# FDTD(3 次元導波管解析) プログラムの説明

v1.1 Apr.2014

## 目次

# １．プログラム名と対応ファイル名との関係

便宜上，プログラムを次のように定義する．

*番号*．*プログラム名* ……*（ファイル名）*

1. module fdtd_lib ……(fdtd_lib.f90)

2. program main ……(main.f90)

3. subroutine lattice_time ……(lattice_time.f90)

4. subroutine media_coeff ……(media_coeff.f90)

5. subroutine modeling ……(modeling.f90)
    5.1 subroutine rectangular_hollow_xy
    5.2 subroutine rectangular_hole_xy
    5.3 subroutine rectangular_media
    5.4 subroutine rectangular_media_2

6. subroutine init_eh_field ……(eh_field.f90)
6. subroutine e_field
6. subroutine h_field

7. subroutine j_source ……(source.f90)

8. subroutine waveguide ……(waveguide.f90)

9. subroutine power ……(power.f90)
    9.1 subroutine poynting_xy
    9.2 subroutine poynting_yz
    9.3 subroutine poynting_zx

10. subroutine init_e_mur1 ……(e_mur1.f90)
10. subroutine e_mur1
    10.1 subroutine e_mur1_xy1
    10.2 subroutine e_mur1_xy2
    10.3 subroutine e_mur1_yz1
    10.4 subroutine e_mur1_yz2
    10.5 subroutine e_mur1_zx1
    10.6 subroutine e_mur1_zx2
    10.7 subroutine e_mur1_edge

11. subroutine output ……(output.f90)
    11.1 subroutine slice_zx
    11.2 subroutine slice_xy
    11.3 subroutine slice_yz

# ２．各プログラムの説明

## 1. module fdtd_lib 〈モジュール関数〉

ライブラリと同じ役割を持ち，以降すべてのプログラムで共通して参照するファイルである．物理定数（真空の誘電率・透磁率，導電率，光速など）の設定と，FDTD で使うすべての変数を集中宣言する．FDTD で使う配列には，実行時割り当て（解析空間の大きさを決定したら，自動的にその分の配列が確保される方法）がを用いて宣言する．

以下は，モジュール関数の内部を一部抜粋したものである．

```
module fdtd_lib
!=============================================================================
   !******** physical constant ********                                    物理定数の設定
   real(8),parameter :: mu0=12.5663706d-7     ! permiability in vacume
   real(8),parameter :: eps0=8.8541878d-12    ! permittivity in vacume
   real(8),parameter :: sig8=5.7000000d8      ! ideal conductivity
   real(8),parameter :: vc=2.9979000000d8     ! light velocity
   real(8),parameter :: pi=3.1415927d0        ! circular constant
   !******** frequency condition ********                                  周波数に関する変数の宣言
   real(8) :: fmin,fmax,dfreq                 ! For frequency loop
   integer :: icyc                            ! 1 cycle iteration step
   real(8) :: freq,omega                      ! frequency
   !******** time condition ********                                       時間に関する変数の宣言
   integer :: it                              ! total iteration step
   real(8) :: dt,time,tt                      ! time
   real(8) :: courant                         ! courant condition
   !******** space condition ********                                      配列の最大値を示す変数の宣言
   integer :: ix,iy,iz                        ! max lattice number
   integer :: mx,my,mz                        ! dimension
   integer :: nx,ny,nz                        ! total cell
   real(8) :: dl,dx,dy,dz                     ! cell size
   real(8) :: lx,ly,lz                        ! total length [m]

   !******** boundary reference ********                                   実行時割り当て形式で，解析領域
   integer :: nxpoint,nypoint,nzpoint           ! number of reference point   の大きさを決める変数を宣言
   integer,dimension(:),allocatable :: xi,yi,zi ! number from origin         （ここでは大きさは確定していない）
   real(8),dimension(:),allocatable :: x,y,z    ! distance from origin [m]
   !******** field ********                                                実行時割り当て形式で，解析領域
   real(8),dimension(:,:,:),allocatable :: ex,ey,ez ! e_field                の電磁界の配列を宣言
   real(8),dimension(:,:,:),allocatable :: hx,hy,hz ! h_field                （ここでは大きさは確定していない）


                    （その他省略）

end module fdtd_lib
```

