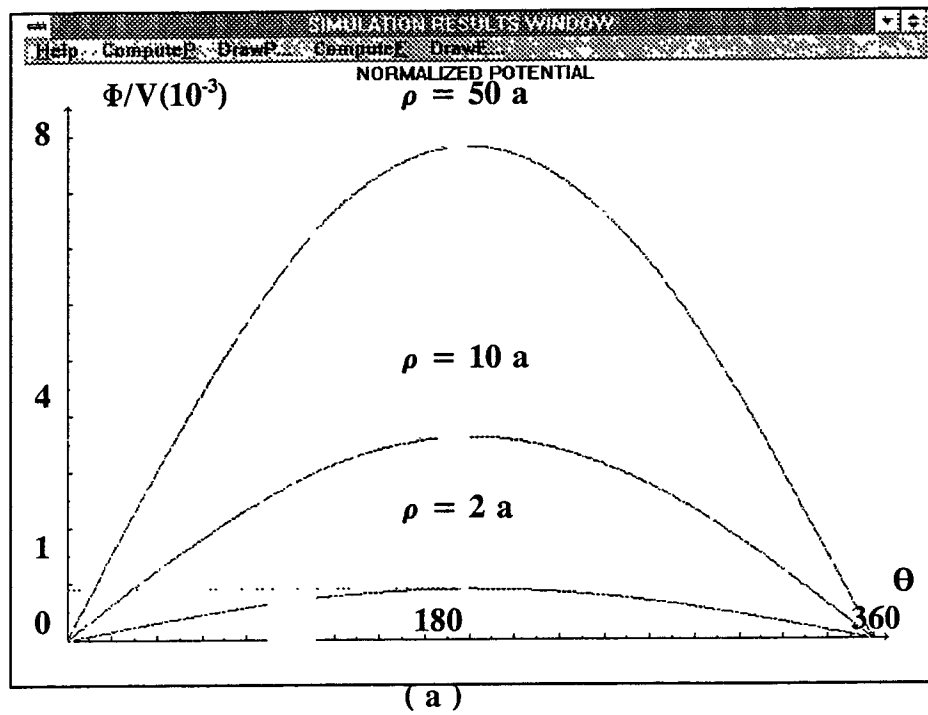
Fig 4.7 The Dependence of Voltage of the Emitter on (ρ, θ_0, θ)

(a) $\theta_0 = 0°$ (b) $\theta_0 = 15°$
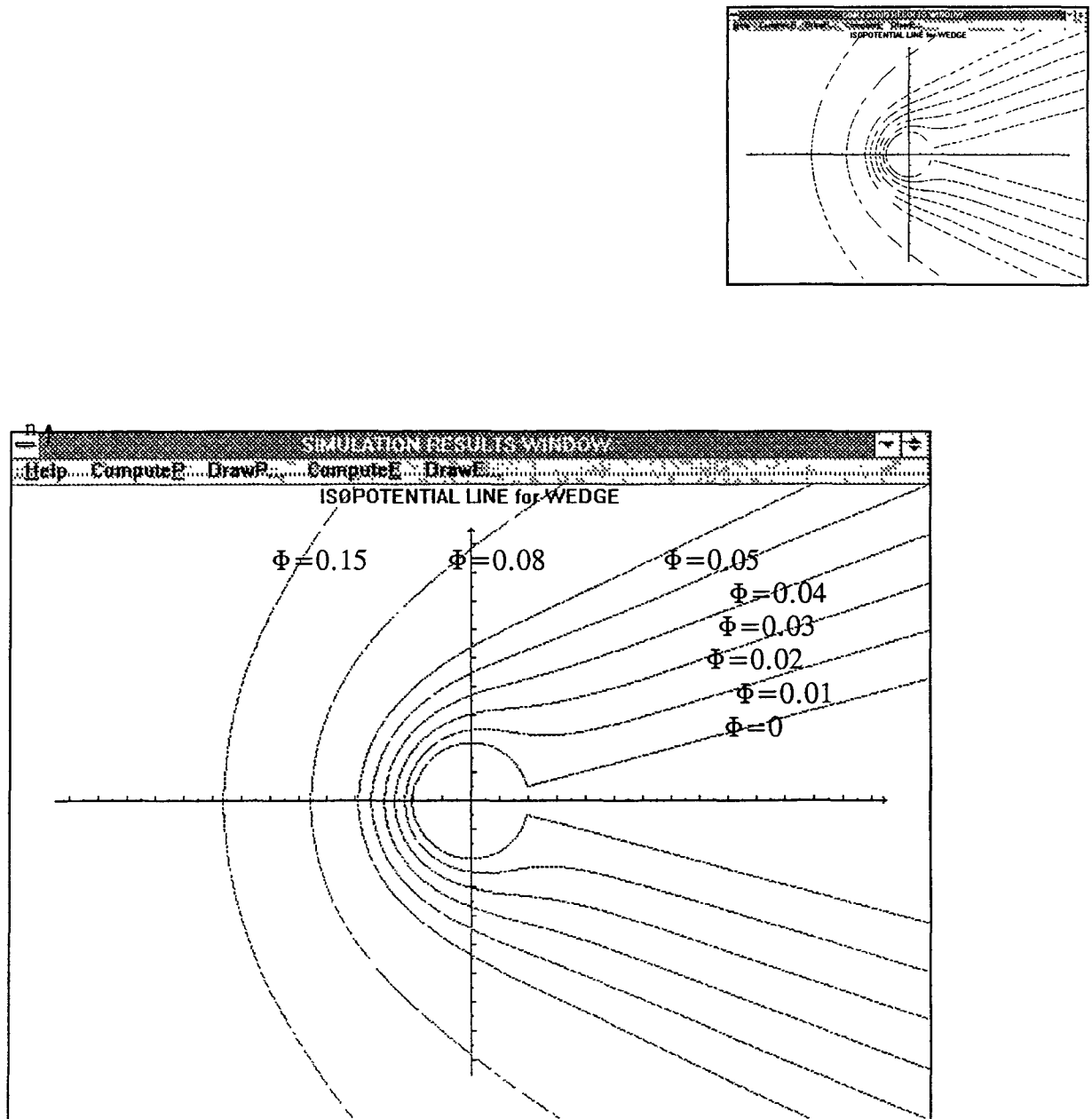
Fig 4.8 The Isopotential Line on the Wedge Surface

