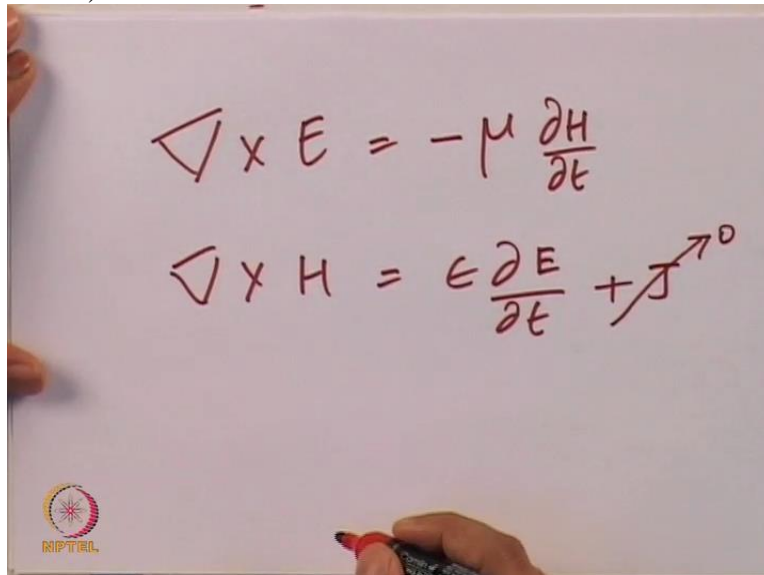


Computational Electromagnetics and Applications
Professor Krish Sankaran
Indian Institute of Technology Bombay
Exercise No. 8
Finite Difference Method -III

In this module we are going to look into some of the modeling examples for Maxwell equations. And we are going to use Finite difference method to do this. And while doing that we are going to see some of the important aspects like modeling and slit experiment or modeling propagation with certain boundary conditions like perfectly match layer of perfect electric conductor or perfect magnetic conductor so on and so forth.

So there is lot going on in this exercise. We will take it step by step and we will look at various aspects of it as we go. One of the important aspects of this module is the boundary truncation. We have not explicitly handled perfectly matched layer in this particular example. We will do it separately in a different example. For us now there is a perfectly matched layer that we are using but we will cover it much more elaborately in a later stage. We will just give a very brief idea what it is in this example. With that introduction let us start that example that we are interested in modeling today. And we will take you step by step into the problem.

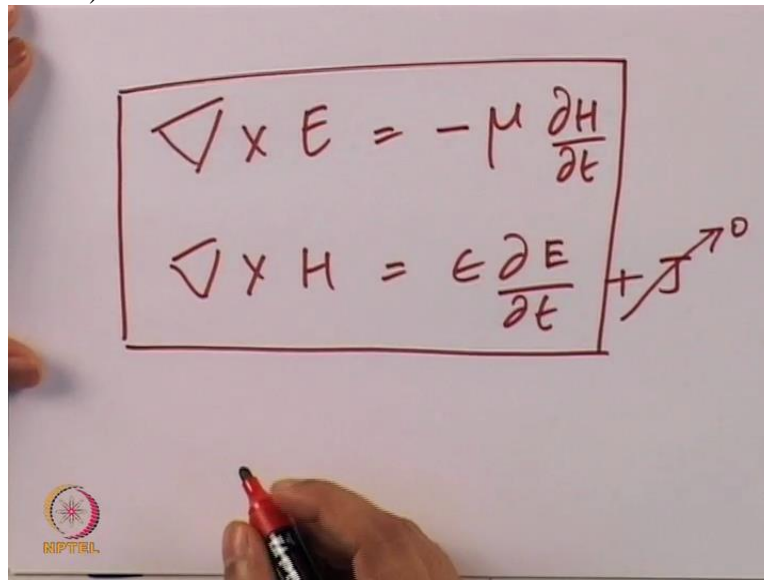
(Refer Slide Time: 01:24)



The image shows a whiteboard with two Maxwell equations written in red marker. The first equation is $\nabla \times E = -\mu \frac{\partial H}{\partial t}$. The second equation is $\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$, where the current density term J has a small arrow pointing to the right above it. In the bottom left corner of the whiteboard, there is a logo for RIPTIL (Research Institute for Plasma Theory and Applications) featuring a stylized sun or starburst design.

And for that first let us begin with the mathematical governing equation. So the initial governing equation that we are interested is the Maxwell equation itself, which is nothing but the curl of E is equal to minus Mu dH by dt. And then the curl of H is equal to Epsilon dE by dt plus J, of course we are not considering the J term the current density term. For us now we keep it as 0

(Refer Slide Time: 02:17)

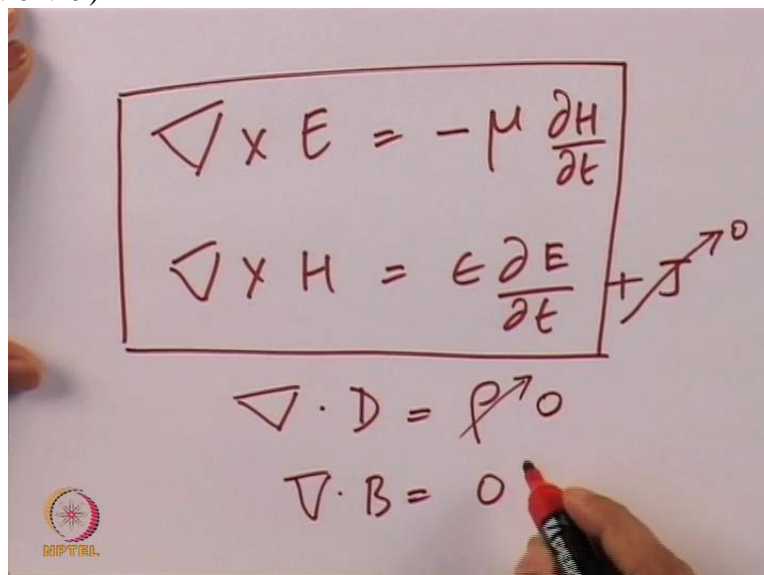


A hand-drawn box on a whiteboard containing two equations. The first equation is $\nabla \times E = -\mu \frac{\partial H}{\partial t}$. The second equation is $\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$, with a vector arrow labeled J pointing upwards and to the right. A hand holding a red marker is visible at the bottom of the frame.

$$\nabla \times E = -\mu \frac{\partial H}{\partial t}$$
$$\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$$

So the two important update equations that we will be focusing on in today's exercise will be coming directly from the curl equation.

(Refer Slide Time: 02:29)



A hand-drawn box on a whiteboard containing three equations. The first equation is $\nabla \times E = -\mu \frac{\partial H}{\partial t}$. The second equation is $\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$, with a vector arrow labeled J pointing upwards and to the right. Below the box, the third equation is $\nabla \cdot D = \rho$, with a vector arrow labeled ρ pointing upwards and to the right. Below that is the fourth equation, $\nabla \cdot B = 0$. A hand holding a red marker is visible at the bottom of the frame.

$$\nabla \times E = -\mu \frac{\partial H}{\partial t}$$
$$\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$$
$$\nabla \cdot D = \rho$$
$$\nabla \cdot B = 0$$

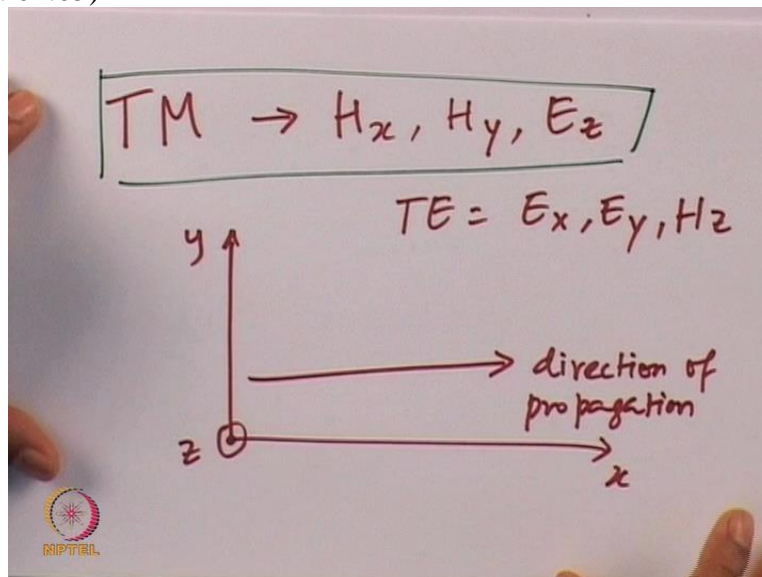
Apart from that there are other two equations which is the divergences of D is equal to ρ and divergences of B is equal to 0. For us in this problem we consider ρ to be also equal to 0, so both the divergence conditions are set to 0. In the finite difference method particularly the modeling that we are going to use which is staggered modeling.

The divergences are not explicitly handled. We are considering the divergence conditions are implicitly satisfied. So specifically forcing them in any of the update equation. With that being

said it is a important source of error in most of the numerical methods which leads to spurious modes and so on and so forth. And we are not worrying about such errors in this case because we are implicitly assuming that the diversions conditions are satisfied and it is also easy to prove in some methods the diversions conditions will be satisfied.

But here that is not the focus we will just focus on the two equations which are the curl equations which I have written here. So now for us to go into problem modeling we will focus on a 2D case. So in this example we are going to focus on a transverse magnetic mode or transverse magnetic wave. So why we are using transverse magnetic because it is a easier choice you can also choose a transverse electric in a 2D model.

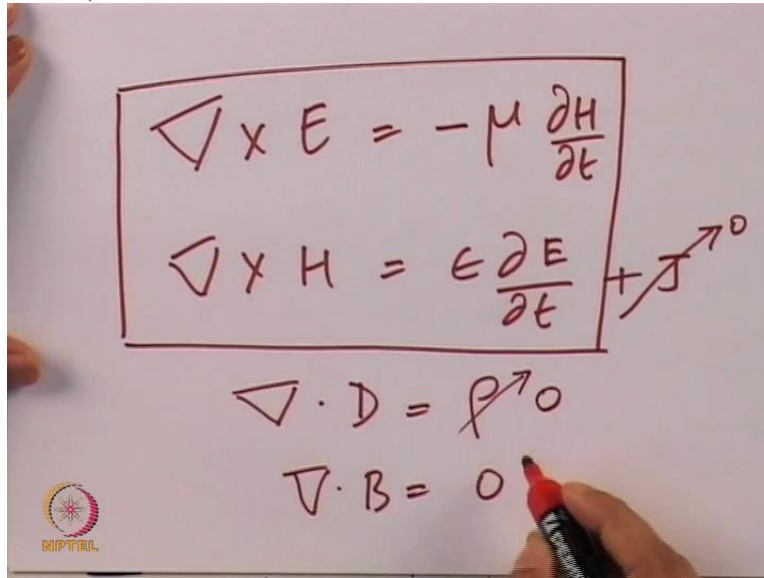
(Refer Slide Time: 04:05)



So we will start with a transverse magnetic aspect which is going to be, TM implies H_x, H_y, E_z , if we say that the domain that we are interested is going to be x, y and z is coming out of this plane. And if we assume that x is the direction of propagation. And we say the H field magnetic field is going to have components in the x and y direction. And the electric field is going to have components only in the z direction.

Different textbooks call this as different names. Different textbooks use different names. So be careful how you are defining this. For us the definition is very clear and simple, that once we define the direction of propagation and the plane of propagation which will be xy plane. Our magnetic field is purely in the plane of propagation, and the electric field is going to be perpendicular to the plane of propagation we use the definition of transverse magnetic.

(Refer Slide Time: 05:29)

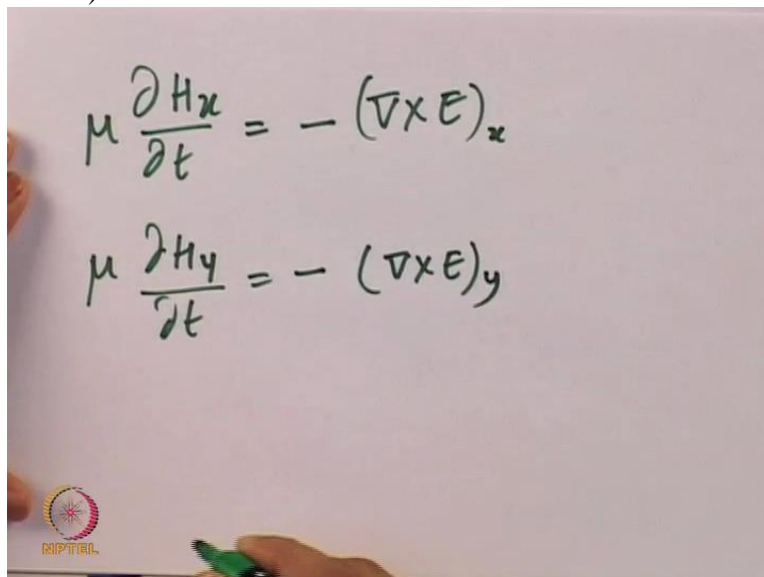


A whiteboard with handwritten Maxwell equations in red ink. The equations are enclosed in a rectangular box. To the right of the box, there is a vector arrow labeled 'J' with a plus sign. Below the box, there are two more equations. A hand holding a red marker is visible at the bottom right.

$$\nabla \times E = -\mu \frac{\partial H}{\partial t}$$
$$\nabla \times H = \epsilon \frac{\partial E}{\partial t} + J$$
$$\nabla \cdot D = \rho$$
$$\nabla \cdot B = 0$$

The counter part of that will be transverse electric where you will have E field in x and y direction and H field in z direction. This is not what you are going to do. In our case it is going to be transverse magnetic so let us put in the box. So with this as our models bases we are going to model the Maxwell equations, the Curl Equations that we have set here in the form of that we want.

(Refer Slide Time: 06:06)



A whiteboard with handwritten component equations for the curl of H in black ink. A hand holding a green marker is visible at the bottom.

$$\mu \frac{\partial H_x}{\partial t} = -(\nabla \times E)_x$$
$$\mu \frac{\partial H_y}{\partial t} = -(\nabla \times E)_y$$

So H field is going to have x and y components. So let us write down the Maxwell equation $\mu \frac{\partial H_x}{\partial t}$ is equal to minus $(\nabla \times E)_x$, this is going to be the x component. And $\mu \frac{\partial H_y}{\partial t}$ is equal to minus $(\nabla \times E)_y$ component.

(Refer Slide Time: 06:51)

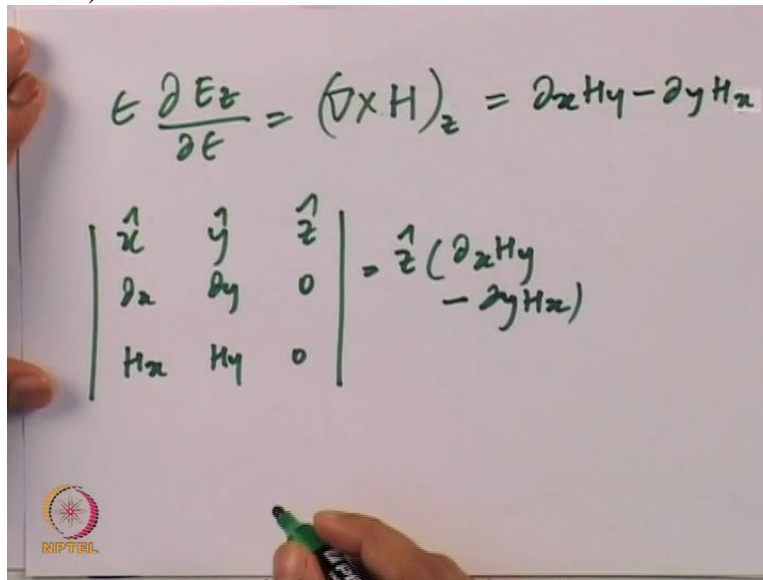
$$\mu \frac{\partial H_x}{\partial t} = -(\nabla \times E)_x = -\partial_y E_z$$
$$\mu \frac{\partial H_y}{\partial t} = -(\nabla \times E)_y = \partial_x E_z$$
$$\begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ \partial_x & \partial_y & \partial_z \\ 0 & 0 & E_z \end{vmatrix} = \begin{vmatrix} \hat{x} (\partial_y E_z) \\ +\hat{y} (-\partial_x E_z) \\ 0 \end{vmatrix}$$

So if we write it down in individual forms what we will get is $\partial_y E_z$, $\partial_x E_z$, and we have got $0 \ 0 \ E_z$. I said the E field is going to have only component in z direction. So we put the other components as 0. Since we have no variation in z direction. We are going to assign this also to 0.