## 2. program main　〈メイン関数〉

プログラムの流れ全体を管理するプログラムである．各サブルーチン（初期条件，配列の確保，媒質の設定，解析構造の入力，電界差分式，電流励振，磁界差分式，電磁界分布の出力，ポインティング電力）を，処理する順番ごとにここで呼びだす．

以下はメイン関数を抜粋したものの説明である．

```
program main
   use fdtd_lib                                  1. module fdtd_lib の読み込み
   implicit none                                 暗黙の型宣言（実数，整数）は使用しない．
   integer :: n ! time step

      !******** initial condition ********        初期設定
      open(1,file='initialdata')                 初期条件を出力するファイル名
      open(2,file='bc_pec')                      PEC 境界条件を出力するファイル名
      open(3,file='bc_media2')                   誘電体構造を出力するファイル名
      call lattice_time                          3. subroutine lattice_time の読み込み
      call init_eh_field                         6. subroutine init_eh_field の読み込み
      call media_coeff                           4. subroutine media_coeff の読み込み
      call modeling                              5. subroutine modeling の読み込み
      call waveguide                             8. subroutine waveguide の読み込み
      call init_e_mur1                           10. subroutine init_e_mur1 の読み込み
!      call init_modalpower                      不使用
      close(1)                                   ファイルクローズ
      close(2)                                   ファイルクローズ
      close(3)                                   ファイルクローズ

      time=0.0d0  ! n=0                          計算開始
      do n=1,it
         write(6,'(1x,d18.9,i6,d18.9)') freq,n,time   周波数，タイムステップ，時刻の出力
         !******** e field ********
         call e_field                            6. subroutine e_field の読み込み
         call e_mur1                             10. subroutine e_mur1 の読み込み
         call j_source                           7. subroutine j_source の読み込み
         time=time+dt/2.0d0                      半ステップ時間を進める
         !******** h field ********
         call h_field                            6. subroutine h_field の読み込み
         time=time+dt/2.0d0                      半ステップ時間を進める
         !******** output ********
         call output                             11. subroutine output の読み込み
         call power                              9. subroutine power の読み込み
!         call modalpower                        不使用
      end do                                     時間ループ繰返し

end program main
```

## 3. subroutine lattice_time　〈サブルーチン関数〉

セル（ラティス）サイズ，タイムステップ，周波数，および，解析領域の大きさなどの初期条件を入力するプログラム．タイムステップはクーラントの安定条件を満たす値を設定する．この安定条件はセルサイズによってのみ決定される値である．

以下は，それぞれセルサイズ，タイムステップ，周波数，解析領域の大きさを決める部分の説明．

クーラントの安定条件について．

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

クーラントの安定条件式(セルサイズのみで決まる)

```
!******** lattice widths ********
dl=0.5d-3
dx=dl
dy=dl
dz=dl
!******** time step and courant condition ********
courant=1/vc/sqrt(1/dx**2 + 1/dy**2 + 1/dz**2)
dt=0.950d-12  ! < 0.9629244958d-12 (for dl=0.50mm)
!******** sinusoidal frequency and time condition ********
freq=10.0d9
omega=2.0d0*pi*freq  ! rad/sec
icyc=nint(1/freq/dt)
```

$\Delta l$　セルサイズを 0.5 mm 均一に設定.
$\Delta x$　（通常λ/20 以下は必要と言われている）
$\Delta y$　この例では dx=dy=dz=0.5 mmの立方体メッシュ
$\Delta z$
安定条件を満たすタイムステップの設定.
クーラントの安定条件
$\Delta t$
正弦波周波数の設定.
$f$
$\omega = 2\pi f$
$T/\Delta t$　正弦波 1 周期あたりのタイムステップ数

