```
def Maxwells_Equations_in_Geom_Calculus():
    Print_Function()
    X = symbols('t x y z',real=True)
    (st4d,g0,g1,g2,g3) = Ga.build('gamma*t|x|y|z',g=[1,-1,-1,-1],coords=X)
    I = st4d.i
    B = st4d.mv('B','vector',f=True)
    E = st4d.mv('E','vector',f=True)
    B.set_coef(1,0,0)
    E.set_coef(1,0,0)
    B *= g0
    E *= g0
    J = st4d.mv('J','vector',f=True)
    F = E+I*B
    print r'\text{Pseudo Scalar\;\;}I =',I
    print '\\text{Magnetic Field Bi-Vector\\;\\;} B = \\bm{B\\gamma_{t}} =',B
    print '\\text{Electric Field Bi-Vector\\;\\;} E = \\bm{E\\gamma_{t}} =',E
    print '\\text{Electromagnetic Field Bi-Vector\\;\\;} F = E+IB =',F
    print '%\\text{Four Current Density\\;\\;} J =',J
    gradF = st4d.grad*F
    print '#Geom Derivative of Electomagnetic Field Bi-Vector'
    gradF.Fmt(3,'grad*F')
    print '#Maxwell Equations'
    print 'grad*F = J'
    print '#Div $E$ and Curl $H$ Equations'
    (gradF.get_grade(1)-J).Fmt(3,'%\\grade{\\nabla F}_{1} -J = 0')
    print '#Curl $E$ and Div $B$ equations'
    (gradF.get_grade(3)).Fmt(3,'%\\grade{\\nabla F}_{3} = 0')
    return
```

Code Output:

Pseudo Scalar $I = \gamma_t \wedge \gamma_x \wedge \gamma_y \wedge \gamma_z$

Magnetic Field Bi-Vector $B = \boldsymbol{B\gamma_t} = -B^x \gamma_t \wedge \gamma_x - B^y \gamma_t \wedge \gamma_y - B^z \gamma_t \wedge \gamma_z$

Electric Field Bi-Vector $E = \boldsymbol{E\gamma_t} = -E^x \gamma_t \wedge \gamma_x - E^y \gamma_t \wedge \gamma_y - E^z \gamma_t \wedge \gamma_z$

Electromagnetic Field Bi-Vector $F = E + IB = -E^x \gamma_t \wedge \gamma_x - E^y \gamma_t \wedge \gamma_y - E^z \gamma_t \wedge \gamma_z - B^z \gamma_x \wedge \gamma_y + B^y \gamma_x \wedge \gamma_z - B^x \gamma_y \wedge \gamma_z$

Four Current Density $J = J^t \gamma_t + J^x \gamma_x + J^y \gamma_y + J^z \gamma_z$

Geom Derivative of Electomagnetic Field Bi-Vector

$$\begin{aligned}
\boldsymbol{\nabla} F = & \left(\partial_x E^x + \partial_y E^y + \partial_z E^z\right) \gamma_t \\
& + \left(-\partial_z B^y + \partial_y B^z - \partial_t E^x\right) \gamma_x \\
& + \left(\partial_z B^x - \partial_x B^z - \partial_t E^y\right) \gamma_y \\
& + \left(-\partial_y B^x + \partial_x B^y - \partial_t E^z\right) \gamma_z \\
& + \left(-\partial_t B^z + \partial_y E^x - \partial_x E^y\right) \gamma_t \wedge \gamma_x \wedge \gamma_y \\
& + \left(\partial_t B^y + \partial_z E^x - \partial_x E^z\right) \gamma_t \wedge \gamma_x \wedge \gamma_z \\
& + \left(-\partial_t B^x + \partial_z E^y - \partial_y E^z\right) \gamma_t \wedge \gamma_y \wedge \gamma_z \\
& + \left(\partial_x B^x + \partial_y B^y + \partial_z B^z\right) \gamma_x \wedge \gamma_y \wedge \gamma_z
\end{aligned}$$

Maxwell Equations

$$\boldsymbol{\nabla} F = J$$

Div E and Curl H Equations

$$\begin{aligned}
\langle \nabla F \rangle_1 - J = 0 = & \left(-J^t + \partial_x E^x + \partial_y E^y + \partial_z E^z\right) \gamma_t \\
& + \left(-J^x - \partial_z B^y + \partial_y B^z - \partial_t E^x\right) \gamma_x \\
& + \left(-J^y + \partial_z B^x - \partial_x B^z - \partial_t E^y\right) \gamma_y \\
& + \left(-J^z - \partial_y B^x + \partial_x B^y - \partial_t E^z\right) \gamma_z
\end{aligned}$$

