

# T06P02

June 2, 2020

## 1 T06P02 - MdSM

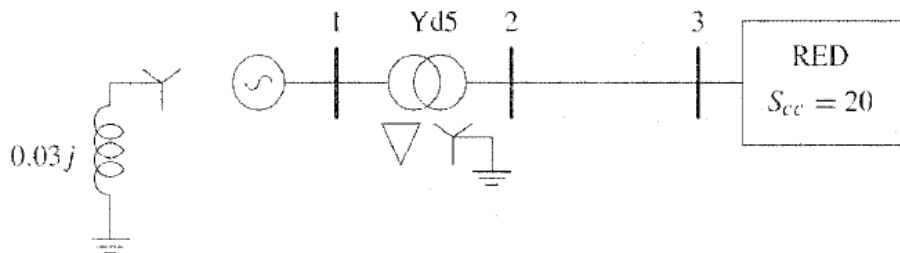
### 1.1 Enunciado

Para la red de la figura con los datos expresados en valores pu bajo una misma base determine:

1. Intensidad por el neutro del transformador para un fallo fase-tierra (monofásico) en la línea, a 5 km de la barra 2 y con una impedancia de falta a tierra  $Z_f = j0.02$  pu.
2. Tensiones en bornas del generador para la condición de falta anterior.
3. Corrientes de cortocircuito en el punto de fallo, para un fallo fase-fase (bifásico aislado) en la línea, en el mismo punto y con la misma impedancia de contacto que en el caso anterior.
4. Componente homopolar de las tensiones en el punto de fallo para el caso anterior (falta bifásica aislada).