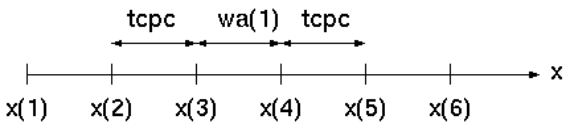
```
!******** define reference point x ********
! reference point location
x(1)=0.0d0
x(2)=x(1)+0.0d-3
x(3)=x(2)+tcpc
x(4)=x(3)+wa(1)
x(5)=x(4)+tcpc
x(6)=x(5)+0.0d-3
lx  =x(6)
! lattice numbering
do i=1,nxpoint
   xi(i)=1+nint(x(i)/dx)
end do
ix=xi(nxpoint)
! total lattice
nx=ix-1
```

境界条件（誘電体や導体）を入れるために便宜上
設けた x 方向の参照番号．y，z も同じ．
ここで入力した構造から解析領域の大きさが決まり，
確保すべき配列の大きさが決定される．
tcpc は金属厚み，wa(1)は導波管の横幅

入力された解析領域の大きさ（長さ）をもとに
整数の参照番号を割り当てる．

ここで，x 方向の最大配列番号が決定される．

ここで，x 方向のセル数が決定される．




x 方向の参照番号の入力概念図

## 4. subroutine media_coeff　〈サブルーチン関数〉

任意媒質の材料定数を入力する.そして媒質ごとに識別 ID を与え,媒質定数を含んだ差分係数を定義するプログラムである.電界・磁界の差分式では，異なる媒質ごと識別 ID を与え，それぞれの媒質で各 ID に対応する計算係数を読み替えて計算を行っている.

差分係数について.

差分係数については次のような置き換えを行っている.（例　Ex を求める差分式の場合）

$$\frac{\partial E_x}{\partial t}=\frac{1}{\varepsilon}\left(\frac{\partial H_z}{\partial y}-\frac{\partial H_y}{\partial z}-\sigma E_x\right)$$　　マクスウェルの方程式そのもの

時刻 $n-1/2$ を中心に時間微分すると，

$$E_x^n=\frac{1-\dfrac{\sigma t}{2\varepsilon}}{1+\dfrac{\sigma t}{2\varepsilon}}E_x^{n-1}+\frac{\dfrac{\Delta t}{\varepsilon}}{1+\dfrac{\sigma t}{2\varepsilon}}\left(\frac{\partial H_z^{n-\frac{1}{2}}}{\partial y}-\frac{\partial H_y^{n-\frac{1}{2}}}{\partial z}\right)$$　　時間微分の結果

空間 $(i+1/2,j,k)$ を中心に空間微分すると，

$$\begin{aligned}E_x^n\left(i+\tfrac{1}{2},j,k\right)=&\,\mathrm{cex}\cdot E_x^{n-1}\left(i+\tfrac{1}{2},j,k\right)\\&+\mathrm{cexly}\cdot\left\{H_z^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j+\tfrac{1}{2},k\right)-H_z^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j-\tfrac{1}{2},k\right)\right\}\\&-\mathrm{cexly}\cdot\left\{H_y^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j,k+\tfrac{1}{2}\right)-H_y^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j,k-\tfrac{1}{2}\right)\right\}\end{aligned}$$　　時間＆空間微分の結果

ただし，置き換えた係数は次の通りである.

$$\mathrm{cex}=\frac{1-\dfrac{\sigma\left(i+\frac{1}{2},j,k\right)\Delta t}{2\varepsilon\left(i+\frac{1}{2},j,k\right)}}{1+\dfrac{\sigma\left(i+\frac{1}{2},j,k\right)\Delta t}{2\varepsilon\left(i+\frac{1}{2},j,k\right)}}\qquad \mathrm{cexly}=\frac{\dfrac{\Delta t}{\varepsilon\left(i+\frac{1}{2},j,k\right)}}{1+\dfrac{\sigma\left(i+\frac{1}{2},j,k\right)\Delta t}{2\varepsilon\left(i+\frac{1}{2},j,k\right)}}\cdot\frac{1}{\Delta y}\qquad \mathrm{cexly}=\frac{\dfrac{\Delta t}{\varepsilon\left(i+\frac{1}{2},j,k\right)}}{1+\dfrac{\sigma\left(i+\frac{1}{2},j,k\right)\Delta t}{2\varepsilon\left(i+\frac{1}{2},j,k\right)}}\cdot\frac{1}{\Delta z}$$