Let us write down all the components. All components in the sense two components are there x component is going to be $\partial_y E_z$ and the y component is going to be plus y it is going to be (minus $\partial_x E_z$) and then the z component will not be there.

So let us write down now the minus of the x component will be minus $\partial_y E_z$ and minus of the y component is going to be minus of minus it will become plus $\partial_x E_z$.

(Refer Slide Time: 08:21)

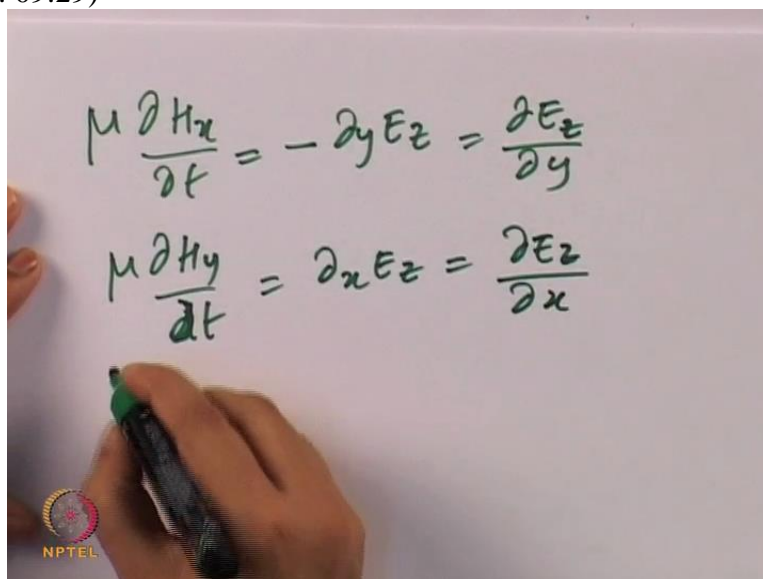


The image shows a whiteboard with handwritten mathematical expressions. At the top, the equation $\epsilon \frac{\partial E_z}{\partial t} = (\nabla \times \mathbf{H})_z = \partial_x H_y - \partial_y H_x$ is written. Below it, a determinant is shown: $\begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ \partial_x & \partial_y & 0 \\ H_x & H_y & 0 \end{vmatrix} = \hat{z} (\partial_x H_y - \partial_y H_x)$. The NPTEL logo is visible in the bottom left corner.

Similarly we will also have the components for the electric field. So let us write it down, Epsilon $\frac{\partial E_z}{\partial t}$ is equal to Curl of H going to have only the z component. So the z component is going to be given by this expression $\hat{x} \hat{y} \hat{z} \partial_x \partial_y 0$ and then we get $H_x H_y 0$. So this will have only the z component because the x components will have the 0 here and the y component will also have the 0.

So we only get the z component it is given by $(\partial_x H_y) - (\partial_y H_x)$. So let us write down this one so the z component is going to be given by $\partial_x H_y - \partial_y H_x$. So this is going to be the third equation.

(Refer Slide Time: 09:29)



The image shows a whiteboard with two handwritten equations. The first equation is $\mu \frac{\partial H_x}{\partial t} = -\partial_y E_z = \frac{\partial E_z}{\partial y}$. The second equation is $\mu \frac{\partial H_y}{\partial t} = \partial_x E_z = \frac{\partial E_z}{\partial x}$. The NPTEL logo is visible in the bottom left corner.

So the first two equations are namely $\mu \frac{\partial H_x}{\partial t} = -\partial_y E_z$ equal to minus $\partial_y E_z$. So it is nothing but $\partial_y E_z$ by ∂t . I am writing it in short form $\mu H_y \partial t$ is equal to $\partial_x E_z$ it is nothing but $\partial_x E_z$ divided by ∂t .

(Refer Slide Time: 10:06)

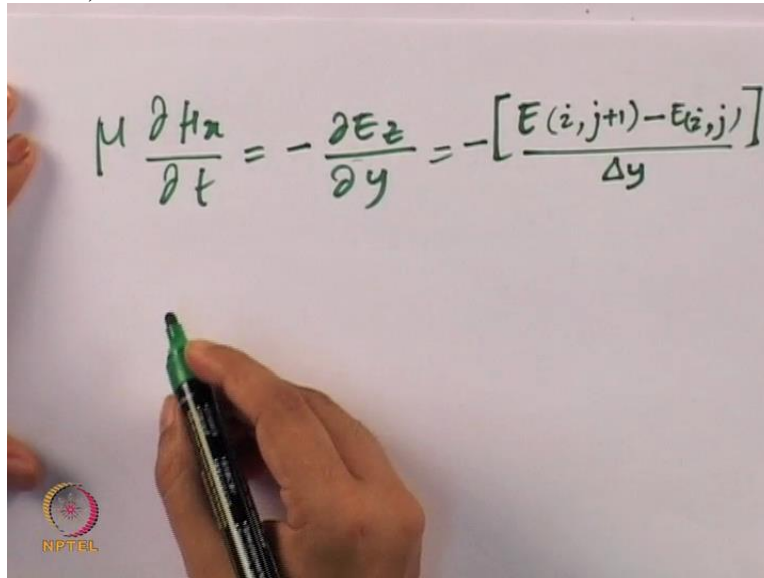
$$\mu \frac{\partial H_x}{\partial t} = -\partial_y E_z = \frac{\partial E_z}{\partial y}$$

$$\mu \frac{\partial H_y}{\partial t} = \partial_x E_z = \frac{\partial E_z}{\partial x}$$

$$\epsilon \frac{\partial E_z}{\partial t} = \partial_x H_y - \partial_y H_x = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}$$

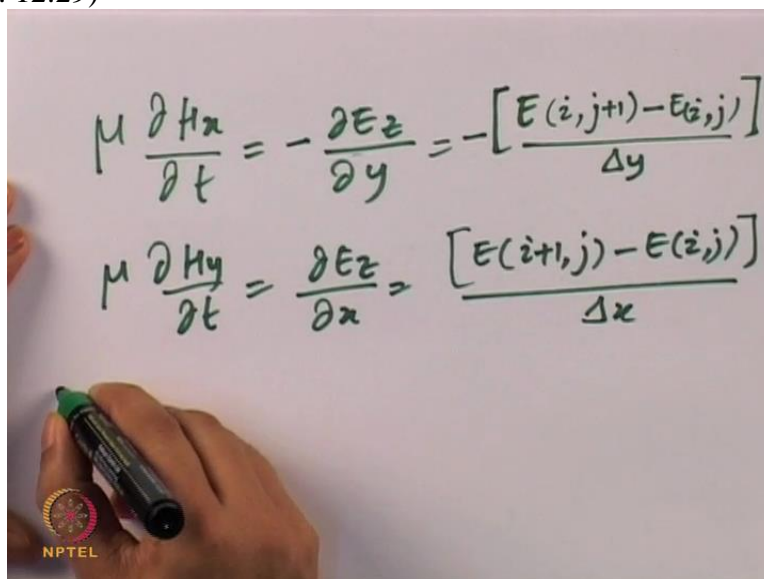
And the last one is $\epsilon \frac{\partial E_z}{\partial t}$ is equal to $\partial_x H_y$ minus $\partial_y H_x$. So this is going to be written in a conventional form as $\partial_x H_y$ divided by ∂t minus $\partial_y H_x$ divided by ∂t . So this is still a continuous form. We need to discretize it using certain algorithms so what is important know here is we are going to combine forward differencing and backward differencing for the fields spatial derivatives and we are going to do Leap frogging in time derivative. So I am going to explain this step by step.

(Refer Slide Time: 11:15)


$$\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\left[\frac{E(z, j+1) - E(z, j)}{\Delta y}\right]$$

So let us take first the spatial derivative which is easy to model. So let us take the first equation what we have. $\mu \frac{\partial H_x}{\partial t}$ is equal to $-\frac{\partial E_z}{\partial y}$ so we will get minus E_z divided by Δy . So we are going to do forward differencing for E_z . So we are going to take since it is a two dimensional problem what we will get is minus sign is out $[E(i, j+1) - E(i, j)]$ divided by Δy . So what we have done is we have taken the forward differencing in the y direction. How do I know which direction I have to forward difference it is given by the partial derivative itself so if it is $\frac{\partial}{\partial y}$ you do forward differencing or backward differencing in the particular variable? If it is $\frac{\partial}{\partial y}$ you do in j if it is $\frac{\partial}{\partial x}$ you do it in i .

(Refer Slide Time: 12:29)


$$\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\left[\frac{E(z, j+1) - E(z, j)}{\Delta y}\right]$$
$$\mu \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x} = \left[\frac{E(i+1, j) - E(i, j)}{\Delta x}\right]$$

So let us say we have the second equation $\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\frac{[E(i, j+1) - E(i, j)]}{\Delta y}$ divided by Δy . So in this case we will do forward differencing in x component which is $[E(i+1, j) - E(i, j)]$ divided by Δx .

(Refer Slide Time: 13:03)

The image shows three equations written in black marker on a whiteboard. The first equation is $\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\frac{[E(i, j+1) - E(i, j)]}{\Delta y}$. The second equation is $\mu \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x} = \frac{[E(i+1, j) - E(i, j)]}{\Delta x}$. The third equation is $\epsilon \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}$. An NPTEL logo is visible in the bottom left corner.

And in the case of magnetic field component you are going to do backward differencing and let us see how we are doing it. So we have the third equation $\epsilon \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}$, which is nothing but $\frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y}$.

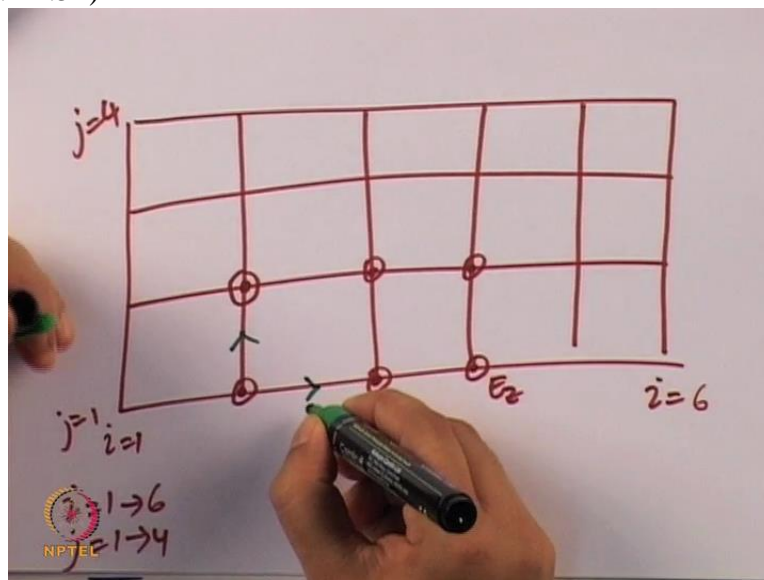
(Refer Slide Time: 13:41)

The image shows three equations written in black marker on a whiteboard. The first equation is $\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\frac{[E(i, j+1) - E(i, j)]}{\Delta y}$. The second equation is $\mu \frac{\partial H_y}{\partial t} = \frac{\partial E_z}{\partial x} = \frac{[E(i+1, j) - E(i, j)]}{\Delta x}$. The third equation is $\epsilon \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = \left[\frac{-(H(i, j) - H(i, j-1))}{\Delta y} + \frac{(H(i, j) - H(i-1, j))}{\Delta x} \right]$. An NPTEL logo is visible in the bottom left corner.

So I am going to write down this equation I have said I will do backward differencing in H. So $[H(i, j) - H(i, j - 1)] / \Delta y$. So I am doing backward differencing in the J component. So this is this term, and of course I have a minus and I put a minus in the front. And since I started as the second term, I will do the first term now it should be plus backward differencing in x component which is i. So I will have Plus $(H(i, j) - H(i - 1, j)) / \Delta x$.

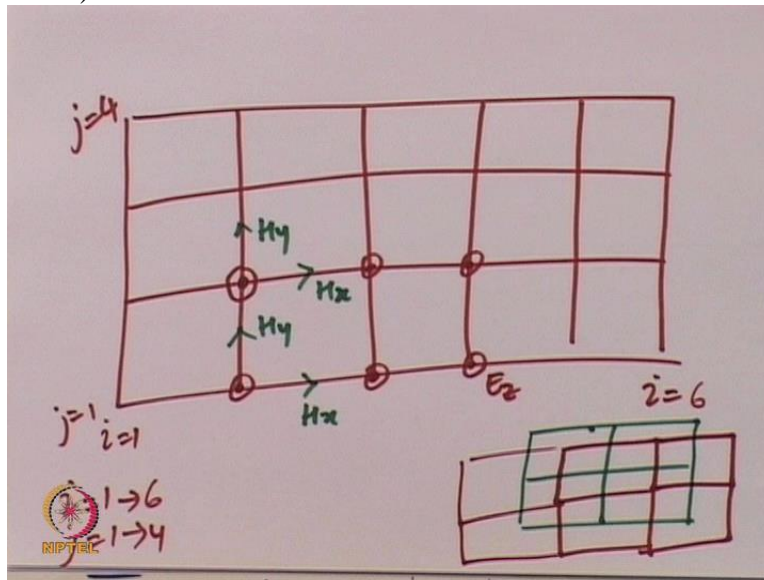
So as you can see the electric field component spatial derivatives are forward difference. The magnetic field components spatial derivatives are backward difference. So let me explain that.

(Refer Slide Time: 14:51)



So this is your domain this is i equal to 1, this is i equal to 2, 3, 4, 5, 6. And this is j equal to 1, 2, 3, 4. So domain is going to have i running from 1 to 6, j running from 1 to 4. And now the magnetic fields components are going to be positioned in such a way that the electric field components will take the nodal values. So we have all the nodes will have electric field.

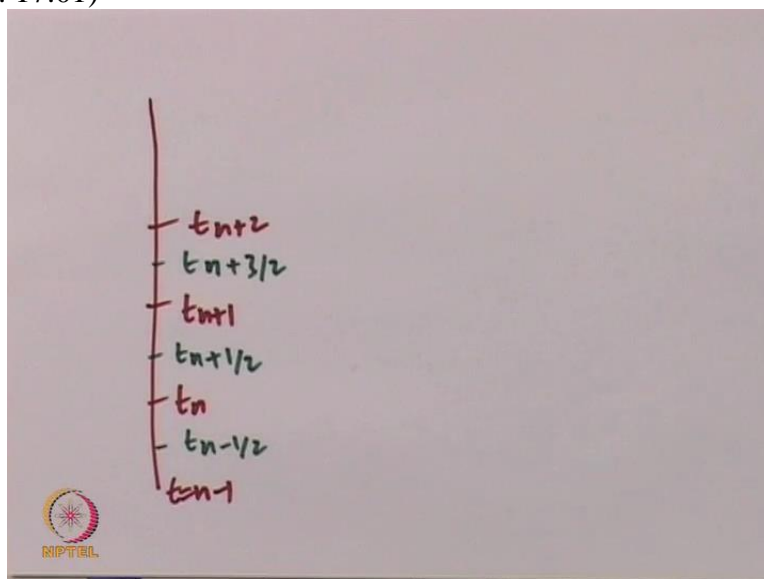
(Refer Slide Time:15:50)



Whereas the sides are going to be components of the magnetic field, so the magnetic fields are going to be H_x , H_y . Similarly H_x , H_y so on and so forth. So the staggering will look in 2D simply like this. So the red color will be a staggered grid to the green color. So the reason for doing that we discussed it already in the finite difference method. But we will give more elaborate discussion about it when we treat algebraic topological method. We will now focus on how the time stepping is going to be done.