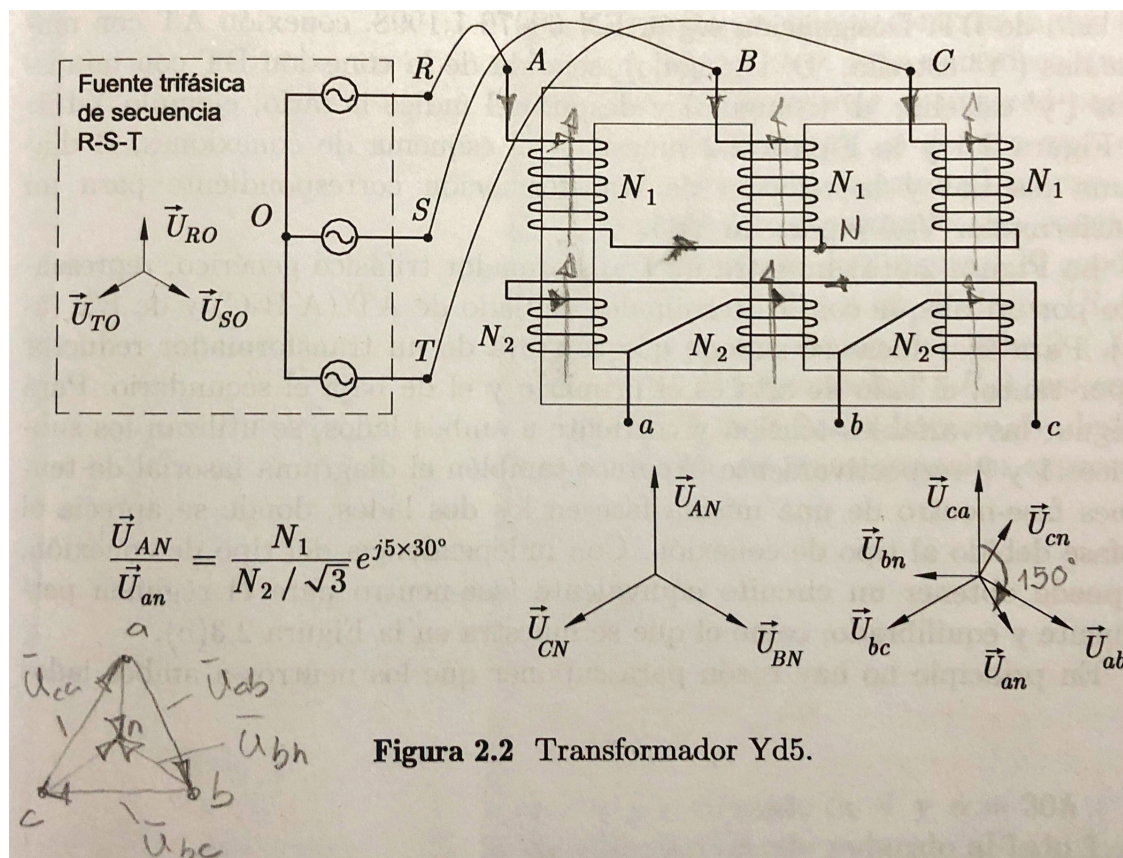
**Sistema eléctrico:**

- Generador:  $\begin{cases} X_0 = 0.04 \\ X_1 = 0.1 \\ X_2 = 0.08 \end{cases}$
- Transformador:  $\begin{cases} X_{cc} = 0.15 \\ X_0 = 0.12 \end{cases}$
- Línea:  $L = 20$  km,  $\begin{cases} X_0 = 0.2 \\ X_1 = X_2 = 0.08 \end{cases}$



**NOTA SOBRE EL DESFASE DEL TRANSFORMADOR:**

Obsérvese el desfase introducido por el índice horario (5) del transformador:



Adap-

tado de [1].

[1] F. Barrero, Sistemas de Energía Eléctrica, Ed. Paraninfo, 2004, vol 1. pág. 35. ISBN: 978-84-9732-283-6.

## 1.2 Resolución

```
[83]: # Importar librerías genéricas:
import numpy as np
import cmath as cm
import matplotlib.pyplot as plt

# Formato general de salida de resultados numéricos:
np.set_printoptions(precision=3, suppress=True)
```

### 1.2.1 Datos del problema

```
[84]: # Impedancias a las tres secuencias [homopolar=0, directa=1 e inversa=2] del
      ↪ generador [pu]:
Zg = np.array([0.04j, 0.1j, 0.08j])
# Impedancias del transformador [pu]:
Zt = np.array([0.12j, 0.15j, 0.15j])
```

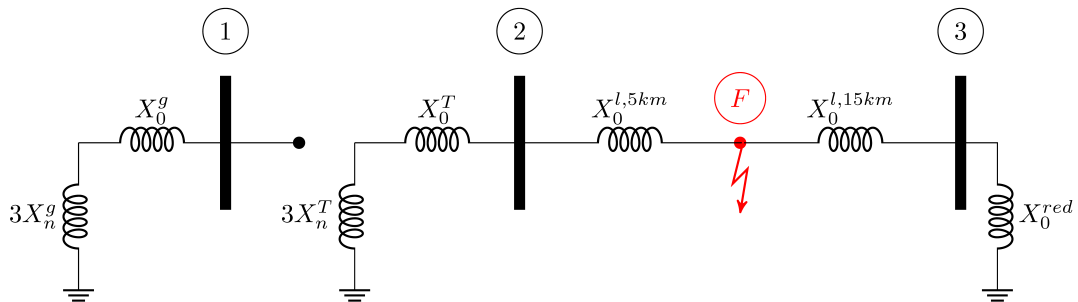
```

# Impedancias absolutas de la línea [pu]:
Zl = np.array([0.20j,0.08j,0.08j])
# Longitud de la línea [km]:
l = 20
# Impedancia de cortocircuito de la red [pu]:
Ucc = 1
Scc = 20
Zcc = (Ucc**2/Scc)*1j
Zred = np.array([Zcc,Zcc,Zcc])
# Tensión del generador:
E = 1
# Desfase del transformador [deg]:
Desf = -150 # Yd5.
# Distancia de la falta a la barra 2 [km]:
lf = 5
# Impedancia de falta/contacto [pu]:
Zf = 0.02j
# Impedancia de puesta a tierra del neutro del generador [pu]:
Zn = 0.03j

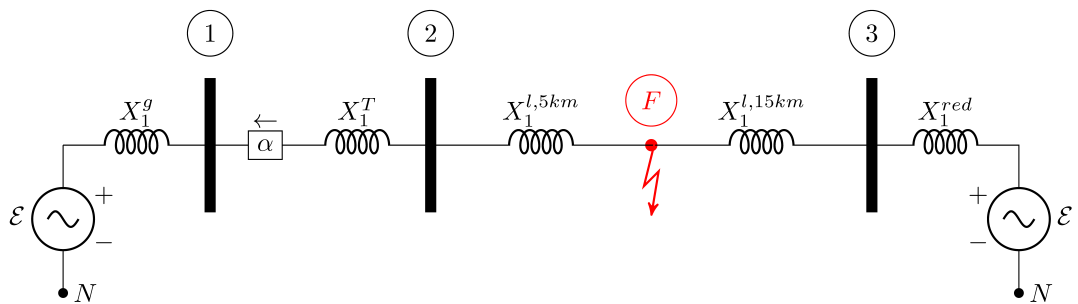
```

