In this section of *Resonance,* we invite readers to pose questions likely to be raised in a classroom situation. We may suggest strategies for dealing with them, or invite responses, or both. “Classroom” is equally a forum for raising broader issues and sharing personal experiences and viewpoints on matters related to teaching and learning science.

Simulation of Electron Motion in Fields – An Interactive Teaching Aid

S W Anwane and N S Choubey#
Department of Physics
Shivaji Education Society
Amravati's Science College
Congress Nagar,
Nagpur 440 012, India
Email:
swanwane@rediffmail.com
#Department of Computer Science
Anuradha Engineering College
Chikhli Dist
Buldana 443201, India.

1. Introduction

While teaching the behaviour of a uniformly moving beam of electrons in electric and/or magnetic field, a mere theoretical discussion does not impart a complete understanding. The present work aims at raising interest and interaction, developing intuition to understand this concept. The proposed programme is developed in the C language. The input parameters and conditions/situations are to be defined by students to see the quantitative effects displayed on the screen matched to the scale. The facility to vary input parameters (keyed every time while running the programme) will make it interesting and interactive for students to learn the effect of these parameters, as well as useful to visualize fabrication parameters in related devices (CRT, CRT for Thomson's *e/m* technique, etc.) for design engineers. Moreover, the principal mechanism in measuring *e/m* in Thomson's method can be explored.

2. Motion of Uniformly Moving Beam of Electrons in Fields

Keywords
Charged particle in electric and magnetic field.

The discussion of motion of a uniformly moving beam of electrons in transverse electric and magnetic fields is followed by a simulation approach for its visual representation.

2.1a *Mechanics behind motion in transverse electric field*

Consider a beam of electrons moving with uniform velocity $\mathbf{v} = v_x \hat{\mathbf{x}}$ that enters a transverse electric field $\mathbf{E} = E_y \hat{\mathbf{y}}$. The moving beam of electrons experiences a force $\mathbf{F} = (-e)(-E_y)\,\hat{\mathbf{y}} = eE_y \hat{\mathbf{y}}$. This force causes an acceleration in the y-direction governed by Newton's 2nd law of motion. Thus, the equation of motion for the electron is:

Please note: $\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}$ refer to unit vectors along the x, y, z directions respectively.

$$\mathbf{F} = m\mathbf{a} \;\Rightarrow\; eE_y \hat{\mathbf{y}} = ma_y \hat{\mathbf{y}} \;\Rightarrow$$

$$a_y = \frac{\mathrm{d}v_y}{\mathrm{d}t} = \frac{e}{m} E_y \Rightarrow$$

$$\mathrm{d}v_y = \frac{e}{m} E_y \mathrm{d}t.$$

Integrating both sides

$$v_y = \frac{e}{m} E_y t + A. \qquad (1)$$

The constant of integration A can be evaluated from the boundary condition that the electron enters the transverse electric field with velocity $\mathbf{v} = v_x \hat{\mathbf{x}}$ (i.e. $v_y = 0$) at time $t = 0$. Substituting $v_y = 0$ at $t = 0$, we get $A = 0$.

Re-writing equation (1), we get

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \frac{e}{m} E_y t$$

$$y = \frac{eE_y t^2}{2m} + B. \qquad (2)$$

The constant of integration B can be evaluated from the boundary condition that the electron enters the transverse electric field with velocity $\mathbf{v} = v_x \hat{\mathbf{x}}$ (i.e $v_y = 0 \Rightarrow y = 0$) at time $t = 0$. Substituting $y = 0$ at $t = 0$, we get $B = 0$. Therefore

$$y = \frac{eE_y t^2}{2m}. \qquad (3)$$

Instead of a parametric equation for the y coordinate, we are interested in the equation of coordinates as a functional dependence. Let l be the length (along x direction) over which the transverse electric field is present. For an electron with initial velocity $\mathbf{v} = v_x \hat{\mathbf{x}}$, the time t taken to traverse a distance x is given by $t = x/v_x$. Substituting this in (3), we get

$$y = \frac{eE_y x^2}{2mv_x^2} \qquad \text{or} \qquad y = \left(\frac{eE_y}{2mv_x^2}\right) x^2. \tag{4}$$

This equation represents a parabola. When the beam travels the length $x = l$ in the transverse electric field, it traces a parabolic path governed by equation (4). When it leaves the transverse field (for $x > l$), it travels along the tangent to the parabola till it hits the screen. Consider the geometry of the path as shown in *Figure* 1.