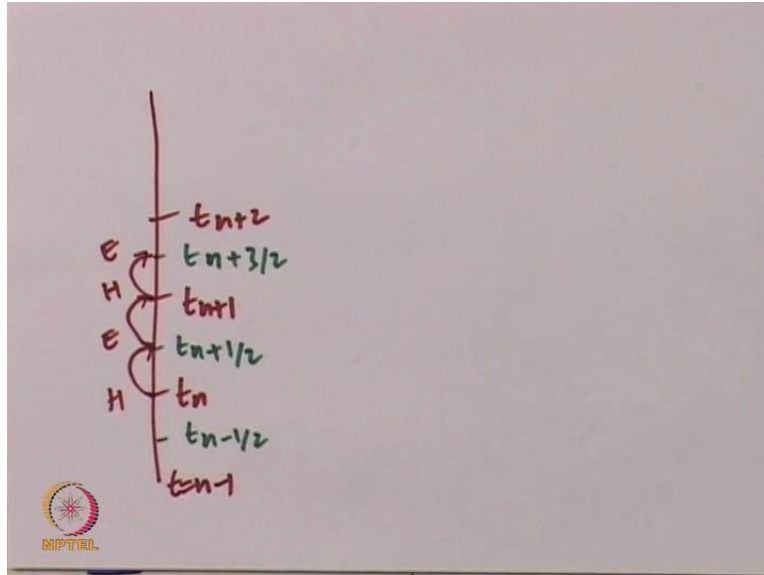
It is going to go through a loop. So loop is going to take care of the staggering requirement what we need.

(Refer Slide Time: 17:01)



So I said this is the timeline, so this is t equal to n minus 1, t equal to n , t equal to n plus 1, t equal to n plus 2, so on and so forth. There is always going to be a intermediate point ; this is going to be $t_{n-1/2}$, $t_{n+1/2}$, $t_{n+3/2}$ so on and so forth.

(Refer Slide Time: 17:26)

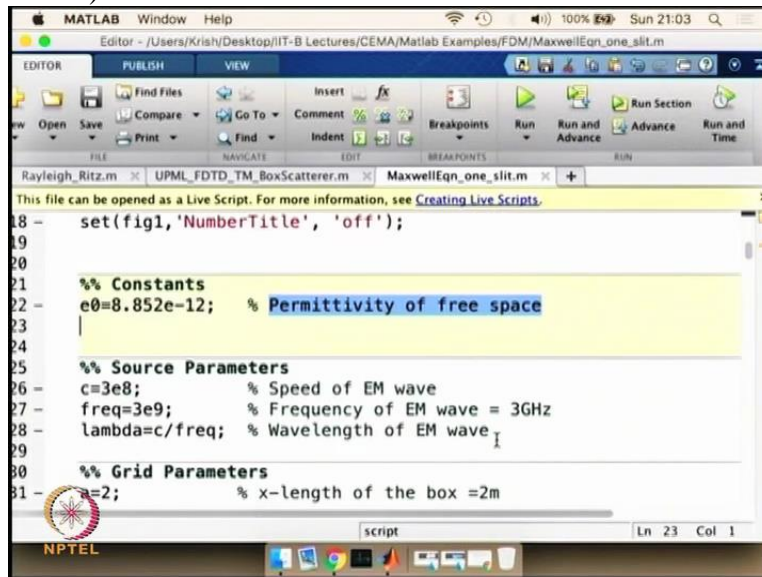


So we are going to first start with the electric field spatial derivative that we are computing to update the magnetic field. So at time t equal to n start with the magnetic field and so we start with H we go and compute E here and we go and compute H field again, we use that to compute the E field again so on and so forth. So we are going step by step, so that is why we call it as Leap frogging method.

(Refer Slide Time: 18:11)

And this is going to be a single slit problem. So we are doing a single slit diffraction experiment. And we are starting with the Finite difference method. And I have already discussed how we are doing the differencing.

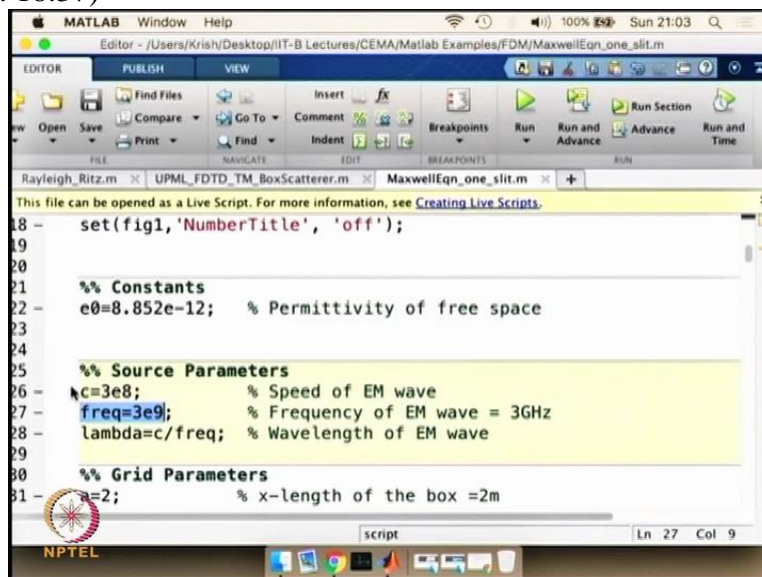
(Refer Slide Time: 18:25)



```
18 - set(fig1,'NumberTitle','off');
19
20
21 %% Constants
22 - e0=8.852e-12; % Permittivity of free space
23
24
25 %% Source Parameters
26 - c=3e8; % Speed of EM wave
27 - freq=3e9; % Frequency of EM wave = 3GHz
28 - lambda=c/freq; % Wavelength of EM wave
29
30 %% Grid Parameters
31 - a=2; % x-length of the box =2m
```

So let us start by defining certain parameters constants Epsilon which is given by 8.852 multiplied by 10 to the power of minus 12, which is the permittivity of free space.

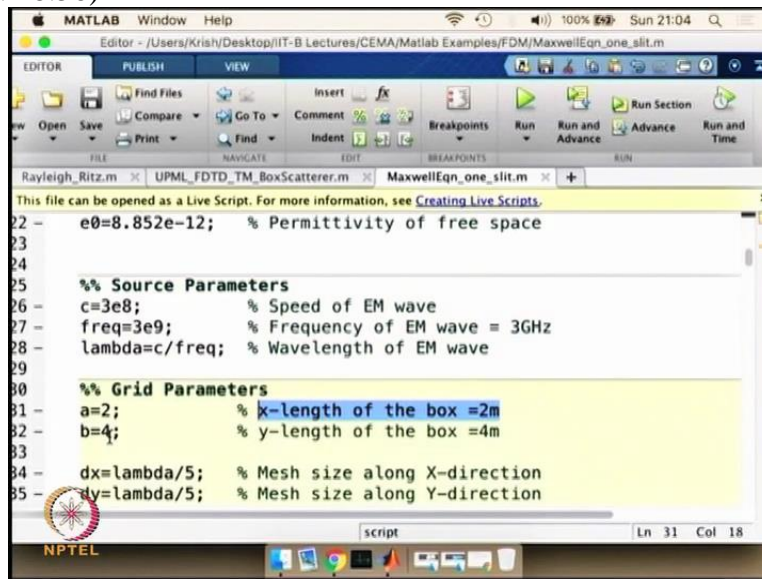
(Refer Slide Time: 18:37)



```
18 - set(fig1,'NumberTitle','off');
19
20
21 %% Constants
22 - e0=8.852e-12; % Permittivity of free space
23
24
25 %% Source Parameters
26 - c=3e8; % Speed of EM wave
27 - freq=3e9; % Frequency of EM wave = 3GHz
28 - lambda=c/freq; % Wavelength of EM wave
29
30 %% Grid Parameters
31 - a=2; % x-length of the box =2m
```

We are setting the value of the frequency that we are exciting as 3 Giga hertz, which corresponds to roughly 0.1 meter or 10 centimeter.

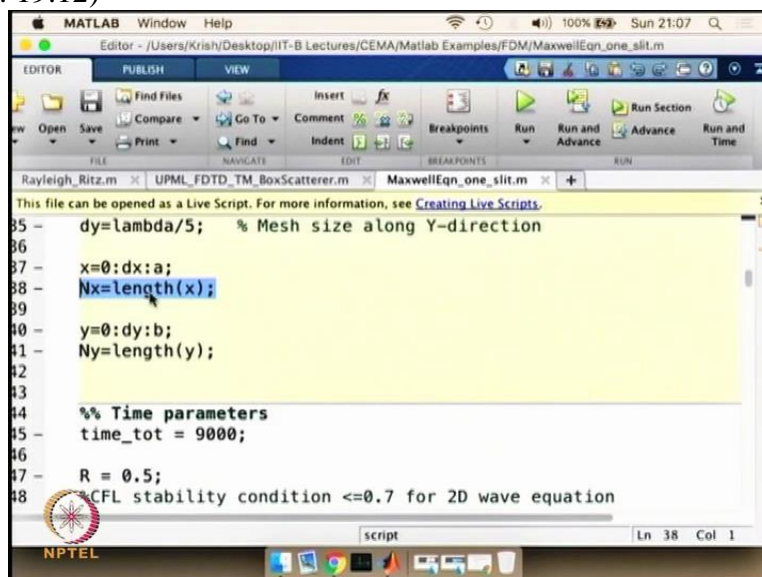
(Refer Slide Time: 18:56)



```
MATLAB Window Help
Editor - /Users/Krish/Desktop/IIIT-B Lectures/CEMA/Matlab Examples/FDM/MaxwellEqn_one_slit.m
EDITOR PUBLISH VIEW
Find Files Insert
Open Save Compare Go To Comment Breakpoints Run Run and Advance Run Section Run and Time
Print Find Indent Breakpoints Run Run and Advance Advance Run and Time
FILE NAVIGATE EDIT BREAKPOINTS RUN
Rayleigh_Ritz.m UPML_FDTD_TM_BoxScatterer.m MaxwellEqn_one_slit.m
This file can be opened as a Live Script. For more information, see Creating Live Scripts.
22 - e0=8.852e-12; % Permittivity of free space
23
24
25 %% Source Parameters
26 - c=3e8; % Speed of EM wave
27 - freq=3e9; % Frequency of EM wave = 3GHz
28 - lambda=c/freq; % Wavelength of EM wave
29
30 %% Grid Parameters
31 - a=2; % x-length of the box =2m
32 - b=4; % y-length of the box =4m
33
34 - dx=lambda/5; % Mesh size along X-direction
35 - dy=lambda/5; % Mesh size along Y-direction
script Ln 31 Col 18
NPTEL
```

And we are setting the boundaries of our parameter a equal to 2, and b equal to 4. a is the x length of the box which is 2 meter and y length of the box is b which is 4 meters.

(Refer Slide Time: 19:12)



```
MATLAB Window Help
Editor - /Users/Krish/Desktop/IIIT-B Lectures/CEMA/Matlab Examples/FDM/MaxwellEqn_one_slit.m
EDITOR PUBLISH VIEW
Find Files Insert
Open Save Compare Go To Comment Breakpoints Run Run and Advance Run Section Run and Time
Print Find Indent Breakpoints Run Run and Advance Advance Run and Time
FILE NAVIGATE EDIT BREAKPOINTS RUN
Rayleigh_Ritz.m UPML_FDTD_TM_BoxScatterer.m MaxwellEqn_one_slit.m
This file can be opened as a Live Script. For more information, see Creating Live Scripts.
35 - dy=lambda/5; % Mesh size along Y-direction
36
37 - x=0:dx:a;
38 - Nx=length(x);
39
40 - y=0:dy:b;
41 - Ny=length(y);
42
43
44 %% Time parameters
45 - time_tot = 9000;
46
47 - R = 0.5;
48 - CFL stability condition <=0.7 for 2D wave equation
script Ln 38 Col 1
NPTEL
```

And nx is going to be the length of x space stripping. And y is going to be the length of y space stripping.

(Refer Slide Time: 19:23)

```
39
40 - y=0:dy:b;
41 - Ny=length(y);
42
43
44 % Time parameters
45 - time_tot = 9000;
46
47 - R = 0.5;
48 %CFL stability condition <=0.7 for 2D wave equation
49
50 - dt=R*dx/c;
51 - tsteps=time_tot;
52
```

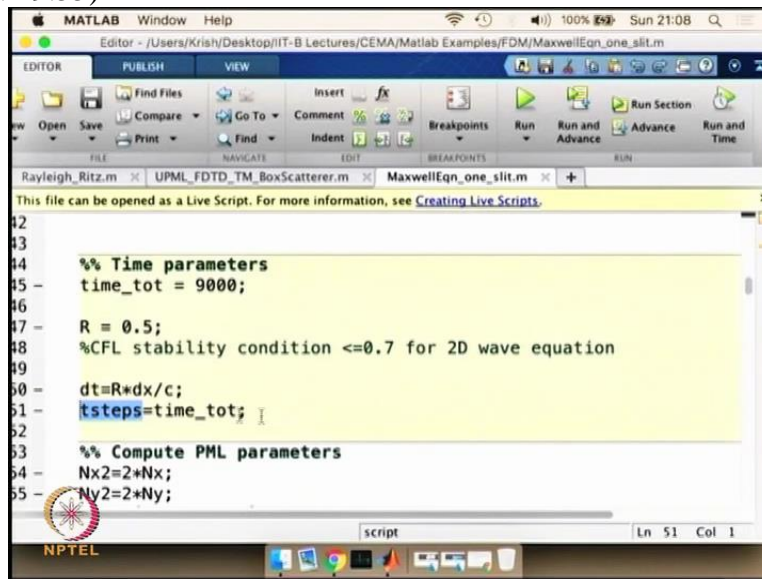
The time duration for which we are going to run this simulation is going to be 9000 time steps. We have kept r value is equal to 5 on condition that we are satisfying the CFL condition.

(Refer Slide Time: 19:32)

```
39
40 - y=0:dy:b;
41 - Ny=length(y);
42
43
44 % Time parameters
45 - time_tot = 9000;
46
47 - R = 0.5;
48 %CFL stability condition <=0.7 for 2D wave equation
49
50 - dt=R*dx/c;
51 - tsteps=time_tot;
52
```

We can go r slightly above 0.5 also, but for this particular experiment we just wanted to show the proof of concept and we are not worried about the time stepping as such. So we are making sure that the stability is satisfied, so we are keeping it in 0.5.

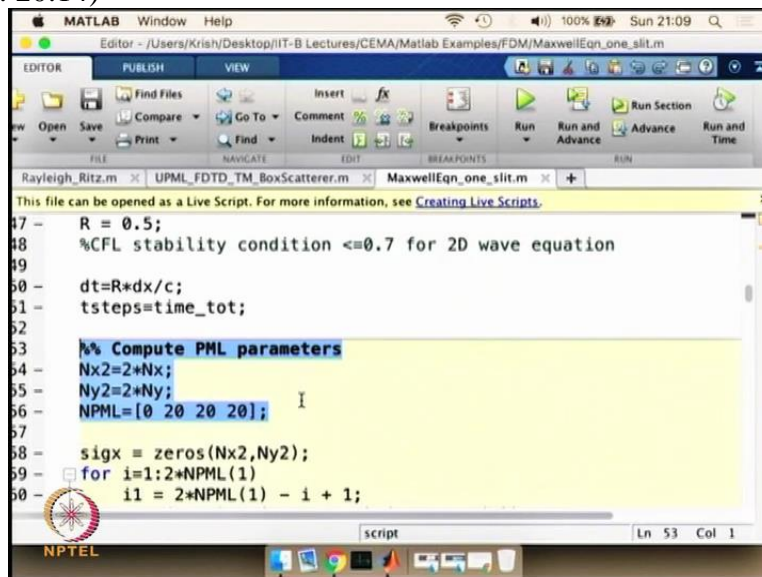
(Refer Slide Time: 19:55)



```
42
43
44 %% Time parameters
45 - time_tot = 9000;
46
47 - R = 0.5;
48 - %CFL stability condition <=0.7 for 2D wave equation
49
50 - dt=R*dx/c;
51 - tsteps=time_tot;
52
53 %% Compute PML parameters
54 - Nx2=2+Nx;
55 - Ny2=2+Ny;
```

Once these parameters are fixed you can compute the value of dt using r, dx and c, and we will get the value for the time stepping. And the total time duration is already given so we call it t steps.