### 1.2.2 Redes de secuencia del sistema

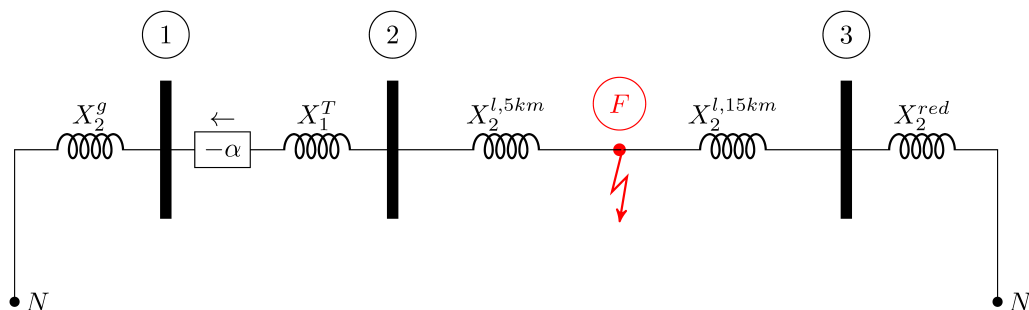
Red de secuencia homopolar/cero (RSH):



Red de secuencia directa (RSD):



### Red de secuencia inversa (RSI):



### 1.2.3 Cálculo de los equivalentes de Thévenin de las redes de secuencia en el punto de falta

### 1.2.4 Red de secuencia homopolar/cero (RSH)

[85]: # El equivalente entre el punto de falta y tierra es el resultado del paralelo  
 ↳ de dos ramas: izquierda (Rama A0) y derecha (Rama B0).

```
RamaA0 = Zt[0] + (lf/1) * Zl[0]
RamaB0 = ((1-lf)/1) * Zl[0] + Zred[0]
Z0 = RamaA0 * RamaB0 / (RamaA0 + RamaB0)
print('Z0 = {:.3j}'.format(Z0))
```

Z0 = 0.0919j

### 1.2.5 Red de secuencia directa (RSD)

[86]: # Ídem que la RSH, teniendo en cuenta que la rama de la izquierda (Rama A1) ↳  
 ↳ llega hasta el generador.

```
RamaA1 = Zg[1] + Zt[1] + (lf/1) * Zl[1]
RamaB1 = ((1-lf)/1) * Zl[1] + Zred[1]
Z1 = RamaA1 * RamaB1 / (RamaA1 + RamaB1)
print('Z1 = {:.3j}'.format(Z1))
```

Z1 = 0.0782j

Obsérvese que la fem de la RSD correspondería con la tensión del equivalente de Thévenin en el punto de análisis. Nótese en este punto que **NO se van a incluir los desfases introducidos por el transformador, los cuáles se tendrán en cuenta sólo a efectos de referirnos a magnitudes del generador**, es decir, a efectos prácticos:

$$\varepsilon_1^{Th} = 1 \angle 0^\circ.$$

[87]: # FEM de la RSD: Tensión del equivalente de Thévenin en el punto de análisis.

```
E1 = E
print('Tensión del equivalente Thévenin E1 = {:.3f}'.format(E1))
```