次の例は ID として，0 番を真空，1 番を PEC（電気壁）または PMC（磁気壁），2 番を誘電体に割り当てている.

```
!******** number of dielectric media ********
! 0 : air
! 1 : pec & pmc
! >2: arbitrary media
nmedia=2

! id=0 is the vacume
eps(0)=eps0
sig(0)=0.0d0
mu(0)=mu0

! id=1 is pec,pmc (no define, see <e-field> or <h-field> )

! id=2 is a dielectric media
! this media constant is 'er = 2-j0' and 'mr=1-j0'
eps(2)=eps0*(5.0d0)
sig(2)=omega*eps0*(0.0d0)
mu(2)=mu0*(0.0d0)
```

ID=0 は真空を表す.
ID=1 は電気壁または，磁気壁を表す.
ID≧2 は任意の媒質（誘電体・磁性体）を表す.
この例では ID=2 まで使うことを示す.

ID=0 に真空の誘電率，導電率，透磁率を与える.
$\varepsilon_0$
$\sigma_0$
$\mu_0$

ID=1 の場合のみ，E=0(PEC のとき)または，
H=0(PMC のとき)が明らかなので，係数を定義せず
差分式内部で別処理する.

ID=2 に誘電率(εr=5)，導電率，透磁率を与える.

$\varepsilon_2$
$\sigma_2$　誘電率の虚部はこの中に含まれる.
$\mu_2$　$\varepsilon=\varepsilon'-j\varepsilon''$ のとき，
等価導電率は $\sigma=\omega\varepsilon''$ である.

## 5. subroutine modeling　〈サブルーチン関数〉

3. subroutine lattice_time で設定した便宜上の参照番号をもとに，導体，誘電体などの具体的な形状を入力するプログラム．円筒形状や，楕円形状，球，三角形などを入力するサブルーチンを予め用意しておけば様々な形状に応用できる．

下の例は最も基本的な直方体を入力する場合である．

```
   ! waveguide 1                          導波管壁面の直方体金属を入力する
   ! yz left
   mxs=xi(2)                              媒質の x 方向の始点を指定
   mxe=xi(3)                              媒質の x 方向の終点を指定
   mys=yi(2)                              媒質の y 方向の始点を指定
   mye=yi(5)                              媒質の y 方向の終点を指定
   mzs=zi(1)                              媒質の z 方向の始点を指定
   mze=zi(2)                              媒質の z 方向の終点を指定
   call rectangular_media                 指定された座標からなる矩形媒質の ID を入力するサブルーチンを呼ぶ．

subroutine rectangular_media              指定された座標に従って，媒質 ID を入力するサブルーチン．
   use fdtd_lib                           この例では，矩形の金属（ID=0）を入力している．
   implicit none
   integer :: i,j,k !

   do i = mxs,mxe                         電界の z 成分 ez の ID 入力
      do j = mys,mye
         do k = mzs,mze-1
            id_ez(i,j,k)=1
         end do
      end do
   end do
   do i = mxs,mxe                         電界の y 成分 ez の ID 入力
      do j = mys,mye-1
         do k = mzs,mze
            id_ey(i,j,k)=1
         end do
      end do
   end do
   do i = mxs,mxe-1                       電界の x 成分 ez の ID 入力
      do j = mys,mye
         do k = mzs,mze
            id_ex(i,j,k)=1
         end do
      end do
   end do

end subroutine rectangular_media
```

## 6. subroutine init_eh_field　〈サブルーチン関数〉

電界と磁界の配列を確保するプログラム.