$$y = L \tan q, \text{ and } \tan q = \frac{v_y}{v_x} \cdot$$

$$y = L\left(\frac{v_y}{v_x}\right) = L\left(\frac{eE_y t}{mv_x}\right) .$$

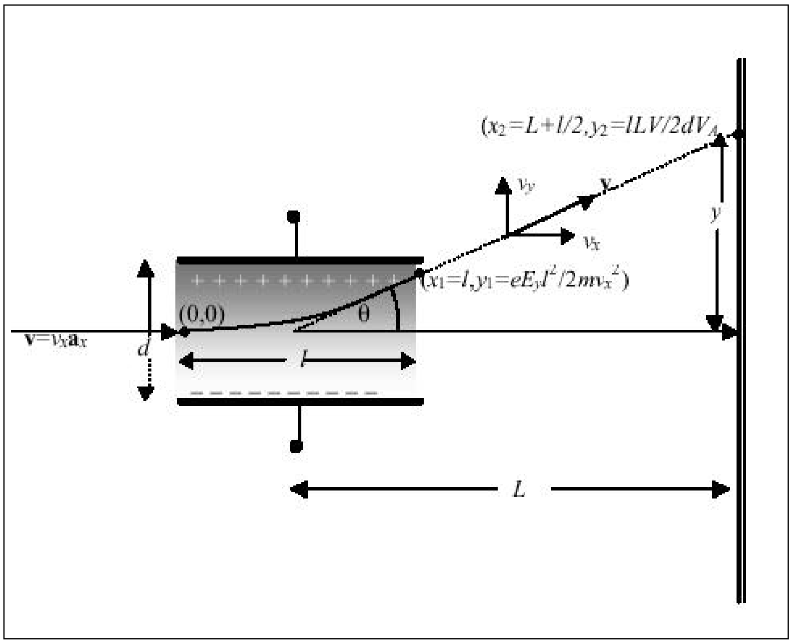

Figure 1. Geometrical sketch of electron path in transverse electric field.

Using $t = \dfrac{l}{v_x}$ (t is the time of flight in the electric field over $x = l$),

$$y = L\left(\frac{eE_y l}{mv_x^2}\right) = \frac{elLE_y}{mv_x^2}.$$

For acceleration through a potential difference of V_a, $\frac{1}{2}mv_x^2 = eV_a$; and $E_y = \frac{V}{d}\cdot$ Therefore

$$y = \frac{lLV}{2V_a d}. \qquad (5)$$

2.1b *Simulation of electron motion in an electric field*

The purpose of the programme is to design a generalized system of simulating an electron in an electric field. The programme inputs the conditions and displays the motion of the electron in a transverse electric field. The input parameters used in the programme from the above discussion are as follows:

$L \rightarrow$ distance between the screen and center of electric field

$l \rightarrow$ length along x-direction over which electric field is present i.e. length of the deflecting plates

$V \rightarrow$ deflection voltage applied across the deflecting plates

$d \rightarrow$ distance between the deflecting plates

$V_a \rightarrow$ acceleration potential imparting v_x velocity to the electron

$e \rightarrow$ charge of electron

$m \rightarrow$ mass of electron.

When an electron enters a transverse electric field it traces a parabolic path. When it leaves the field, it traces a straight line tangent to the parabola at that point.

The flow chart for the simulated electron motion in the transverse electric field is displayed in *Figure* 2.

When the electron (pixel moving along x-direction in simulation programme) enters the transverse electric field $\mathbf{E} = E_y\hat{\mathbf{y}}$ it traces a parabolic path. When it leaves the field, it traces a straight line tangent to the parabola at that point. The geometrical coordinates