Tensión del equivalente Thévenin  $E_1 = 1.000$

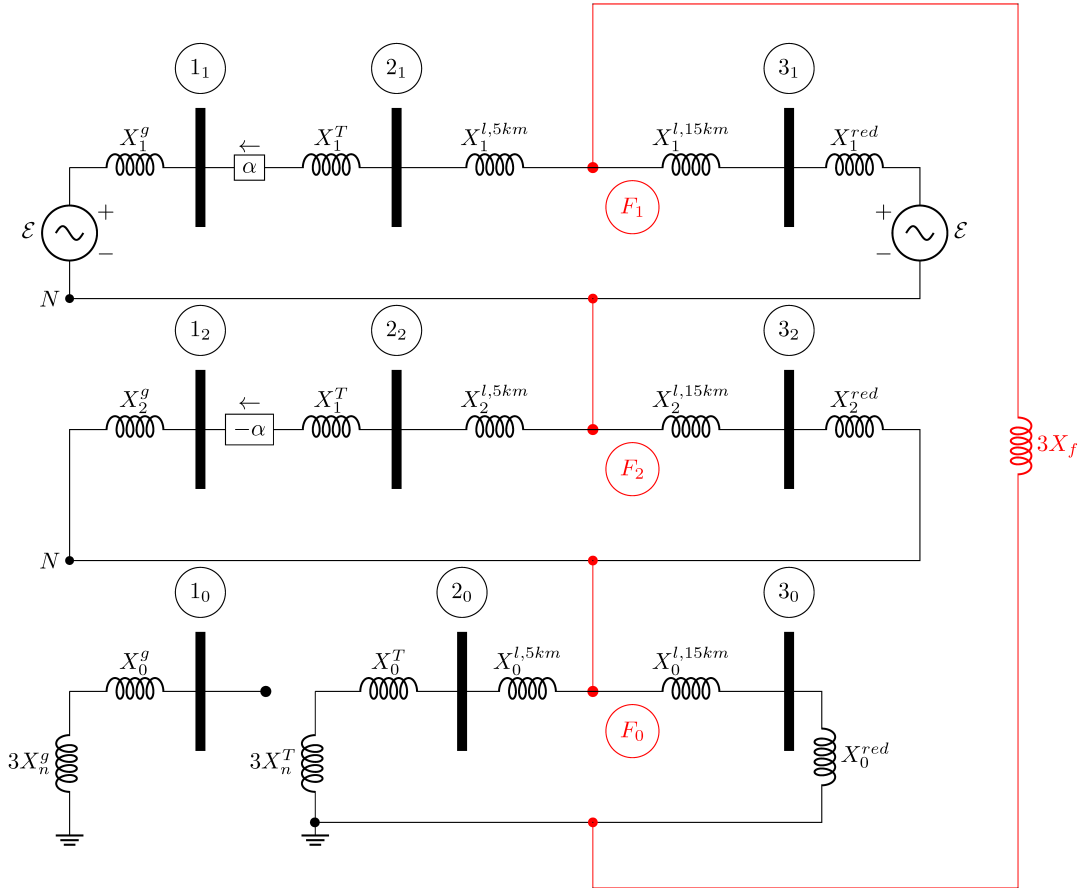
### 1.2.6 Red de secuencia inversa (RSI)

```
[88]: # Ídem que la RSD.
RamaA2 = Zg[2]+Zt[2]+(1f/1)*Zl[2]
RamaB2 = ((1-1f)/1)*Zl[2]+Zred[2]
Z2 = RamaA2*RamaB2/(RamaA2+RamaB2)
print('Z2 = {:.3j}'.format(Z2))
```