(a)
$\theta_0 = 0°$

(b)
$\theta_0 = 15°$

(c)
$\theta_0 = 90°$
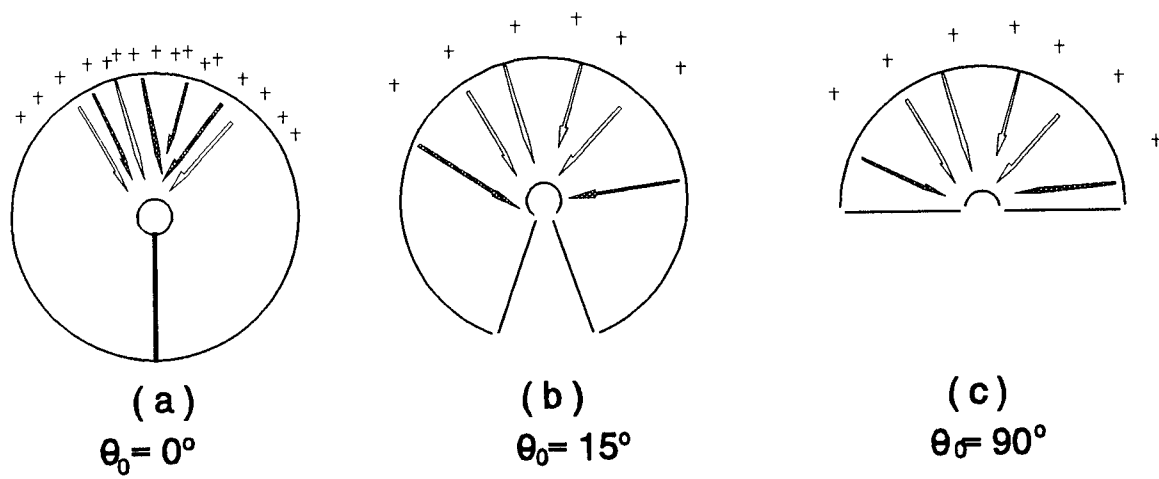
Fig4.9 Wedge Model with different Support Half Angle
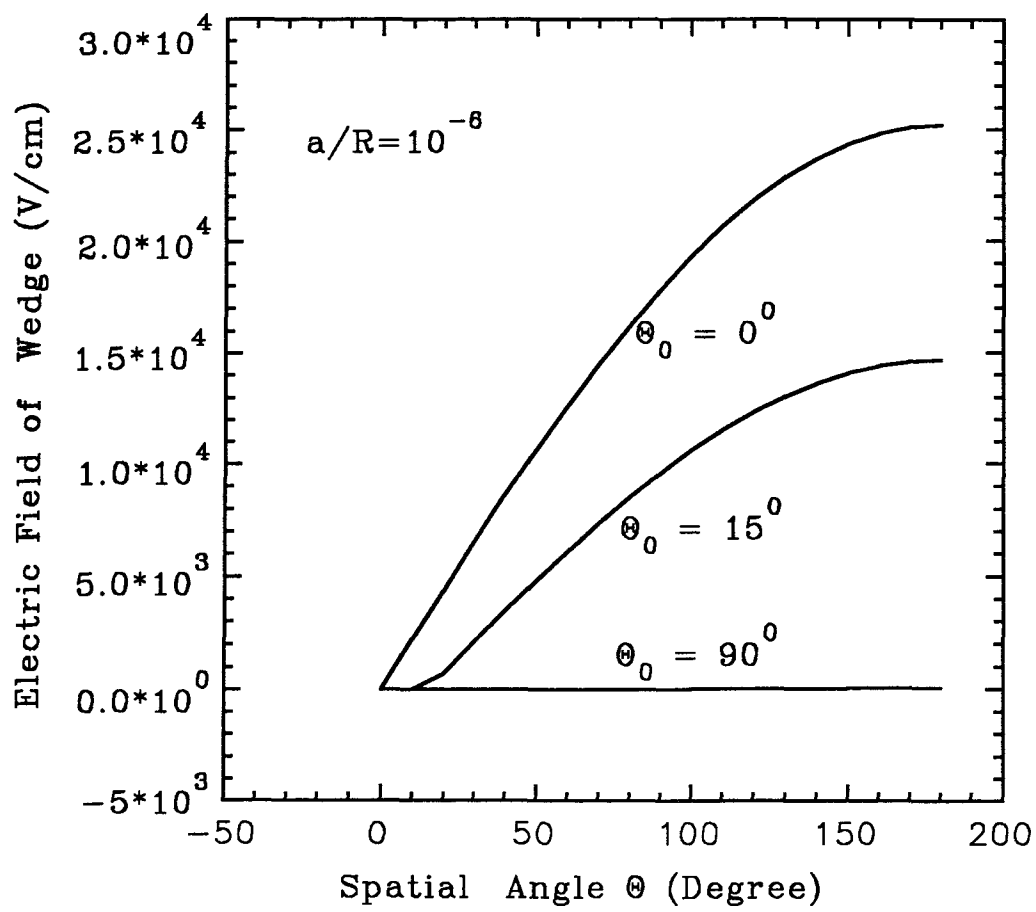
Fig. 4.10 Electric field of the emitter versus spatial angle
under different support half angle
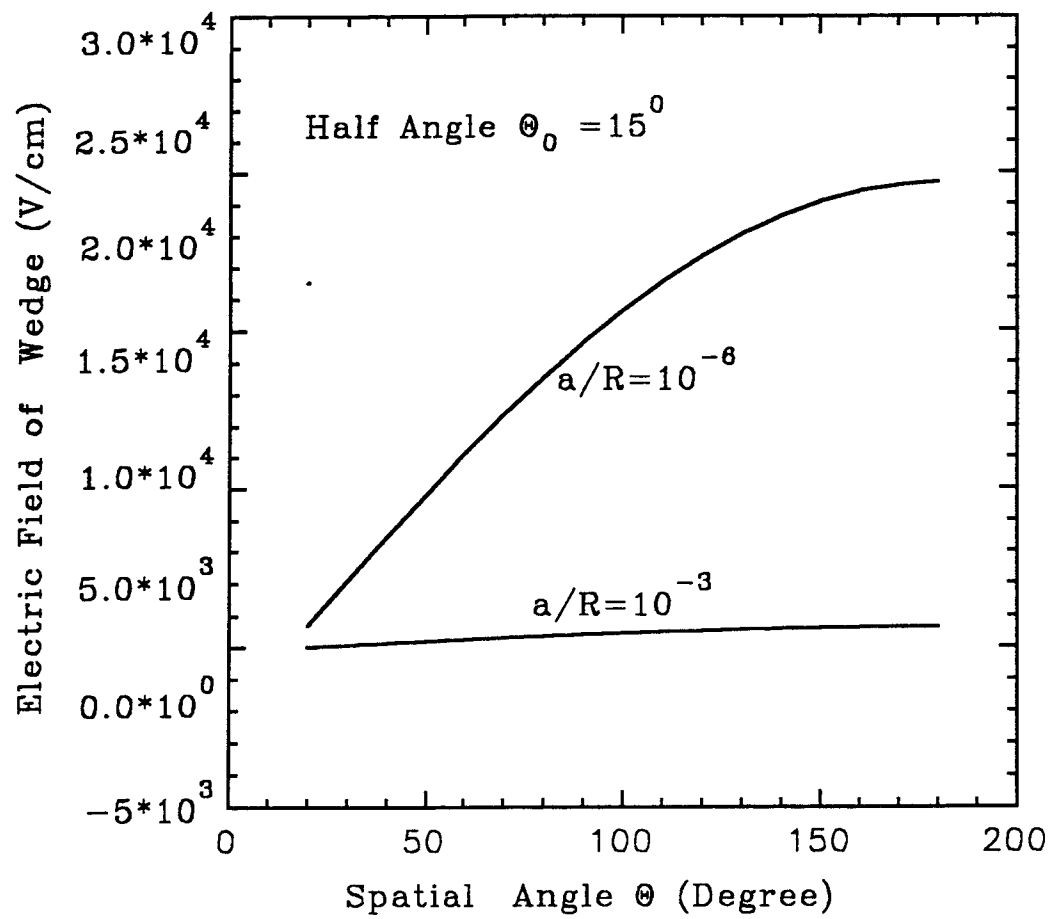
Fig. 4.11 Electric field of the emitter versus spatial angle
under different cathode radius
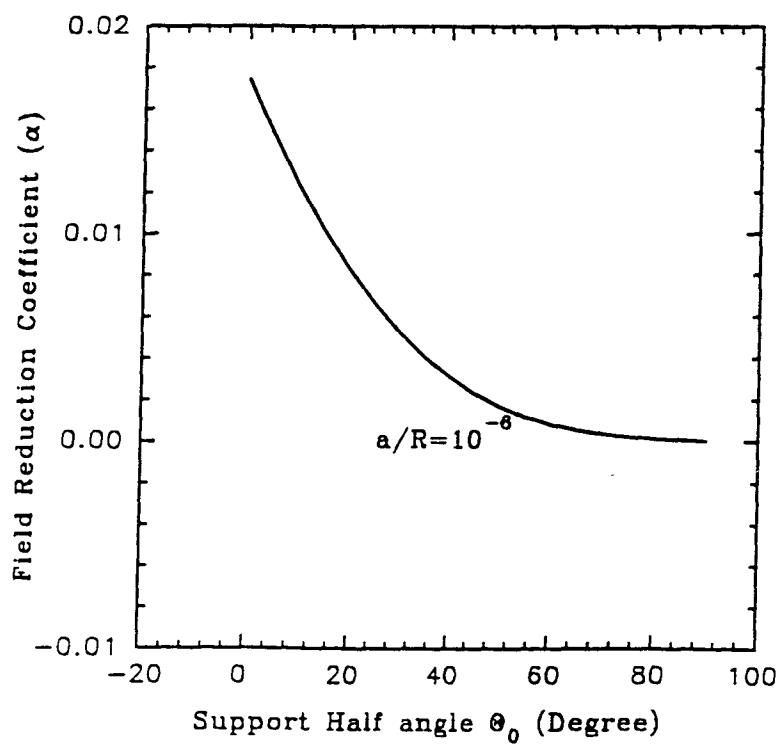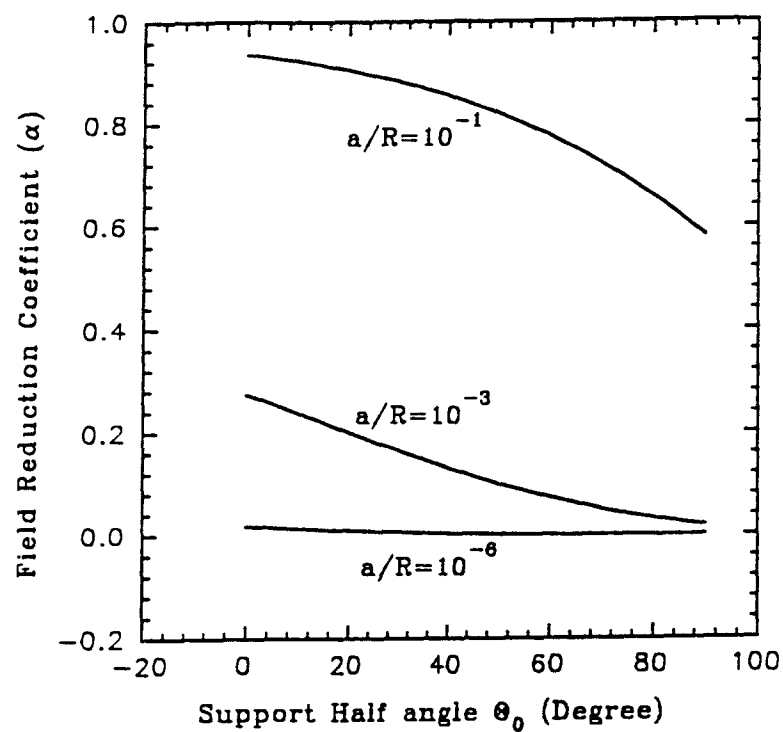
Fig. 4.12 Emission Reduct Coefficient versus Support Half Angle