## 6. subroutine e_field　〈サブルーチン関数〉

電界差分式を計算するプログラム.
以下の例は電界の x 成分 Ex を計算する部分の抜粋である.

```
subroutine e_field
!*****************************************************************************
! electric field calculation by fd method
!*****************************************************************************
   use fdtd_lib
   implicit none
   integer :: i,j,k,id

   !******** ex ********
   do k=2,iz-1                                          配列 2 番目から最大配列 iz の 1 つ前まで
      do j=2,iy-1                                       配列 2 番目から最大配列 iy の 1 つ前まで
         do i=1,ix-1                                    配列 1 番目から最大配列 ix の 1 つ前まで
            id=id_ex(i,j,k)

            if(id.eq.0) then                            ID=0（真空）ならば, 真空の差分係数を呼出す
            ! 0: vacume
               ex(i,j,k)=cex0*ex(i,j,k) &
                        +cexry0*(hz(i,j,k)-hz(i,j-1,k)) &
                        -cexrz0*(hy(i,j,k)-hy(i,j,k-1))
            else if(id.eq.1) then                       ID=0（PEC）ならば, 電界は常に零
            ! 1: ideal conductor
               ex(i,j,k)=0.0d0
            else
            ! 2: arbitrary                              ID=0（誘電体）ならば, 誘電体の差分係数を
               ex(i,j,k)=cex(id)*ex(i,j,k) &            呼出す.
                        +cexry(id)*(hz(i,j,k)-hz(i,j-1,k)) &
                        -cexrz(id)*(hy(i,j,k)-hy(i,j,k-1))
            end if
         end do
      end do
   end do
```

電界差分式について.

$$E_x^n\left(i+\tfrac{1}{2},j,k\right)=\mathrm{cex}\cdot E_x^{n-1}\left(i+\tfrac{1}{2},j,k\right)$$
$$+\mathrm{cexly}\cdot\left\{H_z^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j+\tfrac{1}{2},k\right)-H_z^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j-\tfrac{1}{2},k\right)\right\}$$
$$-\mathrm{cexly}\cdot\left\{H_y^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j,k+\tfrac{1}{2}\right)-H_y^{n-\frac{1}{2}}\left(i+\tfrac{1}{2},j,k-\tfrac{1}{2}\right)\right\}$$

## 6. subroutine h_field　〈サブルーチン関数〉

サブルーチン関数. 磁界差分式を計算するプログラム.
省略

# 7. subroutine j_source　〈サブルーチン関数〉

シート状の面電流を流して励振源とするプログラム．ほかの励振方法として，電界のみ，磁界のみとした場合でも解析はできるが，物理的には電流を入力するのが最良と考えられている．

等価面電流について．

$$\begin{cases} E_y^{\pm} = -Z_{TE}\left(\dfrac{m\pi}{a}\right)A_{m0}^{\pm}\sin\dfrac{n\pi x}{a}e^{\mp\gamma z} \\ H_y^{\pm} = \pm\left(\dfrac{m\pi}{a}\right)A_{m0}^{\pm}\sin\dfrac{n\pi x}{a}e^{\mp\gamma z} \end{cases}$$

TEm0 モードの横電磁界

これを波源の境界条件に代入し，導波管伝送電力が 1 W となるように振幅 A を決めると，電流密度は次式となる．

$$J_y = \frac{2}{\Delta z}\sqrt{\frac{2}{abZ_{TE}}}\sin\left(\frac{m\pi x}{a}\right)e^{j\omega t} \qquad [\mathrm{A/m^2}]$$

なお，電流密度を含む差分式は，マクスウェルの方程式より次式で与えられる．下線部分は，もとの電界差分式そのものである．よって電界差分式の計算の直後に，上式の電流を入力するだけでよい．

$$E^n = \frac{1-\dfrac{\sigma\Delta t}{2\varepsilon}}{1+\dfrac{\sigma\Delta t}{2\varepsilon}}E^{n-1} + \underline{\frac{\dfrac{\Delta t}{\varepsilon}}{1+\dfrac{\sigma\Delta t}{2\varepsilon}}\left(\nabla\times H^{n-\frac{1}{2}}\right) - \frac{\dfrac{\Delta t}{\varepsilon}}{1+\dfrac{\sigma\Delta t}{2\varepsilon}}J^{n-\frac{1}{2}}}$$