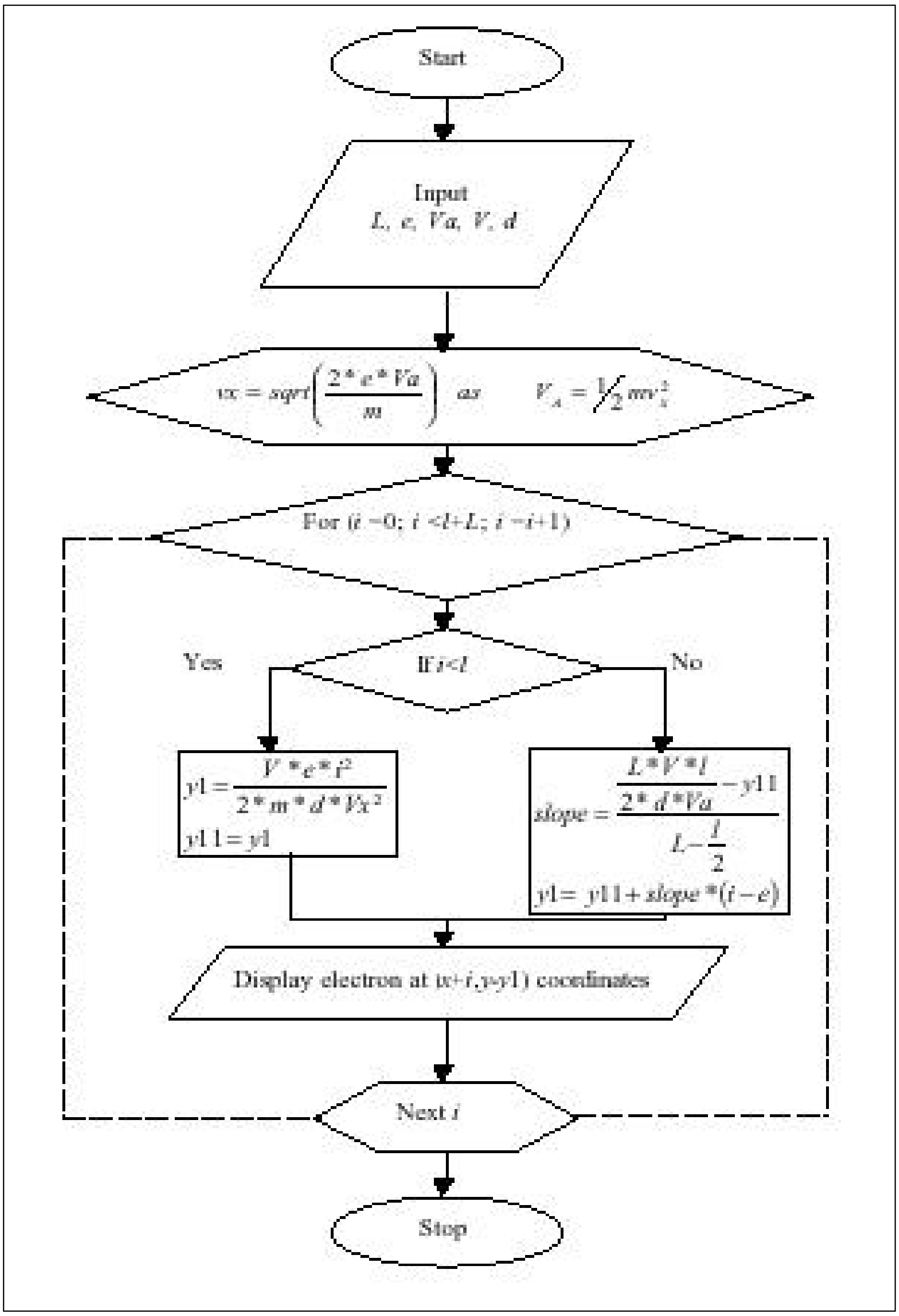

Figure 2. Flow chart for electron motion in transverse electric field.

traced quantitatively in three regions are:

Region I: For $x<0$

In region I, the electron (pixel in simulated programme) traces coordinates $x=v_x t$ and $y=0$. With time, it advances along the x-axis as determined by v_x (proportionate delay in simulated programme), till $x=0$.

Region II: For $0<x<l$

In region II, the electron traces the coordinates:

$$x=v_x t \quad \text{and} \qquad y=\left(\frac{eV}{2mv_x^2 d}\right)x^2 .$$

This parametric equation leads to tracing of a parabola in the region $0<x<l$.

Region III: For $l<x<L+l/2$

In region III, the electron traces a straight line joining the following two points:

$$P_1 \equiv \left(x_1=l,\, y_1=\frac{eVl^2}{2mv_x^2 d}\right) \text{ and } \quad P_2 \equiv \left(x_2=L+\frac{l}{2},\, y_2=\frac{lLV}{2dV_a}\right).$$

$$\text{Slope } m=\frac{y_2-y_1}{x_2-x_1}=\frac{\left(\dfrac{lLV}{2dV_a}-\dfrac{eVl^2}{2mv_x^2 d}\right)}{L-\dfrac{l}{2}}.$$

The flow chart for this simulation (*Figure* 2) reveals the systematic sequence of the steps and the conceptual approach in the simulation of electron motion in the transverse electric field.