Chapter 5
Conclusion and Summary

The analysis using computer simulation, as well as using theoretic model which was based on concentric cylinder connected by a support structure shows that electric field intensity over the surface of atomically sharp field emitter depends not only on the local curvature of the cathode, but also on global geometry of the emitter support structure.

The main results are as following:

1) Results from the modified concentric cylindrical emitter-anode structure show that for a wedge emitter with a half support angle of 15°, and $a/R = 10^{-6}$, the electric field drops off 10% from its maximum value when moving away from the center of the tip by 50°. The elctric field dropped off 50% when moving away from the center by 100°.

2) For a half angle of 15°, when the ratio of the cathode to the anode radius (a/R) changes from 10^{-6} to 10^{-3}, the electric field dropped 93%.

3) The support structure geometry has a stronger effect when the a/R ratio is small. The change of the emitter reduction coefficient with respect to the support from being vertical to being horizontal (half angle change from 0° − 90°) is from 0 01741 to 0.000035 when a/R is 10^{-6}, from 0.2753 to 0.0174, when a/R is 10^{-3}, 0.9363 to 0 5806, when a/R is 10^{-1}

The analysis of the electric field on a conical tip by computer simulation, which is based on a finite element method shows:

The electric field at the tip with $100V$ applied at the gate and $110V$ applied at the anode is $1\ 297 \times 10^{7} \frac{V}{cm}$ with a Spindt structure. The electric field is reduced by a factor of 1.54 compared to an isolated sphere cathode with radius $r = 0\ 05\mu m$ surrounded by a concentric sphere anode of radius $0.65\mu m$. The reduction coefficient calculated by ANSYS differs by only 3.5% compared with the value of Spindt's

calculation which is based on a relaxation method, and by 7.5% from Luo's result. which is based on a derived formula. This shows that the three methods lead to results in close agreement.

The electric field for the Spindt's geometry was studied as a function of various geometry parameters and the following results have been found.

1) The reduction coefficient as a function of gate electrode length gate weekly increases at a rate of 6.45×10^{-6} (nm^{-1}) over the entire range $3000 - 20,000nm$.

2) The electric field on the surface drops off by 10% when moving away the spatial angle reaches 50°.

3) The field reduction coefficient shows a monotonic decrease of 0.0537 per 10° change of the support half angle. Thus with half angle change of 10°, the change of the tip electric field or reduction coefficient, is about 5%.

4) The dependence of the field reduction coefficient on the support height is saturated when the height is greater than $1500nm$. Thus making tip higher will not increase the electric field or electron emission at the tip.

5) The field reduction coefficient decreased with the gate opening of gate electrode R at a rate of $4 \, 8 \times 10^{-3}$ (nm^{-1}). This result indicates that R should made as small as possible to get the largest electric field.

6) The electric field at the center of the emitter tip with radius of $1nm$ is 6 times the value of the electric field with a radius of $100nm$. The electric field of a sharp tip with a radius $1nm$ is calculated to be $10.354 \times 10^{7} \frac{V}{cm}$.

The accuracy of the simulation was studied by changing the mesh size in the finite element simulation. It is found that the field reduction coefficient changes with mesh size at a slope of only 9.68×10^{-3} (nm^{-1}) over the entire range of mesh size $1 - 15nm$. The smallest mesh size used in the simulation was $1nm$, for this size the percentage error from the idea mesh size of 0 is estimated to be 1.4%. This is proved to be accurate enough when comparing the result from the ANSYS with some other calculation. With the help of supercomputer and ANSYS commercial

version, the small mesh size of less than $1nm$ can be used, accuracy will be improved in the future simulation.