以下の例は，方形導波管 TEm0 モードの等価面電流を流すプログラムの抜粋である．

```
subroutine j_source
!*****************************************************************************
!  electric current field source for rectangular TEm0 mode
!*****************************************************************************
   use fdtd_lib
   implicit none
   integer :: i,j,k,id !

   do i=xi(3),xi(4)                                          導波管の内寸
      do j=yi(3),yi(4)-1                                        〃
            k=zi(1)+2                                           〃
            id=id_ey(i,j,k)
               ! waveguide inner edge                        導波管の内壁エッジは強制的に零
               if (i==xi(3) .or. i==xi(4)) then
                  ey(i,j,k)=0.0d0
               ! inside waveguide
               else
                  ey(i,j,k)=ey(i,j,k) &
                            -(dt/eps(id))/(1+(sig(id)*dt/(2.0d0*eps(id)))) &
!                            *(-2.0d0)/real(zet(dmode))/dz &
                         *(2.0d0/dz)*sqrt(2.0d0/(wx*wy*real(zet(dmode)))) &
                            *dsin(dmode*pi*(i-xi(3))/(xi(4)-xi(3))) &
                            *dsin(2.0d0*pi*freq*(time-dt/2.0d0))
               end if
      end do
   end do

end subroutine j_source
```

## 8. subroutine waveguide　〈サブルーチン関数〉

導波管の位相速度や減衰定数，伝搬定数などの導波管パラメータを理論的に計算するプログラム．特に必要がある部分ではないが，吸収境界条件に MUR を使用する場合には，導波管の位相速度が必要になるため，ここで定義している．

以下は，同プログラムの一部を抜粋したものである．

TE 導波管パラメータについて．

$k_0 \geq k_c$ のとき（伝搬モード），

$$\alpha = 0$$
$$\beta = \sqrt{k_0^{\,2} - k_c^{\,2}}$$
$$v = \frac{\omega}{\beta}$$
$$Z_{TE} = \frac{\omega\mu_0}{\beta}$$

$k_0 \leq k_c$ のとき（非伝搬モード），

$$\alpha = \sqrt{k_c^{\,2} - k_0^{\,2}}$$
$$\beta = 0$$
$$v = 0$$
$$Z_{TE} = j\frac{\omega\mu_0}{\alpha}$$

```
subroutine waveguide
!**********************************************************************
!  rectangular waveguide parameters for TE(H) wave
!**********************************************************************
   use fdtd_lib
   implicit none
   integer :: m  ! mode number

   do m=0,nmode
      ! wavenumber
      k0=omega*sqrt(mu0*eps0)                          自由空間の波数
      kc(m)=real(m)*pi/wx                              カットオフ波数
      if (k0 >= kc(m)) then                            伝搬モードの条件が成立するとき，
      ! for propagation mode
         alpha(m)=0.0d0                                減衰定数
         beta(m)=sqrt(k0**2-kc(m)**2)                  位相定数
         gamma(m)=alpha(m)+complex_j*beta(m)           伝搬定数
         vp(m)=omega/beta(m)                           位相速度
         zet(m)=omega*mu0/beta(m)                      導波管インピーダンス
      else
      ! for evanescent mode                            非伝搬モードの条件が成立するとき，
         alpha(m)=sqrt(kc(m)**2-k0**2)
         beta(m)=0.0d0
         gamma(m)=alpha(m)+complex_j*beta(m)
!          vp(m)=omega/(-complex_j*alpha(m))
         vp(m)=0.0d0
         zet(m)=complex_j*omega*mu0/alpha(m)
      end if
   end do
```

## 9. subroutine power　〈サブルーチン関数〉

任意の平面を通過するポインティング電力を計算するプログラム．z方向のポインティング電力を計算するときは，Sz = ex*hy - ey*hx に従って計算している．