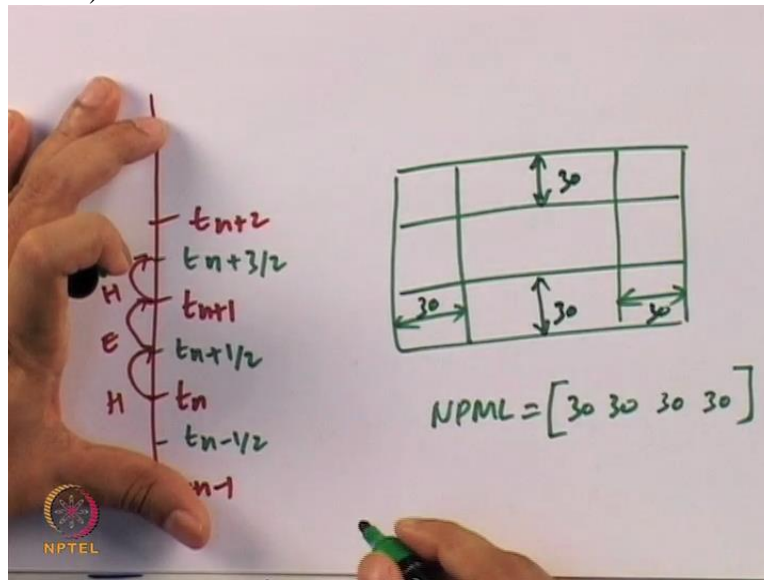
(Refer Slide Time: 20:14)



```
47 - R = 0.5;
48 - %CFL stability condition <=0.7 for 2D wave equation
49
50 - dt=R*dx/c;
51 - tsteps=time_tot;
52
53 %% Compute PML parameters
54 - Nx2=2+Nx;
55 - Ny2=2+Ny;
56 - NPML=[0 20 20 20];
57
58 - sigx = zeros(Nx2,Ny2);
59 - for i=1:2+NPML(1)
60 -     i1 = 2+NPML(1) - i + 1;
```

And now we are going to put the PML conditions. We are not going to discuss specifically the PML conditions in this particular module as I said before. This is going to be a uniaxial PML. I will describe the concept of PML at a later module. But for now it is enough that you know that this NPML module what I have, if I put the value for first second third and fourth component that is going to define the boundaries of the PML and the directions of the PML.

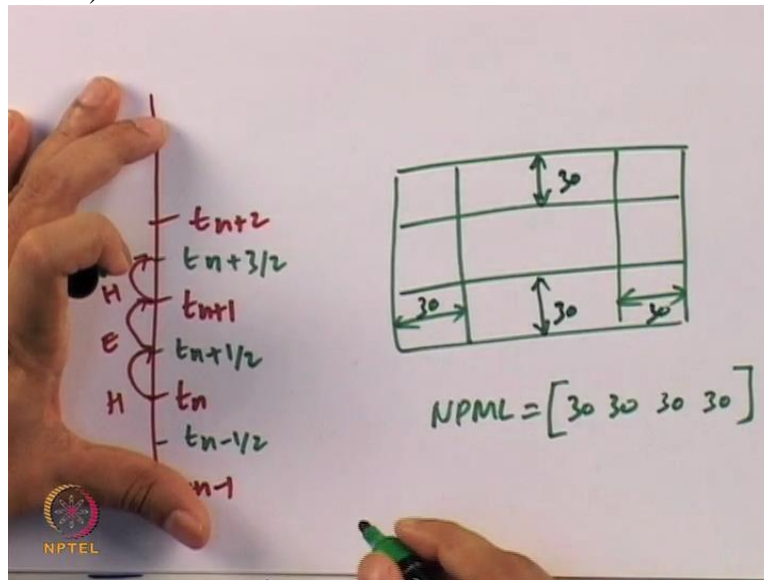
(Refer Slide Time: 20:52)



For example if I have rectangular domain and I am having PML on all the four directions. Then NPML parameter will have components that are going to give me the thickness of the PML in the both minus x direction and plus x direction. So it is going to be if I put value 30 30 that means the thickness of the PML on the left hand side and right hand side is going to be 30 space steps and 30 space steps. So 30 x space steps on the left hand side and 30 x space steps on the right hand side.

Similarly if I put 30 30 also for a third and fourth component it is going to be 30 y space steps and 30 y space steps on both the lower and the upper side of the domain. So if I put this, this is going to be 30, this is going to be 30, this is going to be 30, this is going to be 30 space steps.

(Refer Slide Time: 22:08)



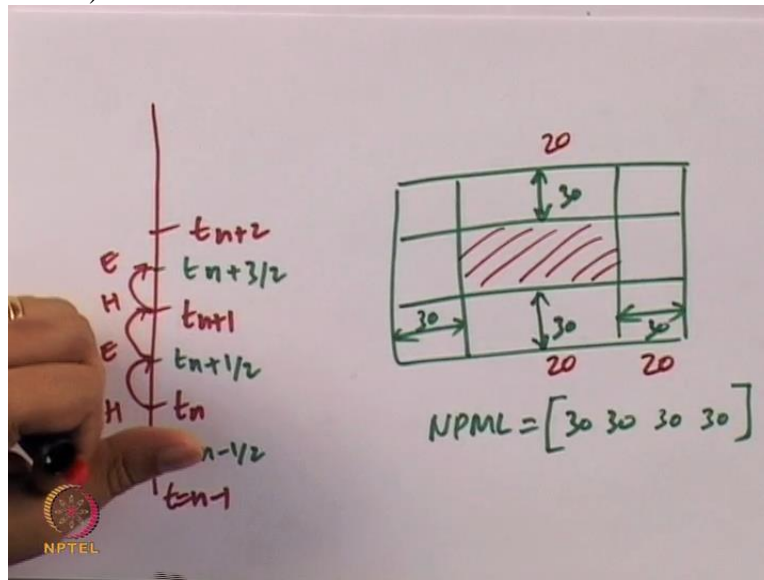
And our computational domain is going to be the one which we are marking in red. Of course I have exaggerated the figure. Your computational domain will be larger than the PML thickness itself but just for the sake of proving the just for the sake of explaining the concept I made it look like this.

(Refer Slide Time: 22:29)

```
47 - R = 0.5;
48 - %CFL stability condition <=0.7 for 2D wave equation
49
50 - dt=R*dx/c;
51 - tsteps=time_tot;
52
53 - % Compute PML parameters
54 - Nx2=2*Nx;
55 - Ny2=2*Ny;
56 - NPML=[0 20 20 20];
57
58 - sigx = zeros(Nx2,Ny2);
59 - for i=1:2*NPML(1)
60 -     i1 = 2*NPML(1) - i + 1;
```

So let us go back to the code what you see is the first component is 0 whereas the last three components are going to have 20 20 20.

(Refer Slide Time: 22:44)



That means coming back to the figure here what you see is the left hand side there is no PML that is why it is 0. Whereas the right hand side it is going to be 20. The lower end it is going to be 20, the upper end it is going to be 20 the thickness of the PML. So that is all you need to know for now about the PML. As I said we will come back and discuss PML at length at the next module. For now we take PML for granted. So all the equations where there is PML I request you to neglect in this exercise and we will look at it at a later stage.

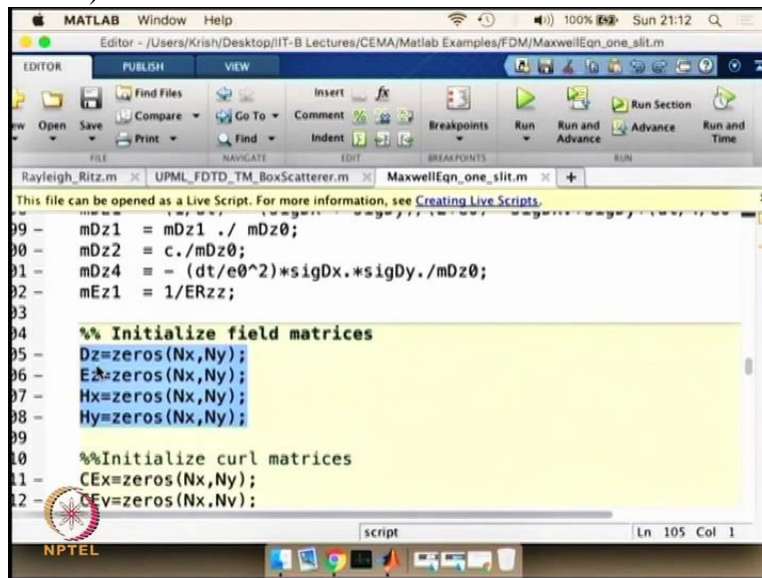
(Refer Slide Time: 23:23)

```
31
32  %% COMPUTE UPDATE COEFFICIENTS
33  sigHx = sigx(1:2:Nx2, 2:2:Ny2);
34  sigHy = sigy(1:2:Nx2, 2:2:Ny2);
35  mHx0 = (1/dt) + sigHy/(2*e0);
36  mHx1 = ((1/dt) - sigHy/(2*e0))./mHx0;
37  mHx2 = - c./URxx./mHx0;
38  mHx3 = - (c*dt/e0) * sigHx./URxx ./ mHx0;
39  sigHx = sigx(2:2:Nx2, 1:2:Ny2);
40  sigHy = sigy(2:2:Nx2, 1:2:Ny2);
41  mHy0 = (1/dt) + sigHx/(2*e0);
42  mHy1 = ((1/dt) - sigHx/(2*e0))./mHy0;
43  mHy2 = - c./URyy./mHy0;
44  mHy3 = - (c*dt/e0) * sigHy./URvv ./ mHy0;
```

So let us go and look at the way the PML components are computed. The PML components are computed accordingly using sigma Hx, sigma Hy. All these components are related to the PML,

we will not look at it at this stage. What is more important is how are we going to update the field equations.

(Refer Slide Time: 23:42)

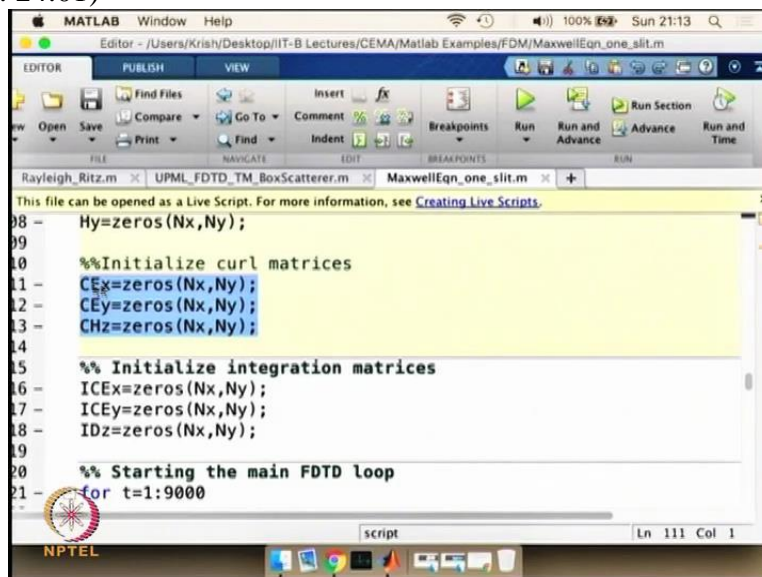


The image shows a MATLAB editor window with the following code:

```
99 - mDz1 = mDz1 ./ mDz0;  
100 - mDz2 = c./mDz0;  
101 - mDz4 = - (dt/e0^2)*sigDx.*sigDy./mDz0;  
102 - mEz1 = 1/ERzz;  
103  
104 % Initialize field matrices  
105 - Dz=zeros(Nx,Ny);  
106 - Ez=zeros(Nx,Ny);  
107 - Hx=zeros(Nx,Ny);  
108 - Hy=zeros(Nx,Ny);  
109  
110 %%Initialize curl matrices  
111 - CEx=zeros(Nx,Ny);  
112 - CEy=zeros(Nx,Ny);
```

As you can see here, there are going to be four field components, Dz, Ez, Hx, Hy. Of course we could have just done it with only three components. In order to make the computational coding easier we have introduced another component Dz.

(Refer Slide Time: 24:01)




The image shows a MATLAB editor window with the following code:

```
108 - Hy=zeros(Nx,Ny);  
109  
110 %%Initialize curl matrices  
111 - CEx=zeros(Nx,Ny);  
112 - CEy=zeros(Nx,Ny);  
113 - CHz=zeros(Nx,Ny);  
114  
115 %% Initialize integration matrices  
116 - ICEx=zeros(Nx,Ny);  
117 - ICEy=zeros(Nx,Ny);  
118 - IDz=zeros(Nx,Ny);  
119  
120 %% Starting the main FDTD loop  
121 - for t=1:9000
```



As you can see here we have three curl components. So the curl Ex component, the curl Ey component, curl Hz component. And these are the components that I have described to you while we were discussing the initial stage.

(Refer Slide Time: 24:21)

$$\mu \frac{\partial H_x}{\partial t} = -(\nabla \times E)_x = -\partial_y E_z$$
$$\mu \frac{\partial H_y}{\partial t} = -(\nabla \times E)_y = \partial_x E_z$$
$$\begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ \partial_x & \partial_y & \partial_z \\ 0 & 0 & E_z \end{vmatrix} = \begin{vmatrix} \hat{x} (\partial_y E_z) \\ \hat{y} (-\partial_x E_z) \\ 0 \end{vmatrix}$$


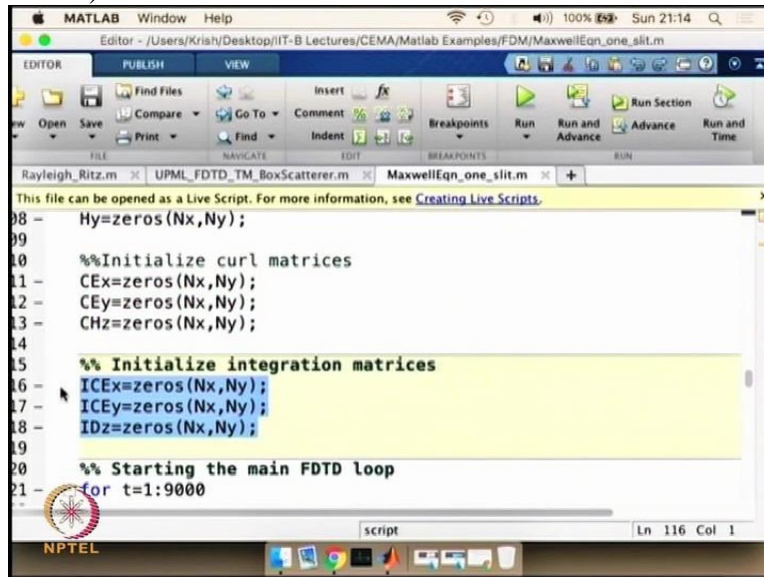
So let us look back those components here. So these are the curl E_x component, curl E_y component,

(Refer Slide Time: 24:37)

$$\epsilon \frac{\partial E_z}{\partial t} = (\nabla \times H)_z = \partial_x H_y - \partial_y H_x$$
$$\begin{vmatrix} \hat{x} & \hat{y} & \hat{z} \\ \partial_x & \partial_y & 0 \\ H_x & H_y & 0 \end{vmatrix} = \hat{z} (\partial_x H_y - \partial_y H_x)$$


And the curl z component is written separately here. So the curl H_z component is here.