2.2a *Mechanics behind motion in transverse magnetic field*

Consider a beam of electrons moving with uniform velocity $\mathbf{v}=v_x\hat{\mathbf{x}}$ that enters a transverse magnetic field $\mathbf{B}=-B_z\hat{\mathbf{z}}$. The moving beam of electrons experiences a Lorentz force $\mathbf{F}=e(\mathbf{v}\times\mathbf{B})=ev_xB_z\hat{\mathbf{y}}$. This force supplies the necessary centripetal

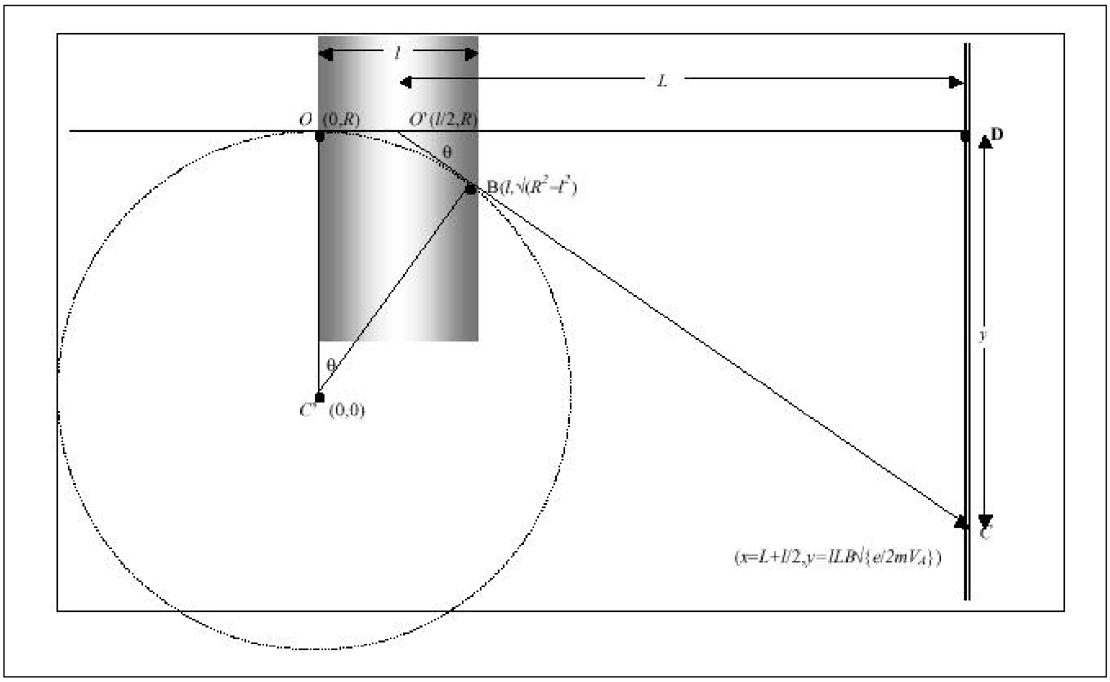

Figure 3. Geometrical sketch of electron path in transverse magnetic field.

force for circular motion.

$$evB_z = \frac{mv^2}{R}, \tag{6}$$

where R represents the radius of the circular orbit. Note that, $\mathbf{v}=v_x\hat{\mathbf{x}}$ is the initial velocity of the electron that enters the magnetic field. It experiences force along the y-direction due to magnetic field and the resultant velocity $\mathbf{v}=v_x\hat{\mathbf{x}} + v_y\hat{\mathbf{y}}$. The work done by a magnetic field is zero and the kinetic energy of a moving charge is invariant in the magnetic field (only the direction of velocity changes). Rearranging equation (6)

$$v = \frac{B_z Re}{m} \quad \Rightarrow$$

$$R = \frac{mv}{eB_z}. \tag{7}$$

The work done by a magnetic field is zero and the kinetic energy of a moving charge is invariant in the magnetic field.