ポインティング電力の計算について．

セルの中央 $\left(i+\frac{1}{2}, j, k+\frac{1}{2}\right)$ における瞬時電力の y 成分は，次のようになる．

$$P_y\left(i+\tfrac{1}{2}, j, k+\tfrac{1}{2}\right) = S_y\left(i+\tfrac{1}{2}, j, k+\tfrac{1}{2}\right)\cdot \Delta x \Delta z$$

ただし，$S_y = E_z H_x - E_x H_z$

電磁界もセルの中央の値を求めるようにすると，次式のように電界は中央を挟む 2 点の平均，磁界は 4 点の平均となる．

$$E_z = \tfrac{1}{2}\left\{E_z\left(i, j, k+\tfrac{1}{2}\right) + E_z\left(i+1, j, k+\tfrac{1}{2}\right)\right\}$$

$$H_x = \tfrac{1}{4}\left\{H_x\left(i, j-\tfrac{1}{2}, k+\tfrac{1}{2}\right) + H_x\left(i+1, j-\tfrac{1}{2}, k+\tfrac{1}{2}\right) + H_x\left(i+1, j+\tfrac{1}{2}, k+\tfrac{1}{2}\right) + H_x\left(i, j+\tfrac{1}{2}, k+\tfrac{1}{2}\right)\right\}$$

$$E_x = \tfrac{1}{2}\left\{E_x\left(i+\tfrac{1}{2}, j, k\right) + E_x\left(i+\tfrac{1}{2}, j, k+1\right)\right\}$$

$$H_z = \tfrac{1}{4}\left\{H_z\left(i+\tfrac{1}{2}, j-\tfrac{1}{2}, k\right) + H_z\left(i+\tfrac{1}{2}, j-\tfrac{1}{2}, k+1\right) + H_z\left(i+\tfrac{1}{2}, j+\tfrac{1}{2}, k+1\right) + H_z\left(i+\tfrac{1}{2}, j+\tfrac{1}{2}, k\right)\right\}$$

```
subroutine poynting_xy
!-------------------------------------------------------------------------------
!   subroutine poynting_xy
!-------------------------------------------------------------------------------
   use fdtd_lib
   implicit none
   integer :: i,j,k

   !******** xy (sz = ex.hy - ey.hx) ********                    ポインティングベクトルの z 成分の式
   psum_xy = 0.0d0
   do i=pxs,pxe-1
      do j=pys,pye-1
         k=pzp
         ! ex
         exc=(ex(i,j+1,k)+ex(i,j,k))/2.0d0
         ! hy
         hyc=(hy(i,j,k-1)+hy(i,j+1,k-1)+hy(i,j+1,k)+hy(i,j,k))/4.0d0
         ! ey
         eyc=(ey(i,j,k)+ey(i+1,j,k))/2.0d0
         ! hx
         hxc=(hx(i,j,k-1)+hx(i+1,j,k-1)+hx(i+1,j,k)+hx(i,j,k))/4.0d0
         ! p_xy
         psum_xy = psum_xy + (exc*hyc-eyc*hxc)*dx*dy  ! [W]
      end do
   end do

   write(7,'(1x,2d18.10)') time,psum_xy

end subroutine poynting_xy
```

## 10. subroutine init_e_mur1　〈サブルーチン関数〉

MUR の一次吸収境界条件で必要になる配列を確保するプログラム.

## 10. subroutine e_mur1　〈サブルーチン関数〉

MUR の一次吸収境界条件（One way equation）に従って，解析領域端面の電界を計算するプログラム．ｘｙ面，ｙｚ面，ｚｘ面の合計 6 面に加えて，8 本のエッジ線部分を計算するサブルーチンも含まれる.

One way equation について.