Appendice A

Electric Field and Field Reduction Coefficient

This is result of Electric field and reduction coefficient alpha due to support at the top of tip when a/R=1e-6 versus support half angle theta0

```
p =    1 a   :   0.000005
cita = 180
a = 0.000005
R = 5.000000
V= 100.000000
```

theta0	E	phi	alpha
0	25212.976716	0.001267	0.017417
5	21288.477604	0.001070	0.014706
10	17781.883852	0.000894	0.012283
15	14678.660727	0.000738	0.010140
20	11961.291459	0.000601	0.008263
25	9609.390352	0.000483	0.006638
30	7599.897479	0.000382	0.005250
35	5907.362238	0.000297	0.004081
40	4504.320090	0.000226	0.003111
45	3361.762589	0.000169	0.002322
50	2449.695200	0.000123	0.001692
55	1737.770359	0.000087	0.001200
60	1195.974997	0.000060	0.000826
65	795.342897	0.000040	0.000549
70	508.653516	0.000026	0.000351
75	311.071774	0.000016	0.000215
80	180.679412	0.000009	0.000125
85	98.849942	0.000005	0.000068
90	50.427826	0.000003	0.000035

This is result of Electric field and reduction coefficient alpha due to support at the top of tip when a/R=1e-1 versus support half angle theta0

```
p =    1 a    :   0.505000
cita = 180
a = 0.500000
R = 5.000000
V= 100.000000
```

theta0	E	phi	alpha
0	81.333308	0.408692	0.936384
5	80.752357	0.405772	0.929696
10	80.100519	0.402497	0.922191
15	79.369487	0.398823	0.913775
20	78.550088	0.394706	0.904341
25	77.632222	0.390093	0.893774
30	76.604806	0.384930	0.881945
35	75.455731	0.379156	0.868716
40	74.171835	0.372704	0.853935
45	72.738894	0.365503	0.837437
50	71.141661	0.357477	0.819049
55	69.363944	0.348544	0.798582
60	67.388764	0.338618	0.775842
65	65.198599	0.327612	0.750627
70	62.775775	0.315437	0.722733
75	60.103022	0.302006	0.691962
80	57.164277	0.287239	0.658128
85	53.945802	0.271066	0.621074
90	50.437718	0.253438	0.580686

This is result of Electric field and reduction coefficient alpha due to support at the top of tip when a/R=1e-3 versus support half angle theta0

```
     p =   1 a   :   0.005050
     cita = 180
     a = 0.005000
     R = 5.000000
     V=  100.000000
```

theta0	E	phi	alpha
0	797.305846	0.040064	0.275380
5	743.023066	0.037336	0.256631
10	688.990790	0.034621	0.237969
15	635.404504	0.031928	0.219461
20	582.476226	0.029269	0.201180
25	530.434049	0.026654	0.183205
30	479.521156	0.024095	0.165621
35	429.994189	0.021607	0.148515
40	382.120802	0.019201	0.131980
45	336.176260	0.016892	0.116111
50	292.438890	0.014695	0.101005
55	251.184238	0.012622	0.086756
60	212.677770	0.010687	0.073456
65	177.166046	0.008902	0.061191
70	144.866345	0.007279	0.050035
75	115.954919	0.005826	0.040049
80	90.554214	0.004550	0.031276
85	68.719767	0.003453	0.023735
90	50.427826	0.002534	0.017417

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle=0, a/R=1e-6)

```
p =    1 a
cita0 = 0
a = 0.000005
R = 5.000000
V=  100.000000
```

theta	E	phi
0	0.000633	0.000000
10	2197.464399	0.000110
20	4378.204373	0.000220
30	6525.622805	0.000328
40	8623.376230	0.000433
50	10655.499236	0.000535
60	12606.525986	0.000633
70	14461.607939	0.000727
80	16206.626852	0.000814
90	17828.302231	0.000896
100	19314.292390	0.000971
110	20653.288364	0.001038
120	21835.099953	0.001097
130	22850.733252	0.001148
140	23692.459075	0.001191
150	24353.871748	0.001224
160	24829.937837	0.001248
170	25117.034435	0.001262
180	25212.976716	0.001267
190	25117.034553	0.001262
200	24829.938072	0.001248
210	24353.872098	0.001224
220	23692.459537	0.001191
230	22850.733823	0.001148
240	21835.100629	0.001097
250	20653.289139	0.001038
260	19314.293259	0.000971
270	17828.303186	0.000896
280	16206.627887	0.000814
290	14461.609045	0.000727
300	12606.527156	0.000633
310	10655.500460	0.000535
320	8623.377500	0.000433
330	6525.624110	0.000328
340	4378.205703	0.000220
350	2197.465745	0.000110
360	0.001492	0.000000

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle = 15, a/R=1e-6)

```
p =    1 a
cita0 = 15
a = 0.000005
R = 5.000000
V=  100.000000
```

theta	E	phi
20	698.439581	0.000035
30	2088.993524	0.000105
40	3460.629094	0.000174
50	4800.924474	0.000241
60	6097.741688	0.000306
70	7339.336522	0.000369
80	8514.464889	0.000428
90	9612.484657	0.000483
100	10623.452026	0.000534
110	11538.211575	0.000580
120	12348.479171	0.000620
130	13046.916989	0.000656
140	13627.199953	0.000685
150	14084.073013	0.000708
160	14413.398728	0.000724
170	14612.194730	0.000734
180	14678.660727	0.000738
190	14612.194805	0.000734
200	14413.398877	0.000724
210	14084.073234	0.000708
220	13627.200245	0.000685
230	13046.917349	0.000656
240	12348.479596	0.000620
250	11538.212061	0.000580
260	10623.452569	0.000534
270	9612.485252	0.000483
280	8514.465529	0.000428
290	7339.337203	0.000369
300	6097.742403	0.000306
310	4800.925218	0.000241
320	3460.629858	0.000174
330	2088.994303	0.000105
340	698.440367	0.000035

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle=90, a/R=1e-6)

    p =   1 a
    cita0 = 90
    a = 0.000005
    R = 5.000000
    V=   100.000000

alpha	E	phi
90	0.000003	0.000000
100	8.756700	0.000000
110	17.247332	0.000001
120	25.213912	0.000001
130	32.414381	0.000002
140	38.629955	0.000002
150	43.671778	0.000002
160	47.386655	0.000002
170	49.661713	0.000002
180	50.427826	0.000003
190	49.661714	0.000002
200	47.386656	0.000002
210	43.671779	0.000002
220	38.629957	0.000002
230	32.414383	0.000002
240	25.213915	0.000001
250	17.247334	0.000001
260	8.756703	0.000000
270	0.000004	0.000000