Let the transverse magnetic field be present in a small region of length l along the x-direction. As the uniformly moving electron enters this region, it traces a section of a circular path and leaves the circle along the tangent till it reaches the screen. *Figure* 3 represents the geometrical sketch of the electron path in a transverse magnetic field. We have,

$$q = \frac{arc(OB)}{R} = \frac{l(DC)}{l(O'D)}$$

for **B** to be small, q will be samll and $arc\ (OB) = l$. Therefore

$$\frac{l}{R} = \frac{y}{L} \quad \Rightarrow \quad y = \frac{lL}{R} \cdot \text{ Substituting } R = \frac{mv}{eB},$$

$$y = \frac{lLeB}{mv};$$

(here L is the distance between the screen and center of field).

$$eV_a = \tfrac{1}{2} mv^2 \quad \Rightarrow \quad v = \sqrt{\frac{2eV_a}{m}} \text{ . Therefore}$$

$$y = lLB\sqrt{\frac{e}{2mV_a}} \ . \tag{8}$$

2.2b *Simulation of electron motion in magnetic field*

The purpose of the programme is to design a generalized system for simulating an electron in a magnetic field. The programme inputs the conditions and displays the motion of the electron in a transverse magnetic field. The input parameters used in the programme from the above discussion are as follows:

$L \rightarrow$ distance between the screen and center of electric field
$l \rightarrow$ length along x-direction over which magnetic field is present i.e. length of the deflecting plates
$B \rightarrow$ transverse magnetic flux density
$V_a \rightarrow$ acceleration potential imparting v_x velocity to the electron
$e \rightarrow$ charge of electron
$m \rightarrow$ mass of electron.

When an electron enters a transverse magnetic field it traces a circular path. When it leaves the field, it traces a straight line tangent to the circular arc at that point.

The flow chart for the simulated electron motion in the transverse magnetic field is displayed in *Figure* 4.