$E$ を電界成分とし，ある時間 $t = n\Delta t$ と座標 $x = i\Delta x, y = j\Delta y, z = k\Delta z$ における電界を次式のように記号で定義すると，

$$E\left(t = n\Delta t, x = i\Delta x, y = j\Delta y, z = k\Delta z\right) \equiv E_{ijk}^{n}$$

z 方向に伝搬する際の MUR の一次近似吸収境界条件は次のようになる.

$$E_{ij0}^{n} = E_{ij1}^{n-1} - \frac{\Delta z - v\Delta t}{\Delta z + v\Delta t}\left(E_{ij1}^{n} - E_{ij0}^{n-1}\right)$$

```
subroutine e_mur1_xy1
!-------------------------------------------------------------------------
!   subroutine mur1_free_xy for xy free space
!-------------------------------------------------------------------------
   use fdtd_lib
   implicit none
   integer :: i,j !

   ptz=(vp(mur_mode)*dt-dz)/(vp(mur_mode)*dt+dz)

   do i=1,ix-1
      do j=2,iy-1
         ex(i,j,1)=ex1_xy(i,j,2)+ptz*(ex(i,j,2)-ex1_xy(i,j,1))
      enddo
   enddo
   do j=1,iy-1
      do i=2,ix-1
         ey(i,j,1)=ey1_xy(i,j,2)+ptz*(ey(i,j,2)-ey1_xy(i,j,1))
      enddo
   enddo
   !******** update past field ********                    時間メモリを節約するために 1 ステップ前の電界を
   do i=1,ix-1                                            ここで保存しておく．あとは順次上書きされる.
      do j=2,iy-1
         ex1_xy(i,j,2)=ex(i,j,2)
         ex1_xy(i,j,1)=ex(i,j,1)
      enddo
   enddo
   do j=1,iy-1
      do i=2,ix-1
         ey1_xy(i,j,2)=ey(i,j,2)
         ey1_xy(i,j,1)=ey(i,j,1)
      enddo
   enddo

end subroutine e_mur1_xy1
```

## 11. subroutine output　〈サブルーチン関数〉

計算結果を一定の時間ごとに出力ファイルに書き出して，電磁界の伝搬を可視化するためのプログラム．任意の二次元平面を観測面とし，その面上にある電磁界を観察する．xy スライス，yz スライス，zx スライスの 3 つのサブルーチンを含む．

以下は zx スライスのサブルーチンの一部を抜粋したものである．

```
   !******** field images per cycle ********
   animation=5
```

正弦波 1 周期あたりに何枚の静止画像を用意するか設定する．

```
subroutine slice_zx
!-------------------------------------------------------------------------
!  slice_zx plane
!-------------------------------------------------------------------------
   use fdtd_lib
   implicit none
   integer :: i,j,k !

   !******** time step counter ********
   tsc3=tsc3+1
   !******** file for zx gnuplot ********
   if (tsc3==icyc/animation) then
     fc3=fc3+1
     tsc3=0   !reset time step counter
       if (fc3<10) then
           write(field_zx,fmt='(''field_zx000'',i1)') fc3
       else if (fc3<100) then
           write(field_zx,fmt='(''field_zx00'',i2)') fc3
       else if (fc3<1000) then
           write(field_zx,fmt='(''field_zx0'',i3)') fc3
       else
           write(field_zx,fmt='(''field_zx'',i4)') fc3
       endif

       open(unit=3,file=field_zx)
       do k=1,iz
          do i=1,ix
               j=slice_y
                write(3,'(1x,3i4,2e18.10)') i,k,j, &
                             ey(i,j,k),-real(zet(dmode))*hx(i,j,k)
          end do
          write(3,*)
       end do
       close(3)
   endif
end subroutine slice_zx
```

例えば icyc=100（1 周期あたりのタイムステップ数），animation=5 のときならば，tsc3 という変数が 20 になるたびにファイル出力がされる．

# 参考文献

FDTD 法に関する国内参考書
[1] 宇野亨，FDTD 法による電磁界およびアンテナ解析，コロナ社，1998
[2] 橋本修，阿部琢己，FDTD 時間領域差分法入門，森北出版，1996

FDTD ソースコードの入手
[3] 東工大平野先生のホームページ　http://www-antenna.ee.titech.ac.jp/~hira/study/fdtd/fdtd.html

Fortran90 の解説
[1] ラリー ニーホフ，サンフォード リーストマー,入門 Fortran90，ピアソンエデュケーション

国産 FDTD シミュレータの紹介
[4] MAGNA-TDM http://www.engineering-eye.com/MAGNA_TDM/