Define NV axes

In[ ]:=
```
(*define rodrigues rotation matrix for
 arbitary rotation of angle θ around an axis vec*)
Rrod[vec_, θ_] := Module[{ux, uy, uz}, {{ux, uy, uz} = {vec[[1]], vec[[2]], vec[[3]]};
    R =
     (   Cos[θ] + ux^2 (1 - Cos[θ])     ux uy (1 - Cos[θ]) - uz Sin[θ]  uz ux (1 - Cos[θ]) + uy Sir
      uy ux (1 - Cos[θ]) + uz Sin[θ]     Cos[θ] + uy^2 (1 - Cos[θ])     uy uz (1 - Cos[θ]) - ux Sin
      uz ux (1 - Cos[θ]) - uy Sin[θ] uz uy (1 - Cos[θ]) + ux Sin[θ]     Cos[θ] + uz^2 (1 - Cos[θ]
     )
   }];

Rx[θ_] = Rrod[{1, 0, 0}, θ][[1]];(*rotation operator about x*)
Ry[θ_] = Rrod[{0, 1, 0}, θ][[1]];(*rotation operator about y*)
Rz[θ_] = Rrod[{0, 0, 1}, θ][[1]];(*rotation operator about z*)

(*100-NV unit vector*)
NVs100 = 2/Sqrt[3]
   {{1/2, 1/2, 1/2}, {-1/2, -1/2, 1/2}, {1/2, -1/2, -1/2}, {-1/2, 1/2, -1/2}};
(*Rotate NVs to get 111 orientation*)
θm = 1/2 ArcCos[-1/3]; (*define angle for y rotn*)
NVs111 = FullSimplify[Table[Ry[θm].(Rz[-π/4].NVs100[[i]])/Norm[Ry[1/2 θm].(Rz[π/4].NVs100[[i]])], {i, 1, 4}]];
(*Rotate NVs to get 110 orientation*)
NVs110 = Normalize /@ FullSimplify[Table[Ry[-π/4].NVs100[[i]], {i, 1, 4}]];
```

Define Projection operation for each NV axis and Spin matrices

In[ ]:=
```
(*make a coordinate system for the NV*)
MakeCS[nv_] := Module[{v1, v2, v3},
  v1 = nv;
  v2 = nv × {0, 0, 1};
  v3 = (v1 × v2) / (Norm[v1] × Norm[v2]);
  {v1, v2, v3}]
(*projects magnetic field along NV axjs ensuring Bz is along NV axis*)
Proj[B_, nv_] := Module[{v1, v2, v3, vx, vy, vz, Bx, By, Bz},
  {v1, v2, v3} = MakeCS[nv];
  {vx, vy, vz} = {v2, v3, v1};
  {B.vx, B.vy, B.vz}]

Sx = 1/Sqrt[2] {{0, 1, 0}, {1, 0, 1}, {0, 1, 0}}; Sy = 1/Sqrt[2] {{0, -i, 0}, {i, 0, -i}, {0, i, 0}}; Sz = {{1, 0, 0}, {0, 0, 0}, {0, 0, -1}};
Stot = {Sx, Sy, Sz};
```

Create Hyperfine Hamiltonian

In[ ]:=
```
(*now try including nuclear part of hamiltonian*)
Hzps = Dz * KroneckerProduct[Sz^2, IdentityMatrix[3]];
(*Electronic zero field splititng*)
Hqp = P * KroneckerProduct[IdentityMatrix[3], Sz^2]; (*Nuclear quadropole of N14*)
HzeemanE = γ * KroneckerProduct[Bx Sx + By Sy + Bz Sz, IdentityMatrix[3]];
(*electron zeeman splitting*)
HzeemanN = -γn * KroneckerProduct[IdentityMatrix[3], Bx Sx + By Sy + Bz Sz];
(*nuclear zeeman splitting*)
Hhf = A1 *
    (KroneckerProduct[Sx, IdentityMatrix[3]].KroneckerProduct[IdentityMatrix[3], Sx] +
      KroneckerProduct[Sy, IdentityMatrix[3]].
       KroneckerProduct[IdentityMatrix[3], Sy]) + Az *
    KroneckerProduct[Sz, IdentityMatrix[3]].KroneckerProduct[IdentityMatrix[3], Sz];
(*hyperfine interaction term between
 electronic
 and
 nuclear
 spins*)
MHz = 10^6;
(*define key parameters*)
numbers = {Dz → 2 π * 2870 MHz, P → -2 π * 4.9457 MHz, γ → 2 π * 2.803 MHz,
   γn → 2 π * 3.077 * 10^-4 MHz, A1 → -2 π * 2.62 MHz, Az → 2 π * 2.2 MHz};
(*A1→2π*2.14MHz,Az→2π*2.7MHz i dont think these are correct  current values
  from:Fast coherent control of nitrogen-14 spins associated with nitrogen-
    vacancy centers in diamonds using dynamical decoupling
     P value from Robust optical
     readout and characterization of nuclear spin transitions
in nitrogen-vacancy ensembles in diamond*)

(*define full hamiltonian*)
HwithB[B_, nv_] = Module[{Bx, By, Bz, Hfull, HzeemanE, HzeemanN},
   {Bx, By, Bz} = Proj[B, nv];
   HzeemanE = γ * KroneckerProduct[Bx Sx + By Sy + Bz Sz, IdentityMatrix[3]];
   HzeemanN = -γn * KroneckerProduct[IdentityMatrix[3], Bx Sx + By Sy + Bz Sz];
   Hfull = Hzps + Hqp + HzeemanE + HzeemanN + Hhf];

(*Diagonalise hamiltonian based on B field and NV axis (vectors)*)
Energies[B_, nv_] := Module[{Bx, By, Bz, Hfull, Ham, evals},
  Ham = HwithB[B, nv] /. numbers;
  evals = Sort[ 1/(2 π MHz) Eigenvalues[Ham]];
  evals]

lorentzian[x_, dx_, Γ_, amp_] = 1 - amp Γ^2/((x - dx)^2 + Γ^2);
```

```
In[ ]:= GenSpec[B0_, θ_, ϕ_, w_, amp_] := 1/24 Total@
          Map[lorentzian[x, #, w, amp] &, Flatten@Map[{#[[2]] - #[[1]], #[[3]] - #[[1]]} &@Partition[
              Energies[{B0 Sin[θ] Cos[ϕ], B0 Sin[θ] Sin[ϕ], B0 Cos[θ]}, #], 3] &, NVs100]]
```

Generate an ODMR spectrum for B = 100G, $\theta = 25°$, $\phi = 32°$, with lorentzian widths = 2.25MHz and amplitude 0.1

```
In[ ]:= spec[x_] = GenSpec[100, 25 °, 32 °, 2.25, 0.1];
        Plot[spec[x], {x, 2500, 3200}, PlotRange → All, Frame → True]
```

Out[ ]=