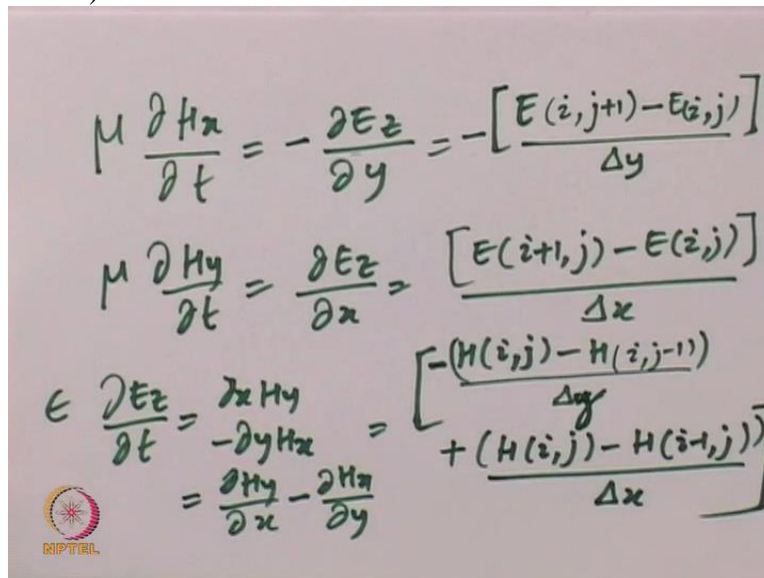
(Refer Slide Time: 24:41)



```
08 - Hy=zeros(Nx,Ny);
09
10 %%Initialize curl matrices
11 - CEx=zeros(Nx,Ny);
12 - CEy=zeros(Nx,Ny);
13 - CHz=zeros(Nx,Ny);
14
15 %% Initialize integration matrices
16 - ICEx=zeros(Nx,Ny);
17 - ICEy=zeros(Nx,Ny);
18 - IDz=zeros(Nx,Ny);
19
20 %% Starting the main FDTD loop
21 - for t=1:9000
```

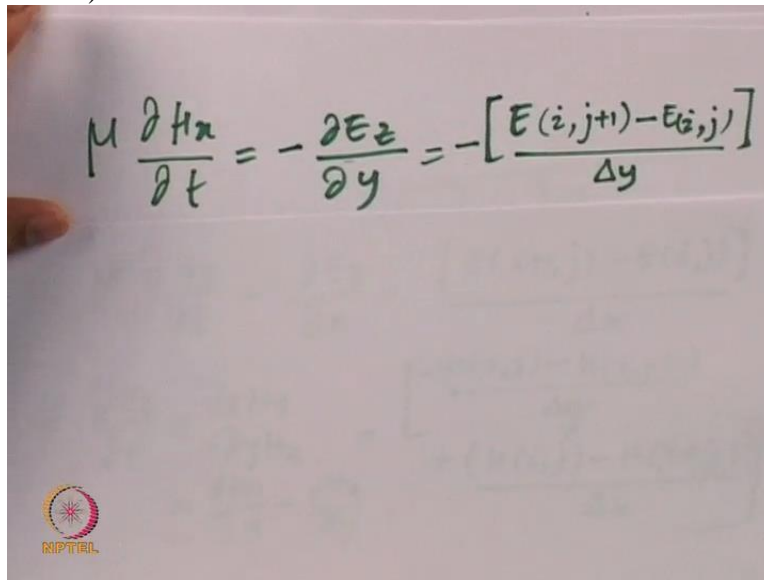
So these are the terms that we are going to use. So let us get back to the code. We are initializing these values here as zeroes. And these are the integral terms. So the integral terms are nothing but the terms that we are going to use in the update equation itself. So I have described that very shortly when we did the update equation.

(Refer Slide Time: 25:06)


$$\begin{aligned} \mu \frac{\partial H_x}{\partial t} &= -\frac{\partial E_z}{\partial y} = -\left[\frac{E(i,j+1) - E(i,j)}{\Delta y} \right] \\ \mu \frac{\partial H_y}{\partial t} &= \frac{\partial E_z}{\partial x} = \left[\frac{E(i+1,j) - E(i,j)}{\Delta x} \right] \\ \epsilon \frac{\partial E_z}{\partial t} &= \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} = \left[\frac{-(H(i,j) - H(i,j-1))}{\Delta y} + \frac{(H(i,j) - H(i-1,j))}{\Delta x} \right] \end{aligned}$$

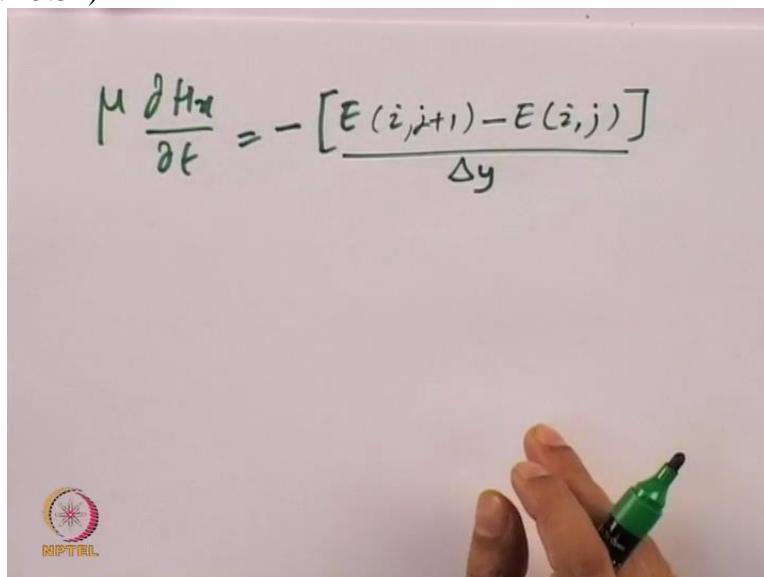
So Let us go back and look at the update equation one more time. So let us take one of the update equation and expand it in a way that it makes sense, for us to see the comparison what we have in the Matlab code.

(Refer Slide Time: 25:18)


$$\mu \frac{\partial H_x}{\partial t} = -\frac{\partial E_z}{\partial y} = -\left[\frac{E(z, j+1) - E(z, j)}{\Delta y}\right]$$

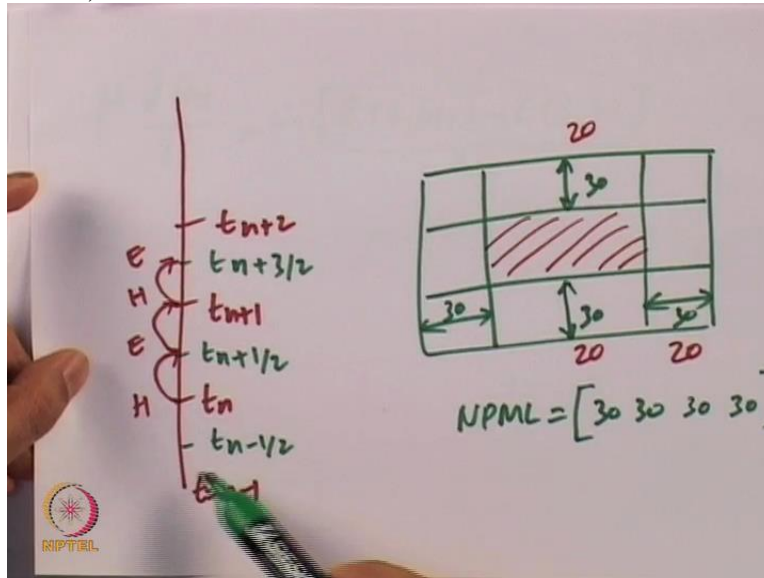
So I am going to take the first equation, and I will not do the other two. The other two you can follow through the logic what we are discussing for the first equation.

(Refer Slide Time: 25:32)


$$\mu \frac{\partial H_x}{\partial t} = -\left[\frac{E(z, j+1) - E(z, j)}{\Delta y}\right]$$

So the first equation is written as $\mu \frac{\partial H_x}{\partial t} = -\left[\frac{E(z, j+1) - E(z, j)}{\Delta y}\right]$

(Refer Slide Time: 26:00)



We are going to compute E at different time steps compared to the value we are going to compute H. So H we are computing at every n and E we are computing at every half steps.

(Refer Slide Time: 26:15)

$$\mu \frac{\partial H_z}{\partial t} = - \frac{[E(i,j+1) - E(i,j)]}{\Delta y}$$

$$\frac{H_z^{n+1} - H_z^n}{\Delta t} = - \frac{1}{\mu} \frac{[E(i,j+1)^{n+1/2} - E(i,j)^{n+1/2}]}{\Delta y}$$

So here we have to write that and this will become clear when we expand this, so this is going to be, I am going to take the Mu on the other side, and I am going to write $H_z^{n+1} - H_z^n$ divided by delta t is equal to minus 1 by Mu $[E(i,j+1)^{n+1/2} - E(i,j)^{n+1/2}]$ divided by delta y. And here they are going from n to n plus 1, so the E value will be at time step n plus 1 by 2 { minus 1 by Mu $[E^{n+1/2}(i,j+1) - E^{n+1/2}(i,j)]$ } divided by delta y. So this is the half time step where we are computing E.

(Refer Slide Time: 27:15)

$$\mu \frac{\partial H_x}{\partial t} = - \left[\frac{E(z, j+1) - E(z, j)}{\Delta y} \right]$$
$$\frac{H_x^{n+1} - H_x^n}{\Delta t} = - \frac{1}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$
$$H_x^{n+1} = H_x^n - \frac{\Delta t}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$

The equations are written in black ink on a whiteboard. A small NPTEL logo is visible in the bottom left corner of the whiteboard image.

So now when I rearrange the term what I get for H_x^{n+1} is equal to H_x^n (I take the x on the other side) minus Δt by $\mu [E_x^{n+1/2}(i, j+1) - E_x^{n+1/2}(i, j)]$ divided by Δy . So this is the update equation that we will be using for H_x component.

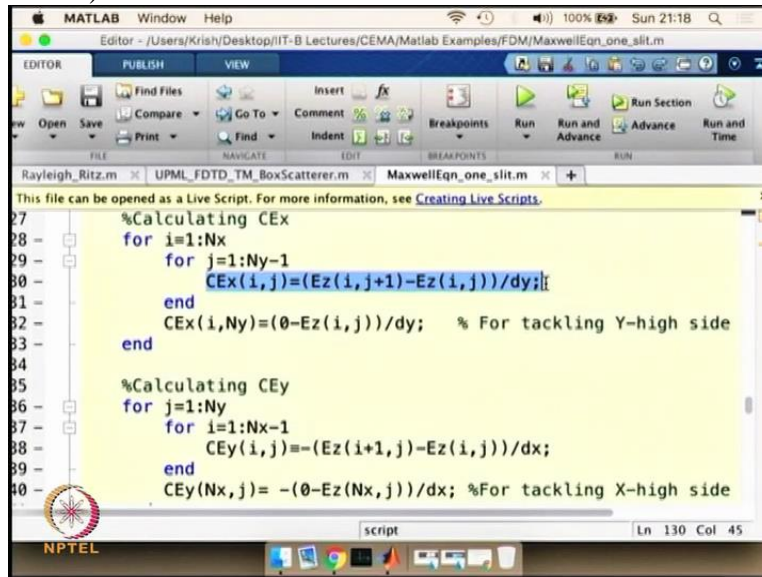
(Refer Slide Time: 28:00)

$$\mu \frac{\partial H_x}{\partial t} = - \left[\frac{E(z, j+1) - E(z, j)}{\Delta y} \right]$$
$$\frac{H_x^{n+1} - H_x^n}{\Delta t} = - \frac{1}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$
$$H_x^{n+1} = H_x^n - \frac{\Delta t}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$

The equations are written in black ink on a whiteboard. The final equation is enclosed in a red rectangular box, and the text "Curl term" is written in red below it. A small NPTEL logo is visible in the bottom left corner of the whiteboard image.

Similarly you can derive for H_y or E_z component. So here the integral term what we are talking about are the terms what we are going to keep adding up. So this is going to be the curl term that we are going to add up. So this is this component will be the curl term. And every time we compute the curl term we keep adding it, we will get the integral term.

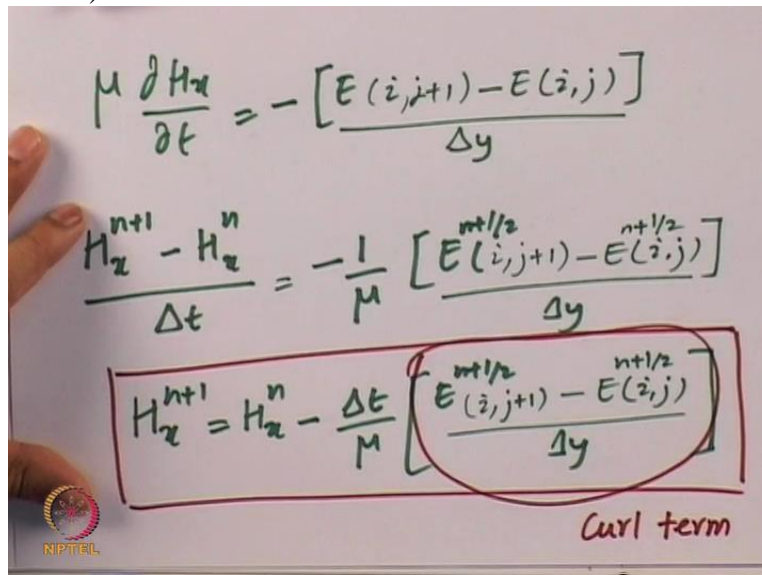
(Refer Slide Time: 28:34)



```
27 %Calculating CEx
28 for i=1:Nx
29     for j=1:Ny-1
30         CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;
31     end
32     CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side
33 end
34
35 %Calculating CEy
36 for j=1:Ny
37     for i=1:Nx-1
38         CEy(i,j)=-(Ez(i+1,j)-Ez(i,j))/dx;
39     end
40     CEy(Nx,j)= -(0-Ez(Nx,j))/dx; %For tackling X-high side
```

So let us look at the curl of E, So this is going to be the $E_z(i,j+1)$ plus 14 minus $E_z(i,j)$ divided by Δy . This is exactly what we have assumed in the equation what we have here.

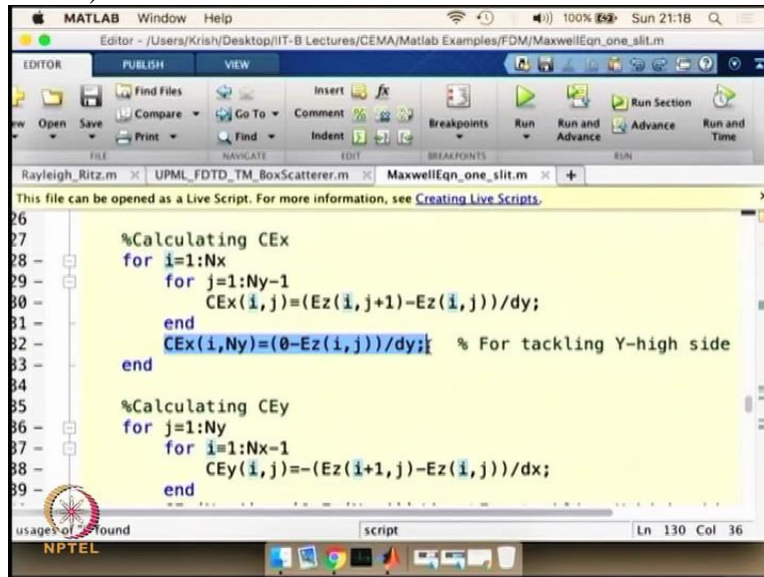
(Refer Slide Time: 28:52)


$$\mu \frac{\partial H_x}{\partial t} = - \frac{[E_z(i,j+1) - E_z(i,j)]}{\Delta y}$$
$$\frac{H_x^{n+1} - H_x^n}{\Delta t} = - \frac{1}{\mu} \frac{[E_z^{n+1/2}(i,j+1) - E_z^{n+1/2}(i,j)]}{\Delta y}$$
$$H_x^{n+1} = H_x^n - \frac{\Delta t}{\mu} \left[\frac{E_z^{n+1/2}(i,j+1) - E_z^{n+1/2}(i,j)}{\Delta y} \right]$$

Curl term

If you see on the paper so this is the value what we have set on the equation in the Matlab file here.