Curl E and Div B equations

$$
\begin{aligned}
\left\langle \nabla F\right\rangle_{3} = 0 = & \left (- \partial_{t} B^{z} + \partial_{y} E^{x} - \partial_{x} E^{y}\right) \gamma_{t}\wedge \gamma_{x}\wedge \gamma_{y} \\
& + \left (\partial_{t} B^{y} + \partial_{z} E^{x} - \partial_{x} E^{z}\right) \gamma_{t}\wedge \gamma_{x}\wedge \gamma_{z} \\
& + \left (- \partial_{t} B^{x} + \partial_{z} E^{y} - \partial_{y} E^{z}\right) \gamma_{t}\wedge \gamma_{y}\wedge \gamma_{z} \\
& + \left (\partial_{x} B^{x} + \partial_{y} B^{y} + \partial_{z} B^{z}\right) \gamma_{x}\wedge \gamma_{y}\wedge \gamma_{z}
\end{aligned}
$$

```
def Dirac_Equation_in_Geog_Calculus():
    Print_Function()
    coords = symbols('t x y z',real=True)
    (st4d,g0,g1,g2,g3) = Ga.build('gamma*t|x|y|z',g=[1,-1,-1,-1],coords=coords)
    I = st4d.i
    (m,e) = symbols('m e')
    psi = st4d.mv('psi','spinor',f=True)
    A = st4d.mv('A','vector',f=True)
    sig_z = g3*g0
    print '\\text{4-Vector Potential\\;\\;}\\bm{A} =',A
    print '\\text{8-component real spinor\\;\\;}\\bm{\\psi} =',psi
    dirac_eq = (st4d.grad*psi)*I*sig_z-e*A*psi-m*psi*g0
    dirac_eq = dirac_eq.simplify()
    dirac_eq.Fmt(3,r'%\text{Dirac Equation\;\;}\nabla \bm{\psi} I \sigma_{z}-e\bm{A}\bm{\psi}-m\bm{\psi}\gamma_{t} = 0')
    return
```

Code Output:

4-Vector Potential $\boldsymbol{A} = A^{t} \gamma_{t} + A^{x} \gamma_{x} + A^{y} \gamma_{y} + A^{z} \gamma_{z}$

8-component real spinor $\boldsymbol{\psi} = \psi + \psi^{tx} \gamma_{t}\wedge \gamma_{x} + \psi^{ty} \gamma_{t}\wedge \gamma_{y} + \psi^{tz} \gamma_{t}\wedge \gamma_{z} + \psi^{xy} \gamma_{x}\wedge \gamma_{y} + \psi^{xz} \gamma_{x}\wedge \gamma_{z} + \psi^{yz} \gamma_{y}\wedge \gamma_{z} + \psi^{txyz} \gamma_{t}\wedge \gamma_{x}\wedge \gamma_{y}\wedge \gamma_{z}$