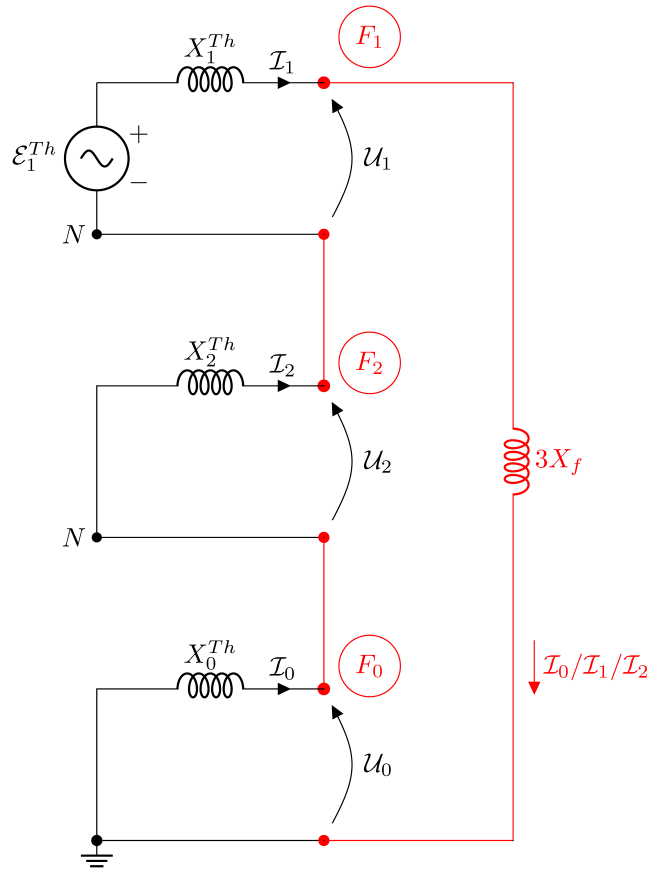
$Z2 = 0.0764j$

### 1.2.7 Asociación de redes de secuencia (falta monofásica)

Ante falta monofásica a tierra en el punto de falta  $F$ , las tres redes de secuencia deben asociarse en serie y considerar la impedancia de falta:



Considerando los equivalentes de Thévenin de las tres redes, el esquema queda simplificado:



[89]: *# Como es falta monofásica, las tres redes se asocian en serie, y por lo tanto*  
*→ el cálculo de las corrientes de las tres redes de secuencia queda:*

```
Z012 = np.array([Z0,Z1,Z2])
I0 = E1/(Z012[0]+Z012[1]+Z012[2]+3*Zf)
I1 = I0
I2 = I0
I012 = np.array([I0,I1,I2])
print('I012=',I012)
```

I012= [0.-3.263j 0.-3.263j 0.-3.263j]

[90]: *# Por su parte, el cálculo de las tensiones de las tres redes se obtendría*  
*→ mediante:*

```
U0 = -I012[0]*Z012[0]
U1 = E1-I012[1]*Z012[1]
U2 = -I012[2]*Z012[2]
U012 = np.array([U0,U1,U2])
print('U012=',U012)
```

U012= [-0.3 +0.j 0.745+0.j -0.249+0.j]

Ante la falta anterior, la corriente circulante por el neutro del transformador sería igual a la suma vectorial de las corrientes de las tres fases, o lo que es lo mismo, tres veces la corriente homopolar en dicho elemento, que en este caso sería la contribución de la rama de la izquierda (Rama A0) a la corriente homopolar calculada:

$$\mathcal{I}_n^T = \mathcal{I}_R + \mathcal{I}_S + \mathcal{I}_T = 3\mathcal{I}_0^T = 3\mathcal{I}_0^A.$$

```
[91]: # Obtenemos la corriente homopolar del transformador mediante un divisor de
      ↪ intensidades:
It0 = (RamaB0/(RamaA0+RamaB0))*I012[0]
In = 3*It0
print('La corriente por el neutro del transformador será',round(np.
      ↪ abs(In),3),'pu')
```

La corriente por el neutro del transformador será 5.292 pu

Por otro lado, la tensión en bornes del generador, es decir, la tensión en la barra 1, se puede obtener mediante la transformación al dominio de fases de las componentes simétricas de la tensión en dicha barra (ante la situación de falta estudiada). Para ello:

1. Determinaremos las componentes simétricas de la tensión en la barra 1, las cuáles obtendremos a partir de las componentes simétricas de las corrientes en el generador y las respectivas caídas de tensión en su impedancia interna:

$$\mathcal{U}_1^g = \mathcal{E} - \mathcal{I}_1^A \mathcal{Z}_1^g.$$

$$\mathcal{U}_2^g = -\mathcal{I}_2^A \mathcal{Z}_2^g.$$

$$\mathcal{U}_0^g = -\mathcal{I}_0^A \mathcal{Z}_0^g.$$

2. Transformación al dominio de fases de las componentes simétricas de la tensión mediante la Matriz de Fortescue ( $[T]$ ):

$$[\mathcal{U}_{RST}^g] = [T] [\mathcal{U}_{012}^g].$$

**NOTA IMPORTANTE:** Obsérvese que el transformador tiene un índice horario no nulo, lo que significa que introduce un desfase en la RSD, y la opuesta en la RSI. Así, para valores a partir del secundario del transformador, el desfase introducido (en este caso de valor  $\alpha = -150^\circ$ ) debe sumarse en la RSD, mientras que debe restarse en la RSI:

$$\mathcal{U}_1^{bus1} = \mathcal{U}_1^{bus2} e^{-j150^\circ}.$$

$$\mathcal{U}_2^{bus1} = \mathcal{U}_2^{bus2} e^{j150^\circ}.$$

**NOTA 2:** Para saber el sentido en el cual aplicar el desfase, obsérvese en este caso que la tensión en el secundario (bus 1) está atrasada 150 grados ( $\phi - 150^\circ$ ) respecto al primario (bus 2).

```
[92]: ## Componentes simétricas de la tensión en la barra 1
# Componente homopolar:
Ug0 = 0
# Componente directa:
It1 = (RamaB1/(RamaA1+RamaB1))*I012[1]
Ug1 = E-It1*Zg[1]
# Aplico el desfase del transformador:
Ug1 = Ug1*cm.rect(1,Desf*cm.pi/180)
E = E*cm.rect(1,Desf*cm.pi/180)
print('Fuerza electromotriz del generador E = {:.3}'.format(E))
# Componente inversa:
It2 = (RamaB2/(RamaA2+RamaB2))*I012[2]
Ug2 = -It2*Zg[2]
# Aplico el desfase del transformador:
Ug2 = Ug2*cm.rect(1,-Desf*cm.pi/180)
# Componentes simétricas de la tensión en bornes del generador:
Ug012 = np.array([Ug0,Ug1,Ug2])
print('Ug012=',Ug012)
# Componentes simétricas de la tensión en bornes del generador en coordenadas
↳polares:
Ug012polar = np.zeros([3,2])
for i in range(0,3):
    Ug012polar[i] = cm.polar(Ug012[i])
    Ug012polar[i][1] = Ug012polar[i][1]*180/cm.pi
print('Ug012polar = ',Ug012polar)
```

```
Fuerza electromotriz del generador E = (-0.866-0.5j)
Ug012= [ 0.   +0.j   -0.784-0.453j  0.069-0.04j ]
Ug012polar = [[ 0.   0.   ]
 [ 0.906 -150.  ]
 [ 0.08  -30.   ]]
```

De forma alternativa, se podría haber considerado en la resolución que la fem del generador y de la red tienen fase nula, es decir:

$$\varepsilon^g = \varepsilon^{red} = 1\angle 0^\circ \text{ pu},$$

y por lo tanto, considerando el desfase del transformador:

$$\varepsilon_1^{Th} = 1\angle 0^\circ \text{ pu} \cdot 1\angle 150^\circ = 1\angle 150^\circ \text{ pu}.$$