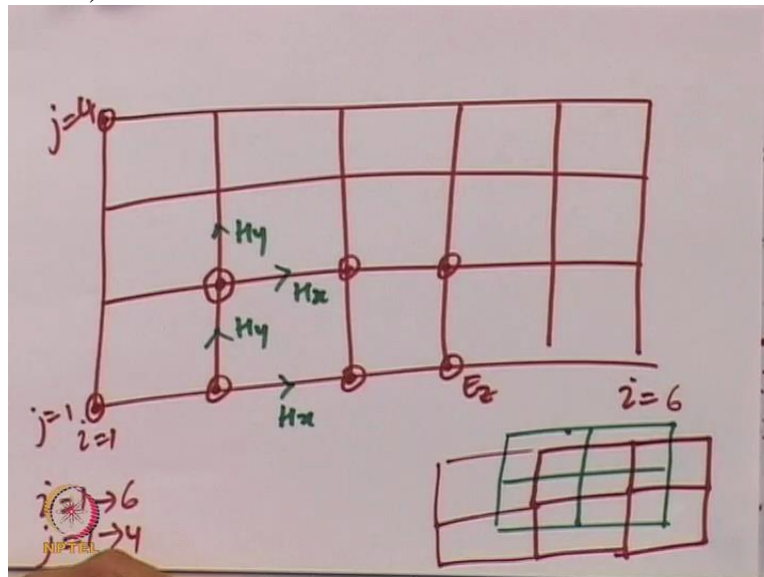
(Refer Slide Time: 29:02)



```
26
27 %Calculating CEx
28 for i=1:Nx
29     for j=1:Ny-1
30         CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;
31     end
32     CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side
33 end
34
35 %Calculating CEy
36 for j=1:Ny
37     for i=1:Nx-1
38         CEy(i,j)=-(Ez(i+1,j)-Ez(i,j))/dx;
39     end
```

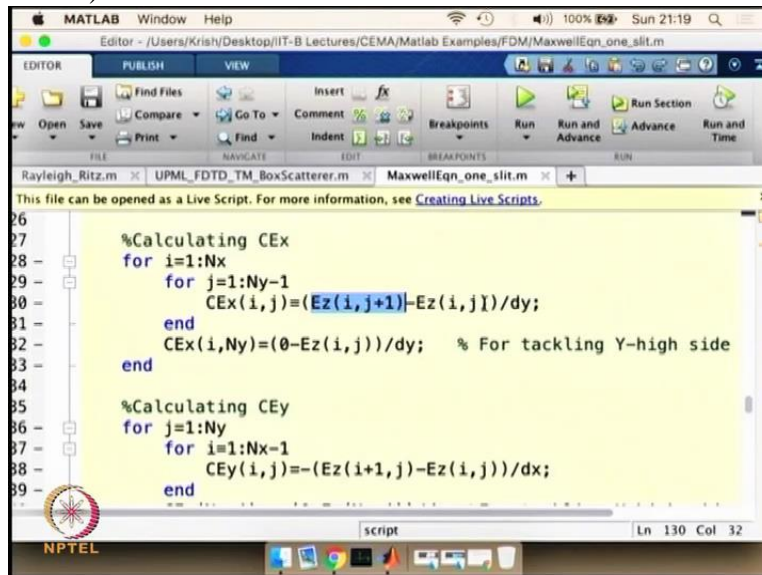
And we are also making sure the boundary conditions are satisfied.

(Refer Slide Time: 29:10)



So let us look back at our computational grid. So E is going to go on the node values as I said. so on the upper side of the domain you donot have enough points for you to compute j plus 1. So the j plus 1 component does not exist. It goes only until the J. The j plus 1 component does not exist.

(Refer Slide Time: 29:32)

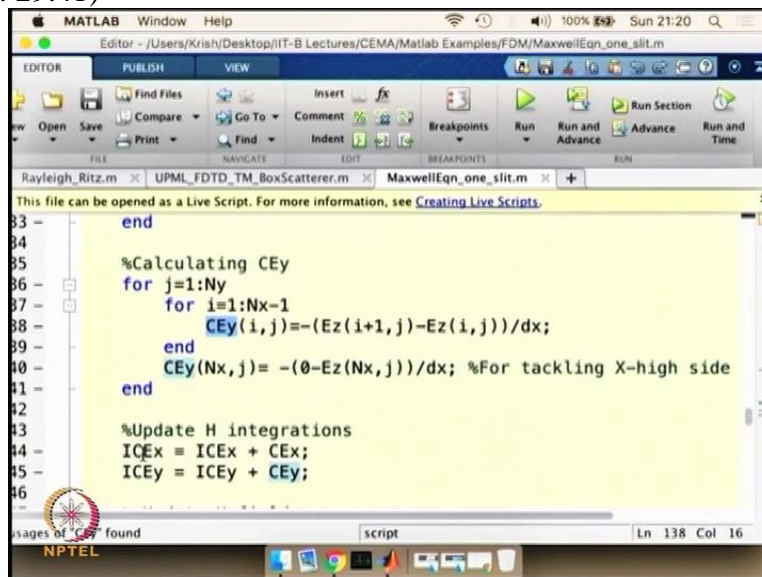


The image shows a MATLAB editor window with the following code:

```
26  
27 %Calculating CEx  
28 for i=1:Nx  
29     for j=1:Ny-1  
30         CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;  
31     end  
32     CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side  
33 end  
34  
35 %Calculating CEy  
36 for j=1:Ny  
37     for i=1:Nx-1  
38         CEy(i,j)=-(Ez(i+1,j)-Ez(i,j))/dx;  
39     end
```

That is why for this particular case we are putting j plus 1 component as 0 and $E(i,j)$ exist as such.

(Refer Slide Time: 29:41)



The image shows a MATLAB editor window with the following code:

```
33 end  
34  
35 %Calculating CEy  
36 for j=1:Ny  
37     for i=1:Nx-1  
38         CEy(i,j)=-(Ez(i+1,j)-Ez(i,j))/dx;  
39     end  
40     CEy(Nx,j)= -(0-Ez(Nx,j))/dx; %For tackling X-high side  
41 end  
42  
43 %Update H integrations  
44 ICEx = ICEx + CEx;  
45 ICEy = ICEy + CEy;  
46
```

Similarly the other components of the Maxwell field in 2D case so this is going to be the curl of E_y component.

(Refer Slide Time: 29:50)

```
MATLAB Window Help
Editor - /Users/Krish/Desktop/IIIT-B Lectures/CEMA/Matlab Examples/FDM/MaxwellEqn_one_slit.m
EDITOR PUBLISH VIEW
Find Files Insert
Open Save Compare Go To Comment Breakpoints Run Run and Advance Run Section Run and Time
Print Find Indent Breakpoints Run Run and Advance Advance Run and Time
FILE NAVIGATE EDIT BREAKPOINTS RUN
Rayleigh_Ritz.m UPML_FDTD_TM_BoxScatterer.m MaxwellEqn_one_slit.m
This file can be opened as a Live Script. For more information, see Creating Live Scripts.
39 - end
40 -     CEy(Nx,j) = -(0-Ez(Nx,j))/dx; %For tackling X-high side
41 - end
42
43 %Update H integrations
44 - ICEx = ICEx + CEx;
45 - ICEy = ICEy + CEy;
46
47 % Update H fields
48 - Hx = mHx1.*Hx + mHx2.*CEx + mHx3.*ICEx;
49 - Hy = mHy1.*Hy + mHy2.*CEy + mHy3.*ICEy;
50
51 % Compute CHz
52 % Curl equations automatically include PEC BC
script Ln 144 Col 2
NPTEL
```

And once we have that we are going to update that into the integration term. So ICEx is the integration component of the curl, so it keeps adding up.

(Refer Slide Time: 30:07)

```
MATLAB Window Help
Editor - /Users/Krish/Desktop/IIIT-B Lectures/CEMA/Matlab Examples/FDM/MaxwellEqn_one_slit.m
EDITOR PUBLISH VIEW
Find Files Insert
Open Save Compare Go To Comment Breakpoints Run Run and Advance Run Section Run and Time
Print Find Indent Breakpoints Run Run and Advance Advance Run and Time
FILE NAVIGATE EDIT BREAKPOINTS RUN
Rayleigh_Ritz.m UPML_FDTD_TM_BoxScatterer.m MaxwellEqn_one_slit.m
This file can be opened as a Live Script. For more information, see Creating Live Scripts.
39 - end
40 -     CEy(Nx,j) = -(0-Ez(Nx,j))/dx; %For tackling X-high side
41 - end
42
43 %Update H integrations
44 - ICEx = ICEx + CEx;
45 - ICEy = ICEy + CEy;
46
47 % Update H fields
48 - Hx = mHx1.*Hx + mHx2.*CEx + mHx3.*ICEx;
49 - Hy = mHy1.*Hy + mHy2.*CEy + mHy3.*ICEy;
50
51 % Compute CHz
52 % Curl equations automatically include PEC BC
script Ln 148 Col 44
NPTEL
```

And once we have that into the equation we are able to find out the value of Hx and Hy using this particular form.

(Refer Slide Time: 30:20)

```

35 - mHx0 = (1/dt) + sigHy/(2*e0);
36 - mHx1 = ((1/dt) - sigHy/(2*e0))./mHx0;
37 - mHx2 = - c./URxx./mHx0;
38 - mHx3 = - (c*dt/e0) * sigHx./URxx ./ mHx0;
39 - sigHx = sigx(2:2:Nx2,1:2:Ny2);
40 - sigHy = sigy(2:2:Nx2,1:2:Ny2);
41 - mHy0 = (1/dt) + sigHx/(2*e0);
42 - mHy1 = ((1/dt) - sigHx/(2*e0))./mHy0;
43 - mHy2 = - c./URyy./mHy0;
44 - mHy3 = - (c*dt/e0) * sigHy./URyy ./ mHy0;
45 - sigDx = sigx(1:2:Nx2,1:2:Ny2);
46 - sigDy = sigy(1:2:Nx2,1:2:Ny2);
47 - mDz0 = (1/dt) + (sigDx + sigDy)/(2*e0) + sigDx.*sigDy*(dt/4/e0^
48 - mDz1 = (1/dt) - (sigDx + sigDy)/(2*e0) - sigDx.*sigDy*(dt/4/e0^
  
```

The value of mHx1 is defined in the previous steps. mHx1 is going to be 1 by dt minus sigma Hy divided by 2Epsilon 0.

So If you see when you put sigma Hy, and sigma Hx and all these components to be 0. They will reduce to the simple update equations that we have derived just now. So this is going to be the same equations what we have on the paper.

(Refer Slide Time: 30:59)

$$\mu \frac{\partial H_x}{\partial t} = - \left[\frac{E(z, j+1) - E(z, j)}{\Delta y} \right]$$

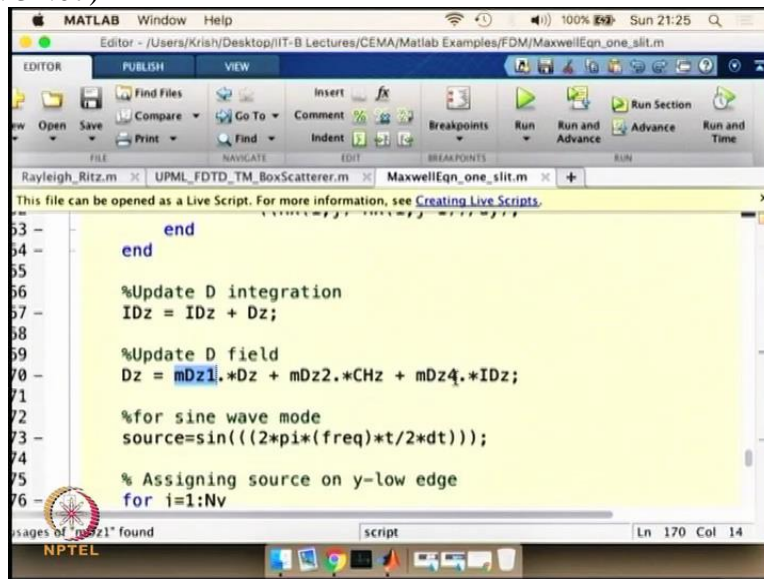
$$\frac{H_x^{n+1} - H_x^n}{\Delta t} = - \frac{1}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$

$$H_x^{n+1} = H_x^n - \frac{\Delta t}{\mu} \left[\frac{E^{n+1/2}(z, j+1) - E^{n+1/2}(z, j)}{\Delta y} \right]$$

Curl term

It is going to be delta t by Mu and so we are not worried about the PML factors here.

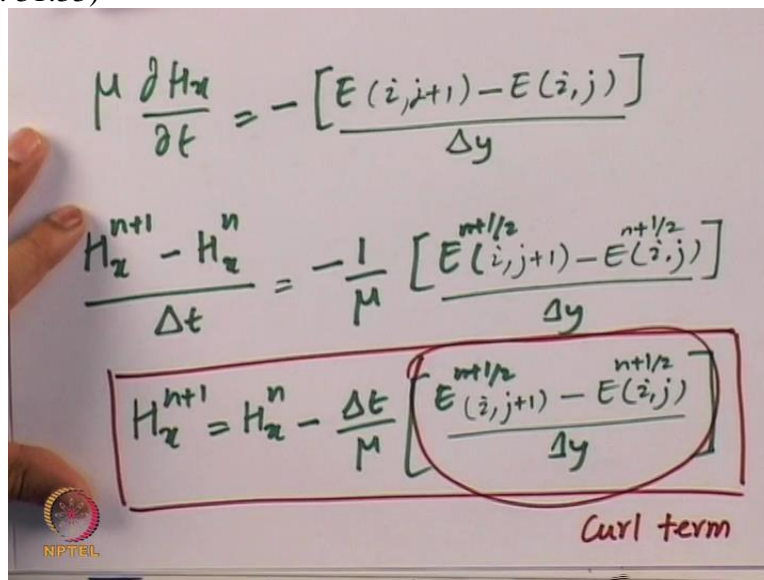
(Refer Slide Time: 31:07)



```
53 -         end
54 -     end
55 -
56 -     %Update D integration
57 -     IDz = IDz + Dz;
58 -
59 -     %Update D field
60 -     Dz = mDz1.*Dz + mDz2.*CHz + mDz4.*IDz;
61 -
62 -     %for sine wave mode
63 -     source=sin(((2*pi*(freq)*t)/2*dt));
64 -
65 -     % Assigning source on y-low edge
66 -     for i=1:Nv
```

So the update equation what we are getting here is similar to the equation what we have here. The reason for having this mDz1 mDz2 mDz4 is to make sure that the PML conditions are satisfied. But other than that this is nothing but the simple update equations what we have described here on the paper.

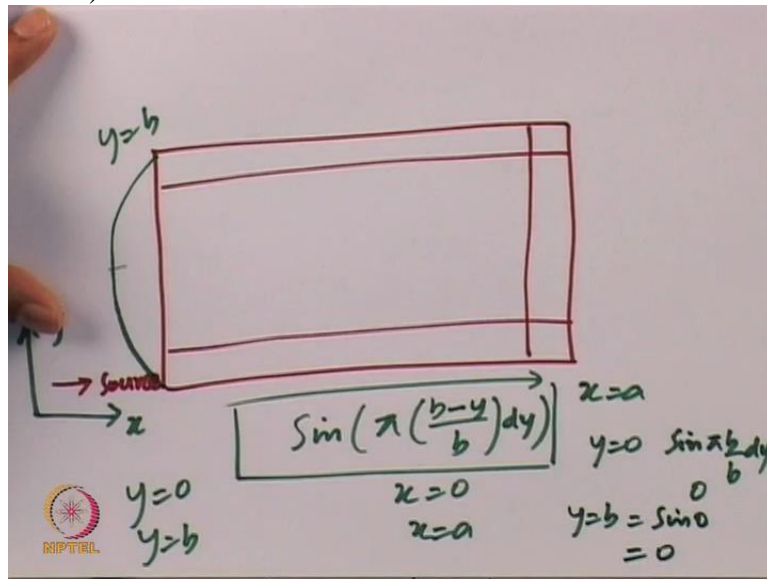
(Refer Slide Time: 31:33)


$$\mu \frac{\partial H_x}{\partial t} = - \frac{[E(z, j+1) - E(z, j)]}{\Delta y}$$
$$\frac{H_x^{n+1} - H_x^n}{\Delta t} = - \frac{1}{\mu} \frac{[E(z, j+1)^{n+1/2} - E(z, j)^{n+1/2}]}{\Delta y}$$
$$H_x^{n+1} = H_x^n - \frac{\Delta t}{\mu} \left[\frac{E(z, j+1)^{n+1/2} - E(z, j)^{n+1/2}}{\Delta y} \right]$$

Curl term

So we are now going to test this particular code for a particular excitation that we are going to have. So what we are going to have is. We are going to start with a simple sinusoidal source. And we are going to modulate this source using a mode. So I am going to explain that step by step on the paper.