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle=0, a/R=1e-3)

```
p =    1 a
cita0 = 0
a = 0.005000
R = 5.000000
V= 100.000000
```

theta	E	phi
0	0.020112	0.000000
10	69.765927	0.003506
20	138.988335	0.006984
30	207.128713	0.010408
40	273.658445	0.013751
50	338.063524	0.016987
60	399.848992	0.020092
70	458.543025	0.023041
80	513.700629	0.025813
90	564.906905	0.028386
100	611.779868	0.030741
110	653.972816	0.032861
120	691.176284	0.034731
130	723.119609	0.036336
140	749.572131	0.037665
150	770.344119	0.038709
160	785.287418	0.039460
170	794.295919	0.039912
180	797.305846	0.040064
190	794.295923	0.039912
200	785.287426	0.039460
210	770.344130	0.038709
220	749.572146	0.037665
230	723.119626	0.036336
240	691.176306	0.034731
250	653.972840	0.032861
260	611.779896	0.030741
270	564.906936	0.028386
280	513.700661	0.025813
290	458.543060	0.023041
300	399.849029	0.020092
310	338.063563	0.016987
320	273.658485	0.013751
330	207.128755	0.010408
340	138.988378	0.006984
350	69.765970	0.003506
360	0.020112	0.000000

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle=15, a/R=1e-3)

```
     p =    1 a
     cita0 = 15
     a = 0.005000
     R = 5.000000
     V= 100.000000
```

theta	E	phi
20	30.298203	0.001522
30	90.616729	0.004553
40	150.104455	0.007543
50	208.216795	0.010463
60	264.422925	0.013287
70	318.210952	0.015990
80	369.092788	0.018546
90	416.608647	0.020934
100	460.331120	0.023131
110	499.868812	0.025118
120	534.869516	0.026876
130	565.022923	0.028392
140	590.062895	0.029650
150	609.769317	0.030640
160	623.969550	0.031354
170	632.539524	0.031784
180	635.404504	0.031928
190	632.539528	0.031784
200	623.969556	0.031354
210	609.769327	0.030640
220	590.062908	0.029650
230	565.022938	0.028392
240	534.869534	0.026876
250	499.868833	0.025118
260	460.331143	0.023131
270	416.608673	0.020934
280	369.092816	0.018546
290	318.210981	0.015990
300	264.422956	0.013287
310	208.216828	0.010463
320	150.104488	0.007543
330	90.616762	0.004553
340	30.298237	0.001522

This is the result of the Electric field versus spatial angle theta, from the result, we con see the area Electric field concentrated (Support half angle=90, a/R=1e-3)

    p =   1 a
    cita0 = 90
    a = 0.005000
    R = 5.000000
    V= 100.000000

theta	E	phi
90	0.002534	0.000000
100	8.756734	0.000440
110	17.247393	0.000867
120	25.213988	0.001267
130	32.414457	0.001629
140	38.630019	0.001941
150	43.671821	0.002194
160	47.386677	0.002381
170	49.661719	0.002495
180	50.427826	0.002534
190	49.661720	0.002495
200	47.386678	0.002381
210	43.671823	0.002194
220	38.630021	0.001941
230	32.414459	0.001629
240	25.213991	0.001267
250	17.247396	0.000867
260	8.756737	0.000440
270	0.002534	0.000000