Si se considera así, los resultados obtenidos coinciden en módulo y relación de fases, pero quedan desfasados respecto a la solución anterior al haber tomado ahora como referencia de fases la fuerza electromotriz del generador, es decir:  $\mathcal{E} = E\angle 0^\circ$ . La siguiente celda demuestra lo expuesto.



```
[93]: # Aplicamos el desfase del transformador a la fem de la fuente equivalente:
E1 = 1*cm.rect(1,-Desf*cm.pi/180)
print('Tensión del equivalente Thévenin E1 = {:.3j}'.format(E1))

# Recalculamos las corrientes y tensiones en el punto de falta:
I0 = E1/(Z012[0]+Z012[1]+Z012[2]+3*Zf)
I1 = I0
I2 = I0
I012 = np.array([I0,I1,I2])

U0 = -I012[0]*Z012[0]
U1 = E1-I012[1]*Z012[1]
U2 = -I012[2]*Z012[2]
U012 = np.array([U0,U1,U2])

## Calculamos las componentes simétricas de la tensión en bornes del generador:
# Componente homopolar:
Ug0 = 0
# Componente directa:
It1 = (RamaB1/(RamaA1+RamaB1))*I012[1]
# Aplicación del desfase del transformador a la componente directa de la
↳ corriente en el lado del generador:
It1 = It1*cm.rect(1,Desf*cm.pi/180)
E = 1
print('Fuerza electromotriz del generador E = {:.3f}'.format(E))
Ug1 = E-It1*Zg[1]
# Componente inversa:
It2 = (RamaB2/(RamaA2+RamaB2))*I012[2]
# Aplicación del desfase del transformador a la componente inversa de la
↳ corriente en el lado del generador:
It2 = It2*cm.rect(1,-Desf*cm.pi/180)
Ug2 = -It2*Zg[2]
# Componentes simétricas de la tensión en bornes del generador:
Ug012 = np.array([Ug0,Ug1,Ug2])
print('Ug012 = ',Ug012)
# Componentes simétricas de la tensión en bornes del generador en coordenadas
↳ polares:
Ug012polar = np.zeros([3,2])
for i in range(0,3):
    Ug012polar[i] = cm.polar(Ug012[i])
    Ug012polar[i][1] = Ug012polar[i][1]*180/cm.pi
print('Ug012polar = ',Ug012polar)
```

```
Tensión del equivalente Thévenin E1 = (-0.866+0.5j)
Fuerza electromotriz del generador E = 1.000
Ug012 = [ 0.    +0.j      0.906+0.j      -0.04 +0.069j]
Ug012polar = [[ 0.      0.    ]
```

```
[ 0.906  0.   ]
[ 0.08  120.  ]]
```

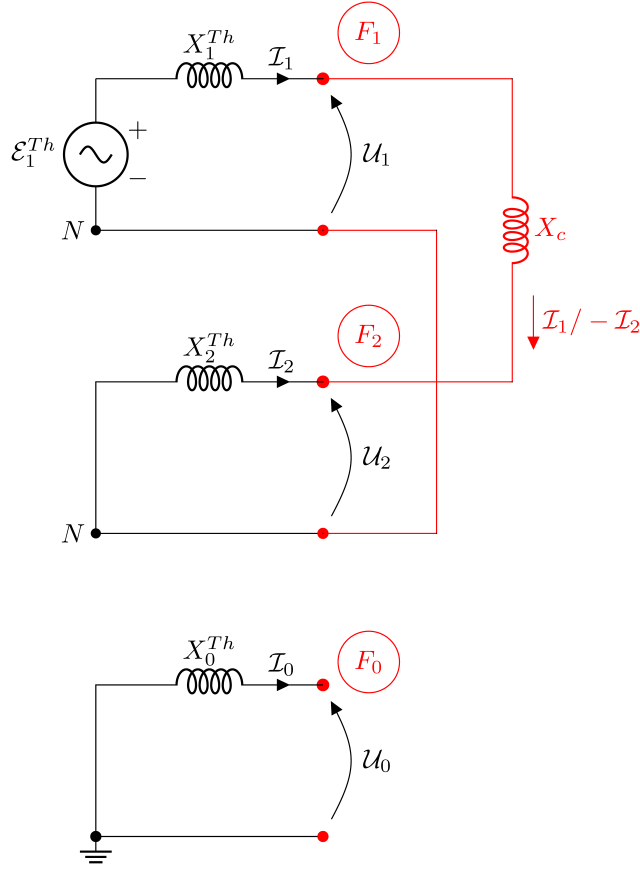
```
[94]: ## Transformación de las componentes simétricas a las componentes de fase
      ↪(cambio de base)
      # Matriz de Fortescue-Stovkis
      a = cm.rect(1,120*cm.pi/180)
      T = np.array([[1,1,1],[1,a**2,a],[1,a,a**2]])
      UgRST = np.dot(T,np.transpose(Ug012))
      print('Tensión en bornes del generador (coordenadas rectangulares) =',UgRST)
      # Tensión en bornes del generador en coordenadas polares:
      UgRSTpolar = np.zeros([3,2])
      for i in range(0,3):
          UgRSTpolar[i] = cm.polar(UgRST[i])
          UgRSTpolar[i][1] = UgRSTpolar[i][1]*180/cm.pi
      print('Tensión en bornes del generador (coordenadas polares) =',UgRSTpolar)
```

```
Tensión en bornes del generador (coordenadas rectangulares) = [ 0.866+0.069j
-0.493-0.853j -0.373+0.784j]
```

```
Tensión en bornes del generador (coordenadas polares) = [[ 0.868 4.563]
[ 0.985 -120. ]
[ 0.868 115.437]]
```