(Refer Slide Time: 31:59)



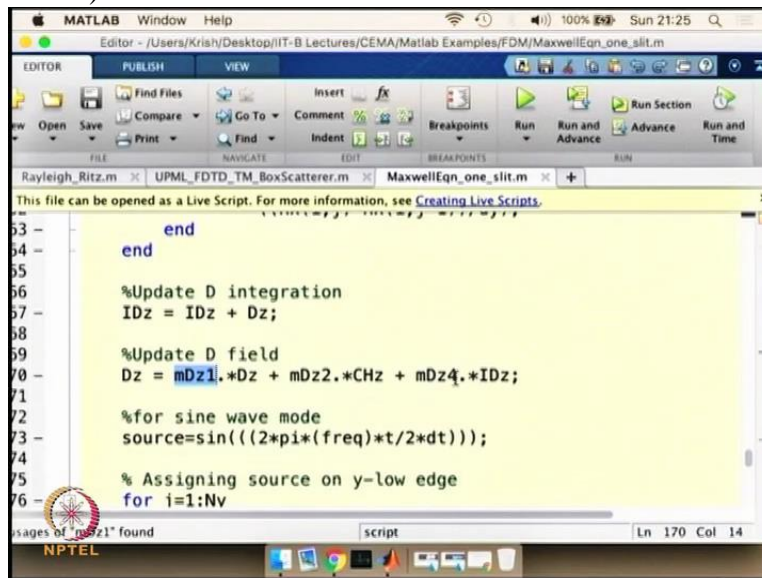
Boundary is going to have PMLs on the three sides. There is going to be a PML here. There is going to be a PML here, There is going to be a PML here, and these are all uniaxial PMLs and this is the direction of the source. The source is going to have both sinusoidal factor and a mode factor. So it is going to have maximum at the middle and minimum at both edges. This is we are going to achieve by using the mode function. So what we are going to have is this is the y direction this is the x direction and from the y direction we are going to say it is going to behave with a function $\text{Sin}(\pi(b - y) / b) dy$ $x = a$.

So the mode is going to have a profile on the y direction. So our y goes from y equal to 0 to y equal to b. And our x goes from x equal to 0 to x equal to a. And I said it is going to have a mode on the y direction. And the middle will be the maximum and we are achieving this by using this particular equation and we are multiplying this equation with the source function.

So this equation will have two functions you will see when you put y equal to 0. This equation will have a y equal to 0. This will become $\text{sin}(\pi b / b \text{ multiplied by } dy)$, which is going to be 0; side of π is 0 and when y equal to b this is equal to b minus b which is 0. So $\text{sin } 0$ will be again

0. So it is going to have 0 at both the ends of the domain and it is going to have maximum at the middle.

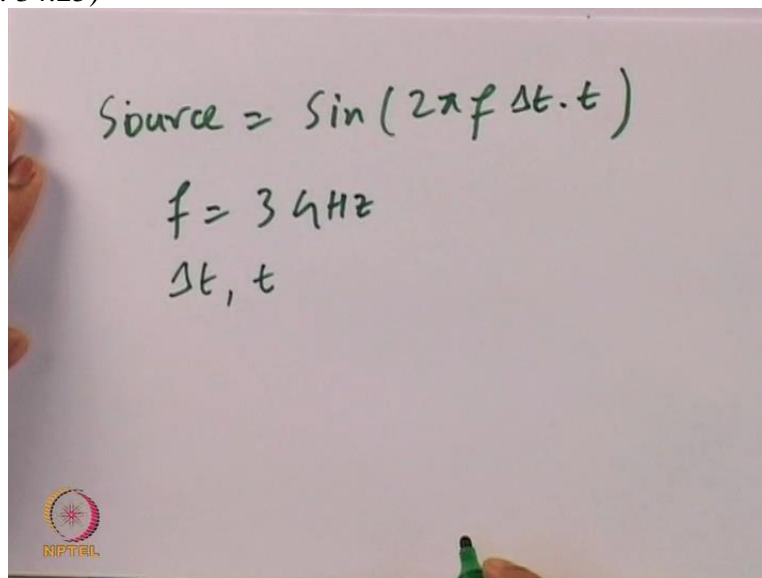
(Refer Slide Time: 34:16)



```
53 - end
54 - end
55 -
56 - %Update D integration
57 - IDz = IDz + Dz;
58 -
59 - %Update D field
60 - Dz = mDz1.*Dz + mDz2.*CHz + mDz4.*IDz;
61 -
62 - %for sine wave mode
63 - source=sin((2*pi*(freq)*t/2*dt));
64 -
65 - % Assigning source on y-low edge
66 - for i=1:Nv
```

So the source is going to be 3 Giga Hertz, so we are going to write the source as also sin function.

(Refer Slide Time: 34:25)


$$\text{Source} = \sin(2\pi f \Delta t \cdot t)$$
$$f = 3 \text{ GHz}$$
$$\Delta t, t$$

So the source function is going to be given by $\sin(2\pi f \Delta t \cdot t)$ multiplied by the time step t itself). So this is f is going to be 3 Giga Hertz. And Δt and t are computed accordingly to the equation. And we can substitute that into this particular formula. We will get the source.

(Refer Slide Time: 35:00)

Source
 $D_z(i,j) \Rightarrow D_z(1,j) = \sin\left(\pi \frac{b-y}{b} \Delta y\right) \times \sin(2\pi f \cdot t \cdot \Delta t)$

(1,j)

The image shows a whiteboard with a handwritten equation and a diagram. The equation is $D_z(i,j) \Rightarrow D_z(1,j) = \sin\left(\pi \frac{b-y}{b} \Delta y\right) \times \sin(2\pi f \cdot t \cdot \Delta t)$. The word "Source" is written above the equation. Below the equation is a hand-drawn rectangle representing a grid. The bottom-left corner of the rectangle is labeled "(1,j)".

So the D_z value is given by (i,j) and we are setting the source exactly at the first line. So this is going to be $1,j$ line. So we are going to set for the source $D_z(1,j)$ is equal to the mode function. So $\sin(\pi (b \text{ minus } y \text{ divided by } b) \Delta y)$ multiplied by the source function which is $\sin(2 \pi f t \text{ multiplied by } \Delta t)$.

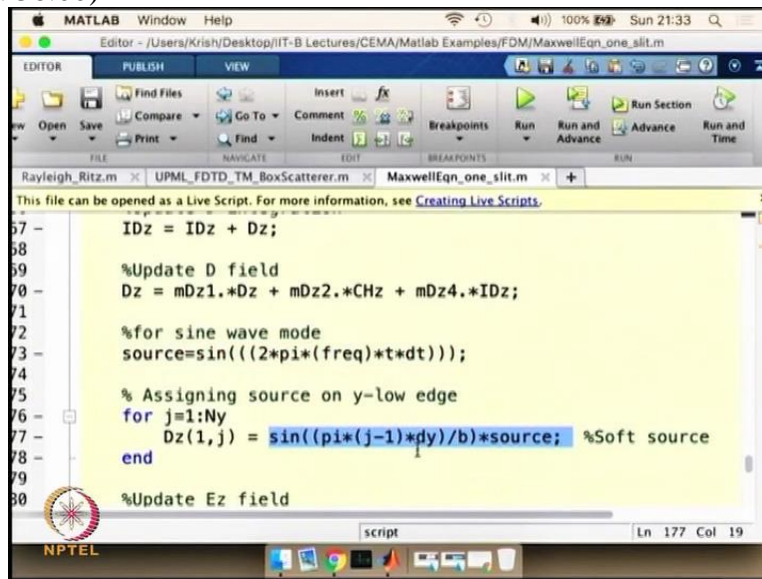
(Refer Slide Time: 35:50)

```
57 - IDz = IDz + Dz;
58
59 - %Update D field
60 - Dz = mDz1.*Dz + mDz2.*CHz + mDz4.*IDz;
61
62 - %for sine wave mode
63 - source=sin(((2*pi*(freq)*t*dt)));
64
65 - % Assigning source on y-low edge
66 - for j=1:Ny
67 -     Dz(1,j) = sin((pi*(j-1)*dy)/b)*source; %Soft source
68 - end
69
70 - %Update Ez field
```

The image shows a MATLAB code editor window. The code is as follows:

So once that is understood we will see here in this equation that this is the value we are feeding in.

(Refer Slide Time: 36:00)

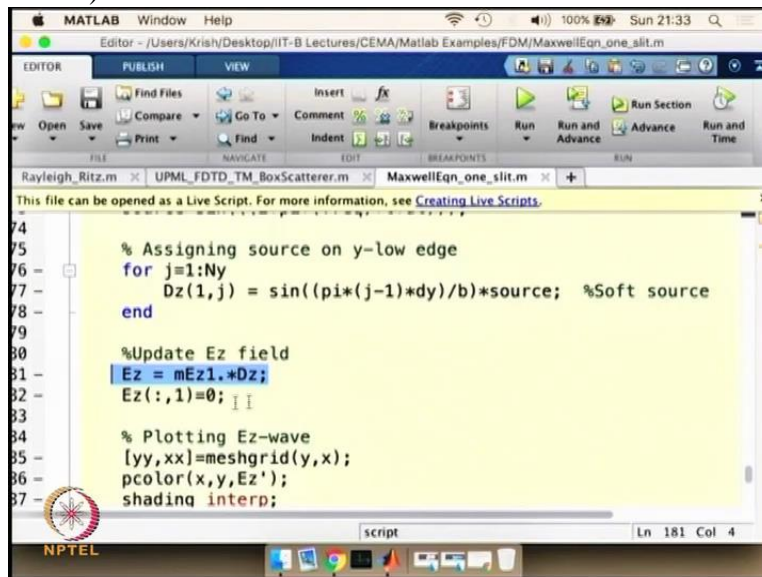


The image shows a MATLAB editor window with the following code:

```
57 - IDz = IDz + Dz;
58
59 %Update D field
60 - Dz = mDz1.*Dz + mDz2.*CHz + mDz4.*IDz;
61
62 %for sine wave mode
63 - source=sin((2*pi*(freq)*t*dt));
64
65 % Assigning source on y-low edge
66 - for j=1:Ny
67 -     Dz(1,j) = sin((pi*(j-1)*dy)/b)*source; %Soft source
68 - end
69
70 %Update Ez field
```

And we are multiplying it using the mode function as I explained.

(Refer Slide Time: 36:04)

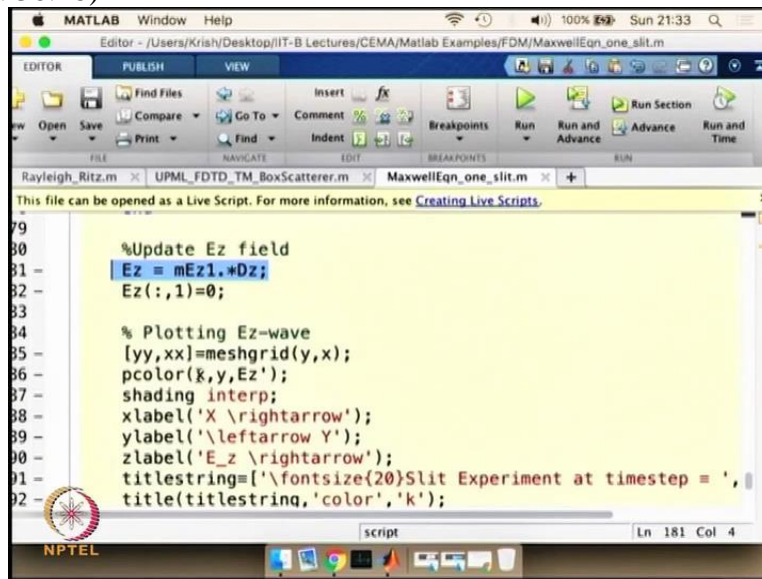


The image shows a MATLAB editor window with the following code:

```
74
75 % Assigning source on y-low edge
76 - for j=1:Ny
77 -     Dz(1,j) = sin((pi*(j-1)*dy)/b)*source; %Soft source
78 - end
79
80 %Update Ez field
81 - Ez = mEz1.*Dz;
82 - Ez(:,1)=0;
83
84 % Plotting Ez-wave
85 - [yy,xx]=meshgrid(y,x);
86 - pcolor(x,y,Ez');
87 - shading interp;
```

And we are updating the Ez field from the source field. The Dz field is just used for simplifying the mathematics. You can also directly do that with Ez field. But we have used here Dz field.

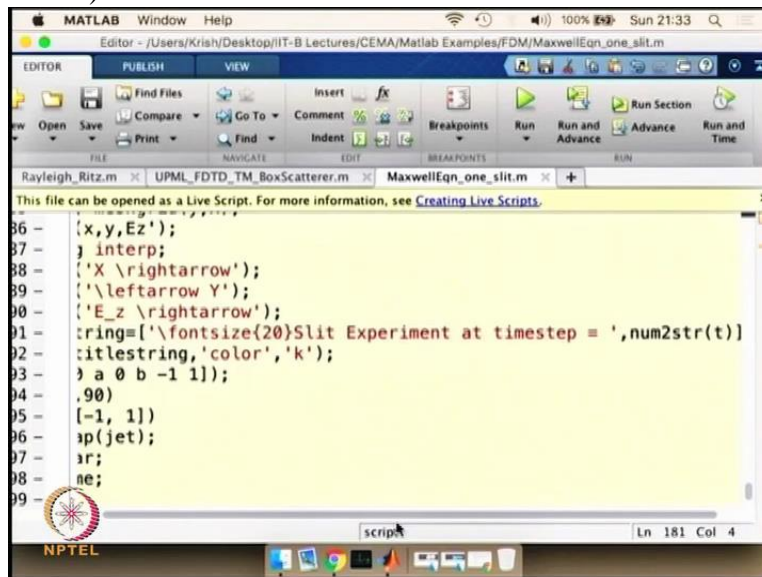
(Refer Slide Time: 36:18)



```
79
80 %Update Ez field
81 Ez = mEz1.+Dz;
82 Ez(:,1)=0;
83
84 % Plotting Ez-wave
85 [yy,xx]=meshgrid(y,x);
86 pcolor(x,y,Ez');
87 shading interp;
88 xlabel('X \rightarrow');
89 ylabel('\leftarrow Y');
90 zlabel('E_z \rightarrow');
91 titlestrng=['\fontsize{20}Slit Experiment at timestep = ',
92 title(titlestrng,'color','k');
```

And now we are going to plot the Ez field for x and y.

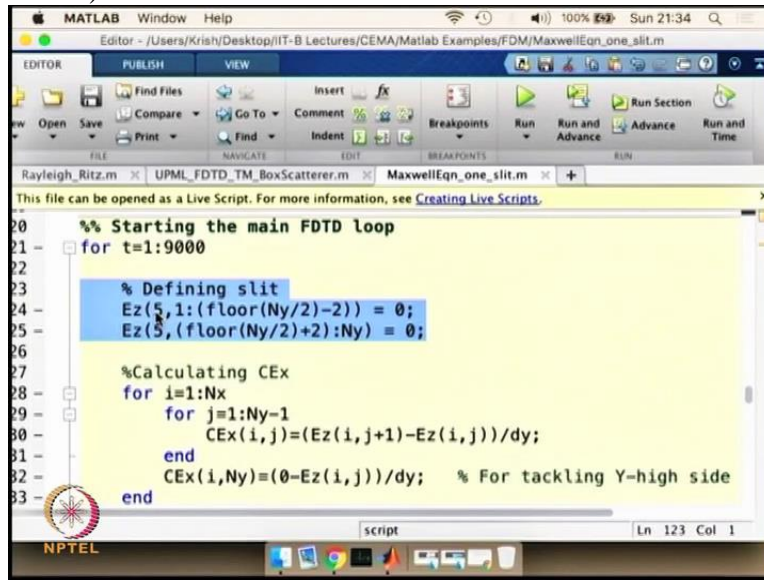
(Refer Slide Time: 36:27)



```
86 [x,y,Ez'];
87 } interp;
88 {'X \rightarrow'};
89 {'\leftarrow Y'};
90 {'E_z \rightarrow'};
91 trng=['\fontsize{20}Slit Experiment at timestep = ',num2str(t)]
92 titlestrng,'color','k');
93 } a 0 b -1 1];
94 .90)
95 [-1, 1])
96 sp(jet);
97 ar;
98 ne;
99
```

And we are going to simulate it using the slit experiment.