Appendice B

Source Code of Wedge Calculation on the Window System

```
/****************************************************************
PROGRAM: Wele.c
    PURPOSE: A Windows applications for Simulation Result
    FUNCTIONS:
     WinMain() - calls initialization function, processes
message loop
     InitApplication() - initializes window data and
registers window
     InitInstance() - saves instance handle and creates main
window
     MainWndProc() - processes messages
     About() - processes messages for "About" dialog box
     ComputeP() - Compute Voltage
     ComputeE() - Compute Isopotential Line
****************************************************************/

#include <windows.h>
#include <string.h>
#include <math.h>
#include "wele.h"

HANDLE hInst;
HPEN hBoldPen;
HPEN hRedPen;
HPEN hGreenPen;
HBRUSH hOldBrush;
HBRUSH hRedBrush;
HBRUSH hGreenBrush;
HBRUSH hBlueBrush;
HCURSOR hSaveCursor;
HCURSOR hHourGlass;

static int xy[641][9];
static int xyb[5];
static int miny = 0;
static double phi[361][5];
static double maxY[5];
static int draw = 0;
/****************************************************************

     FUNCTION: WinMain(HANDLE, HANDLE, LPSTR, int)
```

```
        PURPOSE: calls initialization function, processes
message loop

        ************************************************************
***************/

int PASCAL WinMain(hInstance, hPrevInstance, lpCmdLine,
nCmdShow)
HANDLE hInstance;
HANDLE hPrevInstance;
LPSTR lpCmdLine;
int nCmdShow;
{
    MSG msg;
    if (!hPrevInstance)
     if (!InitApplication(hInstance))
          return (FALSE);
    if (!InitInstance(hInstance, nCmdShow))
     return (FALSE);
    while (GetMessage(&msg, NULL, NULL, NULL)) {
     TranslateMessage(&msg);
     DispatchMessage(&msg);
    }
    return (msg.wParam);
}

/************************************************************
***************

    FUNCTION: InitApplication(HANDLE)

    PURPOSE: Initializes window data and registers window
class

************************************************************
***************/

BOOL InitApplication(hInstance)
HANDLE hInstance;
{
    WNDCLASS  wc;
    wc.style = NULL;
    wc.lpfnWndProc = MainWndProc;
    wc.cbClsExtra = 0;
    wc.cbWndExtra = 0;
    wc.hInstance = hInstance;
    wc.hIcon = LoadIcon(NULL, IDI_APPLICATION);
    wc.hCursor = LoadCursor(NULL, IDC_ARROW);
    wc.hbrBackground = GetStockObject(WHITE_BRUSH);
    wc.lpszMenuName =  "OutputMenu";
    wc.lpszClassName = "OutputWClass";
    return (RegisterClass(&wc));
}
```

```
/********************************************************
****************

     FUNCTION:   InitInstance(HANDLE, int)

     PURPOSE:   Saves instance handle and creates main window

********************************************************
***************/

BOOL InitInstance(hInstance, nCmdShow)
     HANDLE            hInstance;
     int               nCmdShow;
{
     HWND              hWnd;
     hInst = hInstance;
     hWnd = CreateWindow(
       "OutputWClass",
       "SIMULATION RESULTS WINDOW",
       WS_OVERLAPPEDWINDOW,
       CW_USEDEFAULT,
       CW_USEDEFAULT,
       CW_USEDEFAULT,
       CW_USEDEFAULT,
       NULL,
       NULL,
       hInstance,
       NULL
     );
     if (!hWnd)
       return (FALSE);
     ShowWindow(hWnd, nCmdShow);
     UpdateWindow(hWnd);
     return (TRUE);
}

/********************************************************
****************

     FUNCTION: MainWndProc(HWND, unsigned, WORD, LONG)

     PURPOSE:   Processes messages

     MESSAGES:

        WM_COMMAND      - application menu (About dialog box)
        WM_CREATE       - create window and objects
        WM_PAINT        - update window, draw objects
        WM_DESTROY      - destroy window
        WM_COMP0        - compute voltage for cita0 = 0
        WM_COMP15       - compute voltage for cita0 = 15
        WM_COMPE        - compute isopotential line
        WM_DRAWP        - draw voltage
```

```
        WM_DRAWE          - draw isopotential line

****************************************************************
***************/

long FAR PASCAL MainWndProc(hWnd, message, wParam, lParam)
HWND hWnd;
unsigned message;
WORD wParam;
LONG lParam;
{
    FARPROC lpProcAbout;

    HDC hDC;                             /* display-context
variable  */
    PAINTSTRUCT ps;                      /* paint structure
*/
    RECT rcTextBox;                      /* rectangle around
the text */
    HPEN hOldPen;                        /* old pen handle
*/

    switch (message) {
     case WM_COMMAND:
       switch( wParam ) {
           case IDM_ABOUT:
           lpProcAbout = MakeProcInstance(About, hInst);
           DialogBox(hInst,
               "AboutBox",
               hWnd,
               lpProcAbout);
           FreeProcInstance(lpProcAbout);
           break;
            case IDM_DRAW1:
           draw = 1;
           InvalidateRect (hWnd, NULL, TRUE);
           break;
            case IDM_DRAW2:
           draw = 2;
           InvalidateRect (hWnd, NULL, TRUE);
           break;
            case IDM_COMP0:
           SetCapture(hWnd);
           hSaveCursor = SetCursor( hHourGlass );
           Computation(0, 1, 0);
           Computation(0, 2, 1);
           Computation(0, 10, 2);
           Computation(0, 50, 3);
           SetCursor( hSaveCursor );
           ReleaseCapture();
           break;
            case IDM_COMP15:
           SetCapture(hWnd);
           hSaveCursor = SetCursor( hHourGlass );
```

```
                    Computation(15,  1,  0);
                    Computation(15,  2,  1);
                    Computation(15,  10,  2);
                    Computation(15,  50,  3);
                    SetCursor( hSaveCursor );
                    ReleaseCapture();
                    break;
                      case IDM_COMPE15:
                    SetCapture(hWnd);
                    hSaveCursor = SetCursor( hHourGlass );
                    ComputationE(15,  0,       0);
                    ComputationE(15,  0.01,  1);
                    ComputationE(15,  0.02,  2);
                    ComputationE(15,  0.03,  3);
                    ComputationE(15,  0.04,  4);
                    ComputationE(15,  0.05,  5);
                    ComputationE(15,  0.08,  6);
                    ComputationE(15,  0.12,  7);
                    //    ComputationE(15,  0.15,  8);

                    SetCursor( hSaveCursor );
                    ReleaseCapture();
                    break;
                      case IDM_COMPE0:
                    SetCapture(hWnd);
                    hSaveCursor = SetCursor( hHourGlass );
                    ComputationE(0,  1,  0);
                    ComputationE(0,  2,  1);
                    ComputationE(0,  5,  2);
                    ComputationE(0,  10,  3);
                    SetCursor( hSaveCursor );
                    ReleaseCapture();
                    break;
                      default:
                    return (DefWindowProc(hWnd,  message,  wParam,
lParam));
                }
        case WM_CREATE:
            hRedBrush =    CreateSolidBrush(RGB(255,    0,    0));
            hGreenBrush = CreateSolidBrush(RGB(  0,  255,    0));
            hBlueBrush =  CreateSolidBrush(RGB(  0,    0,  255));
            hBoldPen = CreatePen(PS_SOLID, 1, RGB(0, 0, 0));
/* color */
            hRedPen = CreatePen(PS_SOLID, 1, RGB(255, 0, 0));
/* color */
            hGreenPen = CreatePen(PS_DASHDOTDOT, 1, RGB(0, 255,
0));
            hHourGlass = LoadCursor(NULL,  IDC_WAIT);
            break;
        case WM_SIZE:
            if (lParam){
              /* invalidate it's client area.*/
              InvalidateRect (hWnd, NULL, TRUE);
            }
```

```
                break;
            case WM_PAINT:
                HandlePaint (hWnd);
                break;
            case WM_DESTROY:
                DeleteObject(hRedBrush);
                DeleteObject(hGreenBrush);
                DeleteObject(hBlueBrush);
                DeleteObject(hBoldPen);
                DeleteObject(hRedPen);
                DeleteObject(hGreenPen);
                PostQuitMessage(0);
                break;
            default:
                return (DefWindowProc(hWnd, message, wParam,
lParam));
        }
    return (NULL);
};


/****************************************************
****************

    FUNCTION: About(HWND, unsigned, WORD, LONG)

    PURPOSE:  Processes messages for "About" dialog box

    MESSAGES:

     WM_INITDIALOG - initialize dialog box
     WM_COMMAND    - Input received

*****************************************************
**************/

BOOL FAR PASCAL About(hDlg, message, wParam, lParam)
HWND hDlg;
unsigned message;
WORD wParam;
LONG lParam;
{
    switch (message) {
      case WM_INITDIALOG:
            return (TRUE);
      case WM_COMMAND:
            if (wParam == IDOK
             || wParam == IDCANCEL) {
             EndDialog(hDlg, TRUE);
             return (TRUE);
            }
            break;
    }
    return (FALSE);
```

```
}
/**************************************************
****************

     FUNCTION: HandlePaint(HANDLE)

     PURPOSE: draw simulation results

**************************************************
***************/

void FAR PASCAL HandlePaint (hwnd)
HWND hwnd;
{
    HDC hDC;
    PAINTSTRUCT ps;
    RECT rcTextBox;
    HPEN hOldPen;
    RECT rc;
    TEXTMETRIC textmetric;
    int nDrawX1;
    int nDrawY1;
    int nDrawX2;
    int nDrawY2;
    int d,j;
    int offset;
    double uX, uY;
    char szText[80];

    offset = 30;
    hDC = BeginPaint(hwnd, (LPPAINTSTRUCT)&ps);
    GetClientRect(hwnd, (LPRECT)&rc);
    DrawText (hDC,
            "Title",
              5,
              (LPRECT)&rc,
              DT_CENTER | DT_WORDBREAK);
    if(draw == 1)
    {
      int ty = 40;
      DrawText (hDC,
            "NORMALIZED POTENTIAL",
              20,
              (LPRECT)&rc,
              DT_CENTER | DT_WORDBREAK);
    uX = (rc.right-2.0*offset-10)/360.0;
    uY = 1.;

    hOldPen = SelectObject(hDC, hBoldPen);

    MoveTo(hDC, offset, rc.bottom-offset);
    LineTo(hDC, rc.right-offset, rc.bottom-offset); // X
    LineTo(hDC, rc.right-offset-3, rc.bottom-offset-3);
```

```
      MoveTo(hDC, rc.right-offset, rc.bottom-offset);
      LineTo(hDC, rc.right-offset-3, rc.bottom-offset+3);

      MoveTo(hDC, offset, rc.bottom-offset);              // Y
      LineTo(hDC, offset, offset);
      LineTo(hDC, offset-3, offset+3);
      MoveTo(hDC, offset, offset);
      LineTo(hDC, offset+3, offset+3);

      for(j = 10; j < 360; j+=20)            // X
      {
        MoveTo(hDC, offset+j*uX, rc.bottom-offset);
        LineTo(hDC, offset+j*uX, rc.bottom-offset-2);
        MoveTo(hDC, offset+(j+10)*uX, rc.bottom-offset);
        LineTo(hDC, offset+(j+10)*uX, rc.bottom-offset-4);
      }
      for(j = 20; j < rc.bottom-2*offset-20; j+=40)          //
Y
      {
        MoveTo(hDC, offset,    rc.bottom-offset-j);
        LineTo(hDC, offset+2, rc.bottom-offset-j);
        MoveTo(hDC, offset,    rc.bottom-offset-j-20);
        LineTo(hDC, offset+4, rc.bottom-offset-j-20);
      }
      for(j = 1; j < 4; j++)
      {
       SelectObject(hDC, hRedPen);
       nDrawX2 = offset ;
       nDrawY2 = rc.bottom-offset -(int) (phi[0][j]*uY);
       for(d=0; d < 360; d += 1)
       {
              nDrawX1 = nDrawX2;
              nDrawY1 = nDrawY2;
              nDrawX2 = (int)((d+1)*uX) + offset ;
              nDrawY2 = rc.bottom-offset -(int)
(phi[d+1][j]*400);
              if((nDrawY1 <= rc.bottom-offset) &&
              (nDrawY2 <= rc.bottom-offset) )
              {
              MoveTo(hDC, nDrawX1, nDrawY1);
              LineTo(hDC, nDrawX2, nDrawY2);
              }
       }
      SelectObject(hDC, hGreenPen);
      MoveTo(hDC, offset, rc.bottom-offset-
(int)(maxY[j]*400));
      LineTo(hDC, offset + (int)(180*uX), rc.bottom-offset-
(int)(maxY[j]*400));
      }
  }
  else if( draw == 2 )
  {
    DrawText (hDC,
         "ISOPOTENTIAL LINE for WEDGE",
```

```
                    27,
                    (LPRECT)&rc,
                    DT_CENTER | DT_WORDBREAK);

        hOldPen = SelectObject(hDC, hBoldPen);
        MoveTo(hDC, offset, rc.bottom/2);
        LineTo(hDC, rc.right-offset, rc.bottom/2); // X
        LineTo(hDC, rc.right-offset-3, rc.bottom/2-3);
        MoveTo(hDC, rc.right-offset, rc.bottom/2);
        LineTo(hDC, rc.right-offset-3, rc.bottom/2+3);

        MoveTo(hDC, rc.right/2, rc.bottom-offset);              // Y
        LineTo(hDC, rc.right/2, offset);
        LineTo(hDC, rc.right/2-3, offset+3);
        MoveTo(hDC, rc.right/2, offset);
        LineTo(hDC, rc.right/2+3, offset+3);

        for(j = 10; j < rc.right/2-offset-10; j+=20)           //
X
        {
          MoveTo(hDC, rc.right/2+j    , rc.bottom/2);
          LineTo(hDC, rc.right/2+j    , rc.bottom/2-2);
          MoveTo(hDC, rc.right/2+j+10, rc.bottom/2);
          LineTo(hDC, rc.right/2+j+10, rc.bottom/2-4);
          MoveTo(hDC, rc.right/2-j    , rc.bottom/2);
          LineTo(hDC, rc.right/2-j    , rc.bottom/2-2);
          MoveTo(hDC, rc.right/2-j-10, rc.bottom/2);
          LineTo(hDC, rc.right/2-j-10, rc.bottom/2-4);
        }
        for(j = 10; j < rc.bottom/2-offset-10; j+=20)          //
Y
        {
          MoveTo(hDC, rc.right/2    , rc.bottom/2-j);
          LineTo(hDC, rc.right/2+2, rc.bottom/2-j);
          MoveTo(hDC, rc.right/2    , rc.bottom/2-j-10);
          LineTo(hDC, rc.right/2+4, rc.bottom/2-j-10);

          MoveTo(hDC, rc.right/2    , rc.bottom/2+j);
          LineTo(hDC, rc.right/2+2, rc.bottom/2+j);
          MoveTo(hDC, rc.right/2    , rc.bottom/2+j+10);
          LineTo(hDC, rc.right/2+4, rc.bottom/2+j+10);
        }

        SelectObject(hDC, hRedPen);
        for(j = 0; j < 9 ; j++)
        {
         if( xyb[j] >= 0 ) continue;
//            if( j % 2 )
//                 SelectObject(hDC, hRedPen);
//            else
//                 SelectObject(hDC, hGreenPen);
         nDrawX2 = xyb[j]+rc.right/2;
         nDrawY2 = rc.bottom/2 - (xy[0][j]);
         for(d=0; d < 640; d += 1)
```

```
        {
                if ( xy[d+1][j] < 0) break;
                nDrawX1 = nDrawX2;
                nDrawY1 = nDrawY2;
                nDrawX2 = xyb[j]+rc.right/2+d+1;
                nDrawY2 = rc.bottom/2 - xy[d+1][j];
                MoveTo(hDC, nDrawX1, nDrawY1);
                LineTo(hDC, nDrawX2, nDrawY2);
                MoveTo(hDC, nDrawX1, nDrawY1+2*xy[d][j]);
                LineTo(hDC, nDrawX2, nDrawY2+2*xy[d+1][j]);
        }
     }
   }
   SelectObject(hDC, hOldPen);
   EndPaint(hwnd, (LPPAINTSTRUCT)&ps);
}


/*****************************************************
****************

    FUNCTION: Compute(int, int, int)

    PURPOSE: Compute Voltage

*****************************************************
**************/

void FAR PASCAL Computation(icita0,t,j)
int icita0;
int t;
int j;
{
    double p,a,R,V,pi,logpa,logRa,cita0;
    int n,d;
    maxY[j] = 0;

    a = 5e-6;
    if( t == 1 )
     p = 1.01*a;
    else
     p = t*a;
    R = 5.;
    V = 100;
    pi = 3.1415926;
    logpa = log(p/a);
    logRa = log(R/a);
    cita0 = (double)icita0*pi/180;
    for(d=0; d <= 180; d += 1)
    {
     double cita, cici0,tphi,Ep,Eo,k,E;
     cita = d*pi/180;
     cici0 = cita-cita0;
     tphi = 0;
```

```
        Ep = 0;
        Eo = 0;
        for(n = 1; n < 10; n += 2)
        {
            k = 0.5 * n * pi / ( pi - cita0 );
            tphi += sinh(k*logpa)/sinh(k*logRa)*sin(k*cici0)/n;
        };
        phi[d][j] = tphi * 4 * V / pi;
        phi[360-d][j] = phi[d][j];
        if( phi[d][j] > maxY[j] )  maxY[j] = phi[d][j];
    }
}


/*****************************************************************
****************

    FUNCTION: ComputationE(double, double, int)

    PURPOSE: Compute Isopotential Line

*****************************************************************
***************/

void FAR PASCAL ComputationE(cita0, cphi, j)
double cita0;
double cphi;
int j;
{
    double a,R,V,pi,e,logRa, loge;
    double tphi,k;
    double cita, p;
    int x,y;
    int i,n;
    double t;
    t = 1.25e-7;
    a = 5e-6;
    e = 2.718281828;
    loge = log(e);
    R = 5.;
    V = 100;
    pi = 3.1415926;
    logRa = log(R/a);
    cita0 = cita0*pi/180;
    x = (int)(-a/t)+1;
    y = 0;
    tphi = 0;
    while( tphi <= cphi )
    {
     x--;
     p = abs(x)*t;
     cita = pi;
     tphi = 0;
     for(n = 1; n < 10; n += 2)
```

```
     {
         k = 0.5 * n * pi / ( pi - cita0 );
         tphi += sinh(k*log(p/a))/sinh(k*logRa)*sin(k*(cita-
cita0))/n;
     };
     tphi = tphi * 4 * V / pi;
  }
  xy[0][j] = 0;
  xyb[j] = x;
  i = 0;
  while( ++i < 640 )
  {
    double ttt,dx;
    x++;
    dx = (double) x;
    ttt=  p/t*p/t -dx*dx ;
    if ( ttt <= 0 )
   y = 0;
    else
    {
   y = (int)sqrt( ttt );
    }
    y--;
    tphi = 0;
    while( tphi <= cphi )
    {
        double dx,dy;
        y++;
        dx = (double)x;
        dy = (double)y;
        p = sqrt(dx*dx+dy*dy)*t;
        cita = acos(x/p*t);
        if( cita < cita0 ) break;
        tphi = 0;
        for(n = 1; n < 10; n += 2)
        {
         k = 0.5 * n * pi / ( pi - cita0 );
         tphi +=
sinh(k*log(p/a))/sinh(k*logRa)*sin(k*(cita-cita0))/n;
        };
        tphi = tphi * 4 * V / pi;
    }
    xy[i][j] = y;

    if( ( xyb[j] + i > 320) || (xy[i][j] > 240) || cita <=
cita0)
       break;
  }
  for(;i< 641;i++)
    if( cita <= cita0 )
    {
    xy[i][j] = (int)((i+xyb[j])*tan(cita0));
    }
    else
```

```
        xy[i][j] = -1;
}
```