### 1.2.8 Asociación de redes de secuencia (falta bifásica)

Ante falta bifásica aislada en el punto de falta  $F$ , las tres redes de secuencia deben asociarse en paralelo y considerar la impedancia de contacto. Como la falta es aislada de tierra, puede observarse que la RSH queda igualmente aislada (en circuito abierto). Considerando los equivalentes de Thévenin de las tres redes, el esquema queda simplificado:



Ante la falta bifásica descrita, se deduce que la corriente de la RSH es nula, y las corrientes de las RSD y RSI son iguales y opuestas en signo y se calculan fácilmente como:

$$\mathcal{I}_1^F = -\mathcal{I}_2^F = \frac{\mathcal{E}_1^{Th}}{X_1^{Th} + X_c + X_2^{Th}}.$$

```
[95]: E = 1
      I0 = 0
      I1 = E/(Z012[1]+Z012[2]+Zf)
      I2 = -I1
      I012 = np.array([I0,I1,I2])
      print('I012 = ',I012)
```

```
I012 = [ 0.+0.j      0.-5.729j -0.+5.729j]
```

De forma análoga, la tensión de la RSH resulta inmediata al estar en circuito abierto y no existir f.e.m. en esta secuencia, mientras que las tensiones de las RSD y RSI se obtienen como:

$$\mathcal{U}_1^F = \mathcal{E}_1^{Th} - \mathcal{I}_1 \mathcal{Z}_1.$$

$$\mathcal{U}_2^F = -\mathcal{I}_2 \mathcal{Z}_2.$$

```
[96]: U0 = 0
      U1 = E-I012[1]*Z012[1]
      U2 = -I012[2]*Z012[2]
      U012 = np.array([U0,U1,U2])
      print('U012 = ',U012)
```

```
U012 = [0. +0.j 0.552+0.j 0.438+0.j]
```

```
[97]: ## Transformación de las componentes simétricas a las componentes de fase
      ↪(cambio de base)
      # Corrientes:
      IRST = np.dot(T,np.transpose(I012))
      IRSTpolar = np.zeros([3,2])
      for i in range(0,3):
          IRSTpolar[i] = cm.polar(IRST[i])
          IRSTpolar[i][1] = IRSTpolar[i][1]*180/cm.pi
      print('Corriente en el punto de falta (coordenadas polares) = ',IRSTpolar)
      # Tensiones:
      URST = np.dot(T,np.transpose(U012))
      URSTpolar = np.zeros([3,2])
      for i in range(0,3):
          URSTpolar[i] = cm.polar(URST[i])
          URSTpolar[i][1] = URSTpolar[i][1]*180/cm.pi
      print('Tensión en el punto de falta (coordenadas polares) = ',URSTpolar)
```

```
Corriente en el punto de falta (coordenadas polares) = [[ 0.      0.  ]
 [ 9.923 180.  ]
 [ 9.923 -0.  ]]
Tensión en el punto de falta (coordenadas polares) = [[ 0.99      0.  ]
 [ 0.505 -168.663]
 [ 0.505 168.663]]
```