$$
\begin{aligned}
\text{Dirac Equation}\;\; \nabla \boldsymbol{\psi} I \sigma_{z} - e\boldsymbol{A}\boldsymbol{\psi} - m\boldsymbol{\psi}\gamma_{t} = 0 = & \left (- e A^{t} \psi - e A^{x} \psi^{tx} - e A^{y} \psi^{ty} - e A^{z} \psi^{tz} - m \psi - \partial_{y} \psi^{tx} - \partial_{z} \psi^{txyz} + \partial_{x} \psi^{ty} + \partial_{t} \psi^{xy}\right) \gamma_{t} \\
& + \left (- e A^{t} \psi^{tx} - e A^{x} \psi - e A^{y} \psi^{xy} - e A^{z} \psi^{xz} + m \psi^{tx} + \partial_{y} \psi - \partial_{t} \psi^{ty} - \partial_{x} \psi^{xy} + \partial_{z} \psi^{yz}\right) \gamma_{x} \\
& + \left (e A^{x} \psi^{xy} - e A^{y} \psi - e A^{z} \psi^{yz} + \left (- e A^{t} + m\right) \psi^{ty} - \partial_{x} \psi + \partial_{t} \psi^{tx} - \partial_{y} \psi^{xy} - \partial_{z} \psi^{xz}\right) \gamma_{y} \\
& + \left (e A^{x} \psi^{xz} + e A^{y} \psi^{yz} - e A^{z} \psi + \left (- e A^{t} + m\right) \psi^{tz} + \partial_{t} \psi^{txyz} - \partial_{z} \psi^{xy} + \partial_{y} \psi^{xz} - \partial_{x} \psi^{yz}\right) \gamma_{z} \\
& + \left (e A^{x} \psi^{ty} - e A^{y} \psi^{tx} - e A^{z} \psi^{txyz} + \left (- e A^{t} - m\right) \psi^{xy} - \partial_{t} \psi + \partial_{x} \psi^{tx} + \partial_{y} \psi^{ty} + \partial_{z} \psi^{tz}\right) \gamma_{t}\wedge \gamma_{x}\wedge \gamma_{y} \\
& + \left (- e A^{t} \psi^{xz} + e A^{x} \psi^{tz} + e A^{y} \psi^{txyz} - e A^{z} \psi^{tx} - m \psi^{xz} + \partial_{x} \psi^{txyz} + \partial_{z} \psi^{ty} - \partial_{y} \psi^{tz} - \partial_{t} \psi^{yz}\right) \gamma_{t}\wedge \gamma_{x}\wedge \gamma_{z} \\
& + \left (- e A^{t} \psi^{yz} - e A^{x} \psi^{txyz} + e A^{y} \psi^{tz} - e A^{z} \psi^{ty} - m \psi^{yz} - \partial_{z} \psi^{tx} + \partial_{y} \psi^{txyz} + \partial_{x} \psi^{tz} + \partial_{t} \psi^{xz}\right) \gamma_{t}\wedge \gamma_{y}\wedge \gamma_{z} \\
& + \left (- e A^{t} \psi^{txyz} - e A^{x} \psi^{yz} + e A^{y} \psi^{xz} - e A^{z} \psi^{xy} + m \psi^{txyz} + \partial_{z} \psi - \partial_{t} \psi^{tz} - \partial_{x} \psi^{xz} - \partial_{y} \psi^{yz}\right) \gamma_{x}\wedge \gamma_{y}\wedge \gamma_{z}
\end{aligned}
$$

```
def Lorentz_Tranformation_in_Geog_Algebra():
    Print_Function()
    (alpha,beta,gamma) = symbols('alpha beta gamma')
    (x,t,xp,tp) = symbols("x t x' t'",real=True)
    (st2d,g0,g1) = Ga.build('gamma*t|x',g=[1,-1])
    from sympy import sinh,cosh
    R = cosh(alpha/2)+sinh(alpha/2)*(g0^g1)
    X = t*g0+x*g1
    Xp = tp*g0+xp*g1
    print 'R =',R
    print r"#%t\bm{\gamma_{t}}+x\bm{\gamma_{x}} = t'\bm{\gamma'_{t}}+x'\bm{\gamma'_{x}} = R\lp t'\bm{\gamma_{t}}+x'\bm{\gamma_{x}}\rp R^{\dagger}"
    Xpp = R*Xp*R.rev()
    Xpp = Xpp.collect()
    Xpp = Xpp.trigsimp()
    print r"%t\bm{\gamma_{t}}+x\bm{\gamma_{x}} =",Xpp
    Xpp = Xpp.subs({sinh(alpha):gamma*beta,cosh(alpha):gamma})
    print r'%\f{\sinh}{\alpha} = \gamma\beta'
    print r'%\f{\cosh}{\alpha} = \gamma'
    print r"%t\bm{\gamma_{t}}+x\bm{\gamma_{x}} =",Xpp.collect()
    return
```

Code Output:

$$R = \cosh\left(\frac{\alpha}{2}\right) + \sinh\left(\frac{\alpha}{2}\right)\gamma_t \wedge \gamma_x$$

$$t\boldsymbol{\gamma_t} + x\boldsymbol{\gamma_x} = t'\boldsymbol{\gamma'_t} + x'\boldsymbol{\gamma'_x} = R\left(t'\boldsymbol{\gamma_t} + x'\boldsymbol{\gamma_x}\right)R^{\dagger}$$

$$t\boldsymbol{\gamma_t} + x\boldsymbol{\gamma_x} = \left(t'\cosh\left(\alpha\right) - x'\sinh\left(\alpha\right)\right)\gamma_t + \left(-t'\sinh\left(\alpha\right) + x'\cosh\left(\alpha\right)\right)\gamma_x$$

$$\sinh\left(\alpha\right) = \gamma\beta$$

$$\cosh\left(\alpha\right) = \gamma$$

$$t\boldsymbol{\gamma_t} + x\boldsymbol{\gamma_x} = \gamma\left(-\beta x' + t'\right)\gamma_t + \gamma\left(-\beta t' + x'\right)\gamma_x$$