Appendice C

Source Code of the Head File and Defination File

```
NAME            WELE

DESCRIPTION     'SIMULATION RESULTS'

EXETYPE         WINDOWS

STUB            'WINSTUB.EXE'

CODE            PRELOAD MOVEABLE DISCARDABLE
DATA            PRELOAD MOVEABLE MULTIPLE

HEAPSIZE        1024
STACKSIZE       5120

EXPORTS
    MainWndProc     @1
    About           @2
```

Appendice D

Source Code of Current Calculation

```c
#include <string.h>
#include <math.h>

void main()
{
    double da[6],da1[6],j[6],j1[6],e[6],e1[6],inter,t;
    double r,a,b,pi,phi,t2y,temp,slope,sumi,sumi1;
    int i;
    r=50.0*(1e-7);
    a=1.54*(1e-6);
    b=6.87*(1e+7);
    pi=3.14159;
    phi=4.5;
    t2y=1.1;
    e[0]=0.99388*(1e+7);
    e[1]=0.98148*(1e+7);
    e[2]=0.96342*(1e+7);
    e[3]=0.93552*(1e+7);
    e[4]=0.89334*(1e+7);
    e[5]=0.83084*(1e+7);
    e1[0]=1.2970*(1e+7);
    e1[1]=1.2960*(1e+7);
    e1[2]=1.2815*(1e+7);
    e1[3]=1.2537*(1e+7);
    e1[4]=1.2068*(1e+7);
    e1[5]=1.1320*(1e+7);

    temp = exp(b*1.44*(1e-7)/sqrt(phi));
    inter = a*exp(b*1.44*(1e-7)/sqrt(phi))/(phi*t2y);
    slope = b*sqrt(phi*phi*phi);
    for(i=0; i<6; i++)
    {
     j[i]=e[i]*e[i]*inter*exp(-0.95*slope/e[i]);
     j1[i]=e1[i]*e1[i]*inter*exp(-0.95*slope/e1[i]);
    }
    for(i=0; i<6; i++)
    {
     da[i]=2.*pi*5.8*r/(36*5.)*(1e-7);
     da1[i]=2.*pi*sin((5.8+i*11.6)*2*pi/360)*r*(2.*pi*5.8*r/
(36*5.));
    }
```

```
     sumi=0.;
     sumi1=0.;
     for(i=0; i<6; i++)
     {
      sumi1 += 2*da1[i]*j1[i];
      sumi += 2*da[i]*j[i];
//          printf("sumi = %e    sumi1 = %e\n",sumi,sumi1);
//          printf("sumi = %25.20f    sumi1 =
%25.20f\n",sumi,sumi1);
     }
     printf("    n              Cone Current Dencity      Wedge
Current Dencity\n");

printf("=================================================
============\n");
     for(i=0; i<6; i++)
     {
      printf("  %2d               %e
%e\n",i,j1[i],j[i]);
//          printf("j=%e    j1=%e    da=%e
da1=%e\n",j[i],j1[i],da[i],da1[i]);
//          printf("j=%20.15f j1=%20.5f da=%25.15f
da1=%25.15f\n",j[i],j1[i],da[i],da1[i]);
     }
     printf("----------------------------------------------
-----------------\n");
     printf("  TOTAL CURRENT:   %e                %e\n",sumi1,
sumi);
}
```

Bibliography

[1] Marcus, R. B., Ravi, T. S., Chin, K. "Formation of silion tips with $< 1nm$ radius." *Appl. Phys. Lett.* , Vol. 56, No. 3, (1990): pp. 236-238.

[2] Marcus, R. B., Ravi, T. S., Chin, K. "Fabrication of wedge-shaped silicon field emitters with nm-scale radii." *Appl. Phys. Lett.* , Vol. 58, No. 10, (1991): pp. 1042-1043.

[3] Greene, R., Gray, H., and Campisi, G. *Proc. IEDM*, Vol 85, 172(1986).

[4] Yuan, Y., Chin, K. K. "Simulation and Design of Field Emitter." *IEEE Trans. Electron. Dev.*, Vol. 37, 1545(1990): pp. 295-253.

[5] Dyke, W. P., Dolan, W. W. "Field Emission." (1990): 192-198.

[6] Luo, E., Liu, Y.,Huang, W. " Field emission." *Vacumm microelectronics 89*, Paper presented at 2nd Int. Conf. on Vac. Microelectron., Bath, (1989).

[7] Spindt, C. A., Brodie, I. "Physical properties of thin-film field emission cathodes with molydenum cones." *Journal of Applied Physics*, Vol. 47, No. 12, (1976): pp. 5248-5263.

[8] Fowler, R. H. and Nordheim, L. *Proc. R. Soc.*, London A119, 173 (1928).

[9] Spindt, C. A., Holland, C. E., and Rosengreen, A. *Tech. Digest of Intern. Vacuum Microel Conference*, Nagahama, 44 (1991).

[10] Cade, N. *Vacumm Microelectronics*, Iop Publishing, Ltd., Bristol, 109(1989).

[11] Feeney, R. F., Cochran, J.K., and Hill, D.N. *Vacumm Microelectronics*, Iop Publishing, Ltd., Bristol, 117 (1989).

[12] Norrie, D. H. and Vries, G. "An Introduction to Finite Element Analysis." *Academic Press*, (1978).

[13] Lin W. "Microwave Theory and Technique." *Science and Technology Publisher* Bejing, China, (1978).

[14] Dalies, A. J. "The Finite Element Method: A First Approach." *Clarendon Press, Oxford*, (1980).

[15] Schwarz, H. R. "Finite Element Methods." *Academic Press*, (1988).

[16] Fenner, R. T. "Finite Element Methods for Engineers." (1975).

[17] Csendes, Z. J. "A Note on the Finite -element solution of exterior field problem." *IEEE Trans.*, Vol. MTT-24(1976): pp. 468-472.

[18] "ANSYS Primer for Thermal Analysis." Revision 4.4, Swanson Analysis System, Inc.

[19] Jackson, J. D. "Classic Electrodymics." *John Wiley & Sons, Inc.*, Second Edition.

[20] "Microsoft windows programming guide." *Microsoft, version 3*, (1990).