When the electron (pixel moving along x-direction in simulation

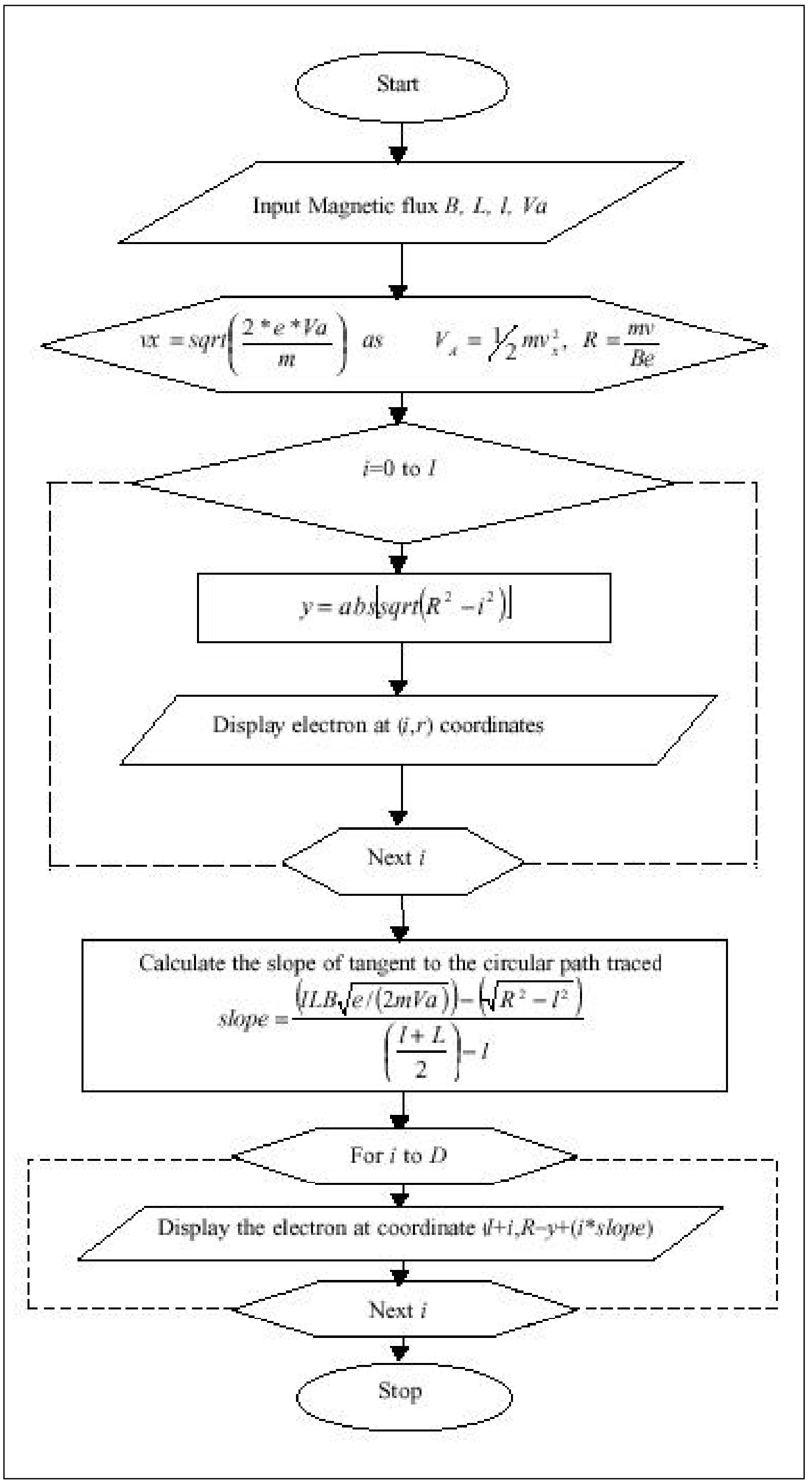

Figure 4. Flow chart for electron motion in transverse magnetic field.

programme) enters a transverse magnetic field B it traces a circular path. When it leaves the field, it traces a straight line tangent to the circular arc at that point. The geometrical coordinates traced quantitatively in three regions are:

Region I: For $x < 0$

In region I, the electron traces the coordinates $x = v_x t$ and $y = 0$. With time, it advances along the x-axis as determined by v_x (proportionate delay in simulated programme), till $x = 0$.

Region II: For $0 < x < l$

In region II, the electron traces coordinates:

$$x = v_x t \quad \text{and} \quad y = \sqrt{R^2 - v_x t}\,.$$

This parametric equation leads to tracing of portion of a circle of radius $R = mv/(eB)$ with center $C'(0,0)$.

Region III: For $l < x < L + l/2$

In region III, the electron traces a straight line joining the following two points:

$$P_1 \equiv \left(x_1 = l,\ y_1 = \sqrt{R^2 - l^2}\right) \text{ and}$$

$$P_2 \equiv \left(x_2 = L + \frac{l}{2},\ y_2 = lLB\sqrt{\frac{e}{2mV_a}}\right).$$

$$\text{Slope } m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{\left(lLB\sqrt{\dfrac{e}{2mV_a}} - \sqrt{R^2 - l^2}\right)}{L - \dfrac{l}{2}}.$$

The flow chart for this simulation (*Figure* 4) reveals the systematic sequence of the steps and conceptual approach in the simulation of electron motion in the transverse magnetic field.

3. Determination of *e/m* using Thomson's method

The combined effect of transverse electric and magnetic fields can be used to calculate *e/m* as:

$$\frac{e}{m} = \frac{VD}{lLdB^2}.$$

Here all terms carry the usual meaning as discussed in earlier sections. This value experimentally calculated by Thomson's method is in excellent agreement with the accepted value of 1.759×10^{11} C/kg.

In simulation module 2.1b, apply deflection voltage (V) and note down displacement (y). In simulation module 2.2b, adjust the value of B such that the same displacement occurs in opposite direction, for fixed value of accelerating potential. The values of d, l, L, V and B will enable calculation of *e/m*.

In an alternate way, comparing equations (5) and (8) under condition of equal and opposite displacement resulting from equal and opposite forces (electric and magnetic), we get,

$$\frac{lLV}{2V_a d} = lLB\sqrt{\frac{e}{2mV_a}}.$$

Simplifying the above equation, we get $\frac{e}{m} = \frac{E^2}{2V_a B^2}$.

By substituting the values of E and B required for producing a null force, for a given value of the accelerating potential V_a, one can calculate *e/m* from the present simulation programme.

Conclusion

Since the input parameters and conditions/situations are defined by students to see the quantitative effects displayed on the screen, this C programme becomes an interesting and interactive teaching aid. The two simulation modules can be used to determine e/m using the principle proposed in Thomson's method.