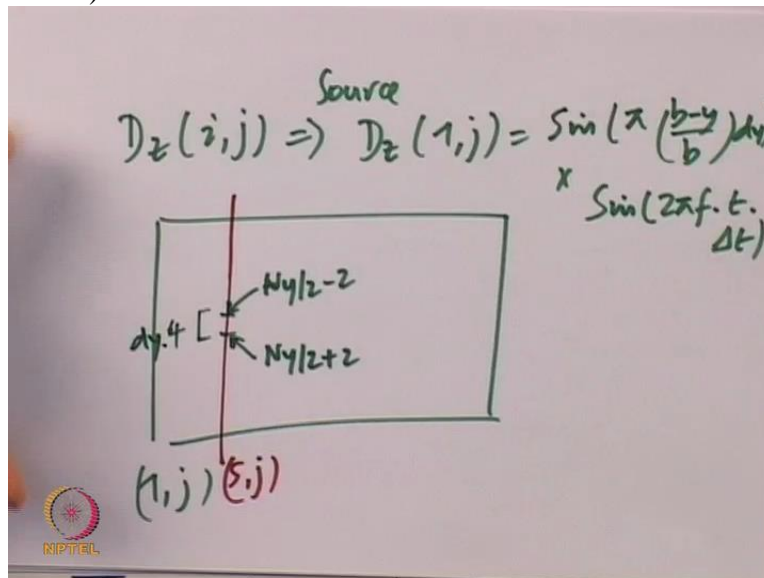
(Refer Slide Time: 36:35)



```
20 % Starting the main FDTD loop
21 for t=1:9000
22
23 % Defining slit
24 Ez(5,1:(floor(Ny/2)-2)) = 0;
25 Ez(5,(floor(Ny/2)+2):Ny) = 0;
26
27 %Calculating CEx
28 for i=1:Nx
29     for j=1:Ny-1
30         CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;
31     end
32     CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side
33 end
```

And the slit location is going to be at the point defined by these two steps. So it is going to be a point that is between $Ny/2$ minus 2 and $Ny/2$ plus 2. So that is going to be four cells minus 2 to plus 2 is going to be the four cells where we are going to keep the slit. And it is going to be at the line x equal to 5.

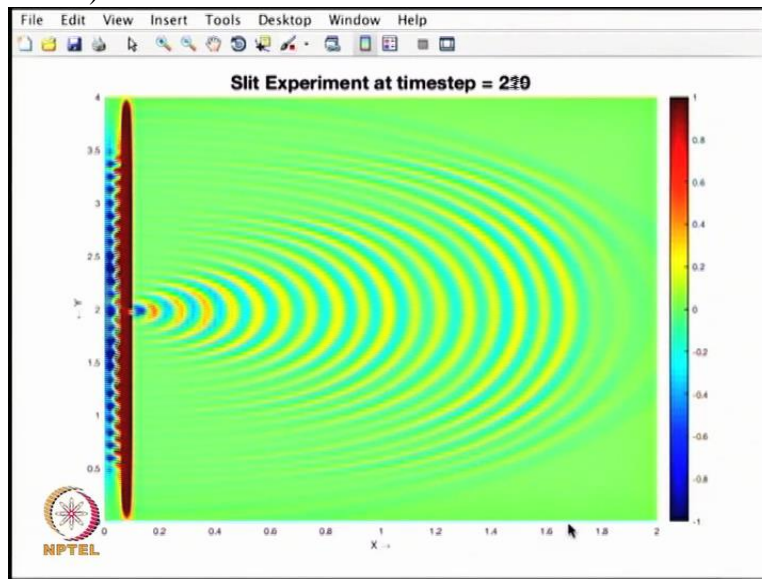
(Refer Slide Time: 37:04)



So if this is x equal to 1, so this is going to be $5,j$). So this is the place where we have the slit and the slit is basically in the point given by this value $Ny/2$ minus 2. So this is the upper point is $Ny/2$ minus 2, the lower point is $Ny/2$ plus 2. So you have a distance of $4 \cdot dy$. So $4 \cdot dy$ is going to be your slit dimension and we will see that while we are simulating.

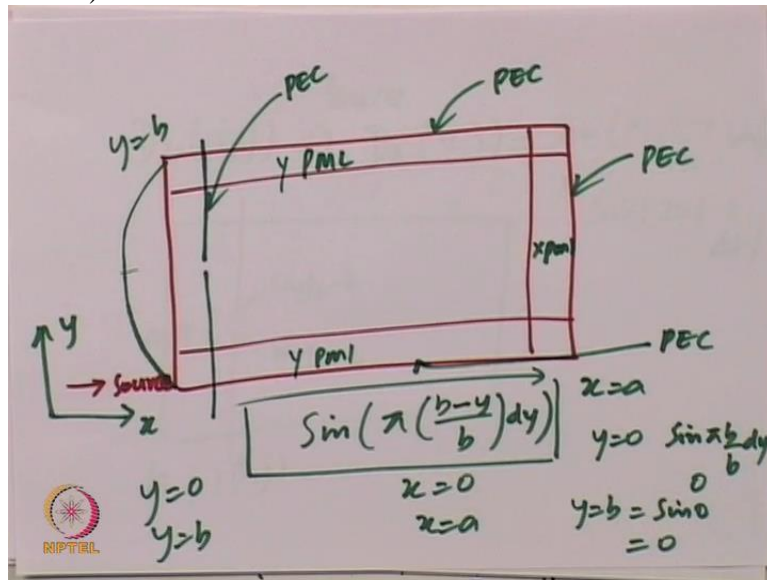
So let us start working on this problem. So we have set all the parameters. So let us start running it.

(Refer Slide Time: 37:59)



And while running it I will zoom it and you will see that the propagation of the wave is coming. So it is just a PEC that we have put here. It is going to reflect all the waves except the points here. And this is the point where there is a slit which is roughly $4 \Delta y$. And you are seeing that the mode is getting leaked through the slit and it is coming out. And whatever is coming out you see that both in y direction and in the x direction it is getting absorbed. So you can see the waves are getting absorbed. We have a pc condition on all the three sides. So this is a source side so the PEC is here, PEC is here. And the PML is going to be here. So the PMLs are mainly there on the all three sides. And the PMLs are truncated by PEC conditions.

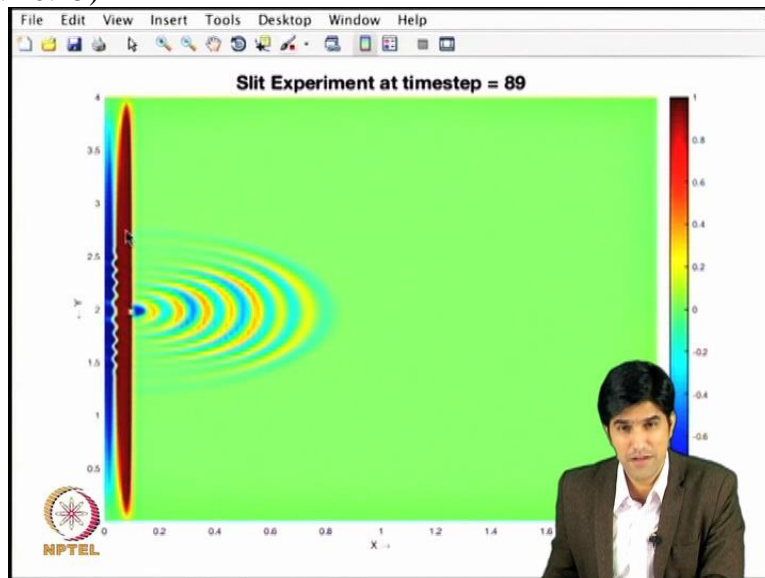
(Refer Slide Time: 39:09)



So let us look back. What we have is so this is going to be PML so this is the y oriented PML this is the y oriented PML, this is the x oriented PML and we have PEC truncating the PML. So this is important for you to know and again what we have done is we have put a small slit here and this is also a PEC. And you can see both the PMLs y oriented PML and x oriented PML is absorbing the waves as expected. And the slit experiment is behaving the way we want.

I want you take this particular example as a test case and code yourself the PML and also the slit experiment.

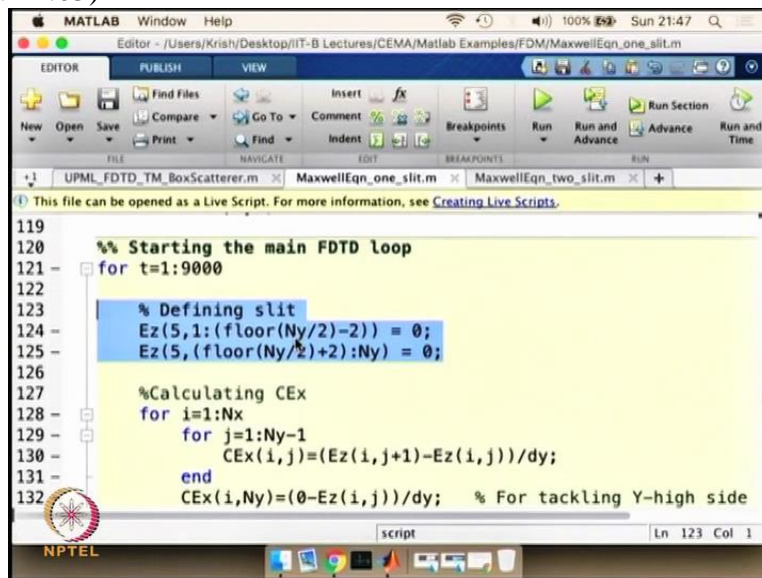
(Refer Slide Time: 40:15)



So we looked at one of the interesting problem of simulating Maxwell equation using one slit experiment. It was very fun experiment to try. I encourage you to try I donot know how much of you have really going to try, but I really encourage you to try that particular experiment. And now we are going to complicate this a little bit by having two slits. It is nothing difficult to code it in the finite difference program what we have.

So I am going to show you step by step what we have done. What you will see is it is exactly the same code except for the point where we have defined the slits.

(Refer Slide Time: 41:03)

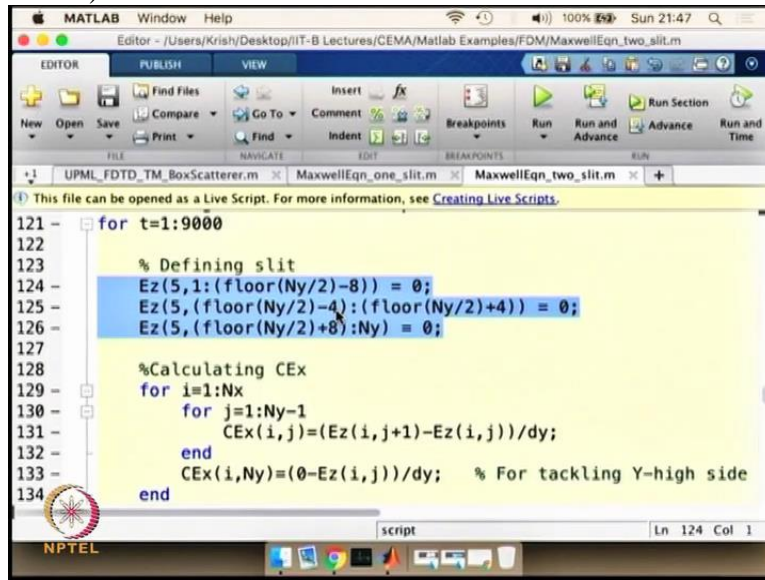


The image shows a MATLAB editor window with the following code:

```
119
120 % Starting the main FDTD loop
121 for t=1:9000
122
123 % Defining slit
124 Ez(5,1:(floor(Ny/2)-2)) = 0;
125 Ez(5,(floor(Ny/2)+2):Ny) = 0;
126
127 %Calculating CEx
128 for i=1:Nx
129     for j=1:Ny-1
130         CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;
131     end
132     CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side
```

So this is the place where we are defining the slits. So if we look at the one slit experiment there is only going to be one slit.

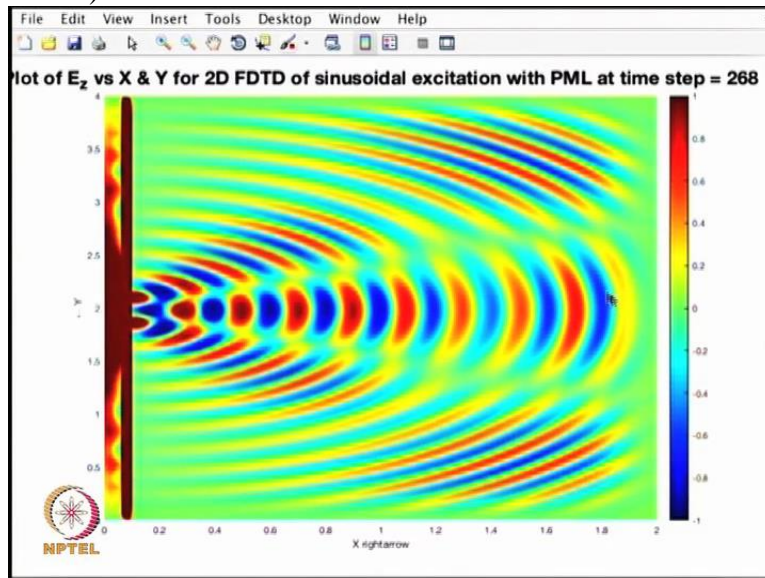
(Refer Slide Time: 41:07)



```
121 - for t=1:9000
122
123     % Defining slit
124 -   Ez(5,1:(floor(Ny/2)-8)) = 0;
125 -   Ez(5,(floor(Ny/2)-4):(floor(Ny/2)+4)) = 0;
126 -   Ez(5,(floor(Ny/2)+8):Ny) = 0;
127
128     %Calculating CEx
129     for i=1:Nx
130         for j=1:Ny-1
131             CEx(i,j)=(Ez(i,j+1)-Ez(i,j))/dy;
132         end
133         CEx(i,Ny)=(0-Ez(i,j))/dy; % For tackling Y-high side
134     end
```

Whereas in the two slit experiments we are going to have 2 slits and whose dimensions are given here. It is going to run from Ny by 2 minus 8 to Ny by 2 minus 4. And then again Ny by 2 plus 4 to Ny by 2 plus 8. So there are going to be two slits of this dimension. And they are going to be in the same location.

(Refer Slide Time: 41:32)



So let us run this one. What we are going to see is the mode is going to come in and you see that there are going to be two slits that are leading to two waves. They act like a point source here and they are going to have constructive and destructive differences in the wave front. And what you see is a kind of a line straight line basically joining the center of two slits and they are going

to radiate out. And you also see the maximum and minimums on the wavefronts. So these are the minimums where you have the destructive wavefronts and the maximums are the wave you have the constructive wavefronts. And the wave is propagating nicely and you see that the PML is absorbing whatever is coming in.

This is a very beautiful way to test two things. One is the understanding of your Maxwell equation model in finite difference method and also to test various boundary conditions. In this case we are using two boundary conditions one is a simple boundary condition which is the PEC boundary condition. The second boundary condition is the source boundary condition which we are going to give as a hard source. And then there is going to be a boundary layer which is a PML. We have not discussed it here; we will describe this later on.

So I encourage you to test such problems to practice various applications of Maxwell equations. And once you know how to model that kind of problem you can test various examples, for example you can model a horn antenna. You can feed them using a particular mode and then see the radiation pattern so on and so forth.

So with this we come to the end of this particular module. I have shown you a practical way to simulate Maxwell equation using finite difference method. We have taken a simple example of slit experiment and we have slightly complicated it using two slits in order to see the maxima and minima the constructive and destructive interference so on and so forth. And we also see that how the wave is propagating and how the boundaries are behaving.

So this is an excellent example for you to try out some of your programming skills and also to model Maxwell equation related problems in a 2D finite difference method.

So please use such codes to learn and model simple problems so that when we go into advanced method like finite element method, finite volume method, method of moments. You will be able to appreciate various aspects and various dimensions of modeling much more easily. So finite difference method gives you that foundation so with that I come to the end of this particular module.

Thank you!