Appendix 1: Statements of electron motion in electric field

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<math.h>
void  main(void)
{
char *nsc;
int gd=DETECT,gm;
float slop,y11,d=26,D=112.8,l=23.6,V=0.3, x=0,Va,Vx=1*pow (10,5.50),
y=0,i,j,k,y1,e  =  1.6*pow(10,-19),m  =  9.11*pow(10,-31),scale,delay1;
printf("Enter the value of  Distance bet'n Screen & Centre of electric field
D :: ");
scanf("%f",&D);
printf("Enter the value of  Distance bet'n DEFLECTION PLATES l :: ");
scanf("%f",&l);
printf("Enter the value of  ACCELERATING VOLTAGE Va :: ");
scanf("%f",&Va);
printf("Enter the value of  DEFLECTION VOLTAGE V :: ");
scanf("%f",&V);
printf("Enter the value of  DISTANCE BET'N PLATES d :: ");
scanf("%f",&d);
Vx= sqrt(2*e*Va/m);
delay1  =  50*Vx/(3*pow(10,8));
initgraph(&gd,&gm,"c:\\tc3\\bgi");
rectangle(1,1,630,470);
rectangle(2,2,629,469);
rectangle(4,4,627,467);
sprintf(nsc,"d=  %f",d);
outtextxy(10,10,nsc);
sprintf(nsc,"D=  %f",D);
outtextxy(10,20,nsc);
sprintf(nsc,"l=  %f",l);
outtextxy(10,30,nsc);
sprintf(nsc,"V=  %f",V);
outtextxy(10,40,nsc);
sprintf(nsc,"Vx= %f",Vx);
outtextxy(10,50,nsc);
outtextxy(300,60,"PARABOLA");
```

```
scale = 400/(D+l/2);
d=d*scale;l=l*scale;D=D*scale;
rectangle(100,250-d,100+l,245-d);
rectangle(100,250+d,100+l,245+d);
rectangle(100+l+D,25,100+l+D+5,450);
for(i=0;i<=21+l+D;i=i+1)
{
        if(i>20 && i<(20+l))
        {
        y1=(V*e*(i-20)*(i-20))/(2*m*d*Vx*Vx);
        y11=y1;
        putpixel(80+x+i,250+y-y1,12);
        }
        if(i>=(20+l))
        {
        slop=((l*D*V)/(2*d*Va)-y11)/(D-l/2);
        y1=y11+slop*(i-l-20)+1;
        putpixel(80+x+i,250+y-y1,14);
        }
        delay(50-(int)delay1);
}
outtextxy(410,450,"Press Any key to continue....");
getch();
}
```

Appendix 2. Statements of electron motion in magnetic field

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<math.h>
void electron(float x,float y)
{   int i;
for( i=0;i<1;i++)
circle(x,y,i);
}
void main(void)
{
int gd=DETECT,gm;
float i,y,l=30,L=50,D=200,R,THETA,d=30,slop,Va,delay1,x1,y1,x2,y2;
float m=9.11*pow(10,-31),e=1.6*pow(10,-19),B=1.5*pow(10,-4),v =
1*pow(10,10);
```

```
printf("Enter the MAGNETIC FLUX B (in GAUSS)::");
scanf("%f",&B);
B=B*pow(10,-11);
printf("Enter the value of L ::");
scanf("%f",&L);
printf("Enter the value of l ::");
scanf("%f",&l);
printf("Enter the value of ACCELERATING VOLTAGE    Va::");
scanf("%f",&L);
printf("R= %f\n v = %f\n",R,v);
getch();
v= abs(sqrt((2*e*Va)/m));
printf("R= %f\n v = %f\n",R,v);
getch();
R = m*v/(B*e);
printf("R= %f\n v = %f\n",R,v);
getch();
delay1 = 50*v/(3*pow(10,8));
initgraph(&gd,&gm,"c:\\tc3\\bgi");
rectangle(100,250-d,100+l,245-d);
rectangle(100,250+d,100+l,245+d);
rectangle(100+l+D,25,100+l+D+5,450);
for(i=0;i<=l;i++)
{
y = abs(sqrt(abs((R*R)-(i*i))));
putpixel(100+i,250+R-y,14);
delay(50-(int)delay1);
}
x1= l;x2=L+l/2;y1=abs(sqrt(abs(R*R-l*l)));y2=l*L*B*abs(sqrt(e/
(2*m*Va)));
slop = (y2-y1)/(x2-x1);
for(i=0;i<D;i++)
{
putpixel(100+l+i,250+R-y+(i*slop),15);
  delay(50-(int)delay1);
 }
outtextxy(400,440,"Press any key to continue ...");
getch();
}
```