



RED
ELÉCTRICA
DE ESPAÑA

Grupo Red Eléctrica



CITCEA

Modelo de convertidor *grid-following* en PSCAD

Versión 2.0

CITCEA
Universitat Politècnica de Catalunya

23 de marzo de 2022



Control de cambios

Versión	Descripción
1.0 (27/09/2021)	Primera versión
2.0 (23/03/2022)	<p>Modificaciones principales en respuesta a los comentarios de Javier Renedo (REE):</p> <ul style="list-style-type: none">• Se ha añadido el diseño de los parámetros de control de la PLL, lazo de corriente y lazos externos.• Se ha modificado el modelo de la carga (de potencia constante a corriente constante), que causaba problemas en la recuperación de la falta.• Se ha eliminado la sección del efecto del paso de integración sobre el cálculo RMS, ya que se ha corregido tras el cambio de modelo de carga.• Se han añadido los escenarios 9 (cambio de SCR de la red) y 10 (desconexión de la red).

Índice

1	Introducción	1
2	Descripción del modelo en PSCAD	2
2.1	Modelo de la red: equivalente Thévenin.....	3
2.2	Carga activa: fuente de corriente	5
2.3	Convertidor con control grid-following	6
2.3.1	Phase Locked Loop (PLL)	7
2.3.2	Lazo de corriente	8
2.3.3	Cálculo de referencias de corriente – controles externos	10
2.3.4	Control para operación transitoria – inyección rápida de corriente reactiva	13
2.3.5	Limitación de corriente	15
3	Descripción de las simulaciones	17
3.1	Código de Python para ejecutar el modelo	17
4	Resultados de los escenarios 1-10	22
4.1	Escenario 1: sin perturbaciones.....	22
4.2	Escenario 2: cambio de Pref	24
4.3	Escenario 3: cambio de Qref.....	25
4.4	Escenario 4: cambio de ángulo de la red	26
4.5	Escenario 5: cambio de frecuencia de la red	27
4.6	Escenario 6: cambio de tensión de la red	28
4.7	Escenario 7: faltas simétricas	29
4.8	Escenario 8: faltas asimétricas.....	31
4.9	Escenario 9: cambio de SCR de la red	32
4.10	Escenario 10: desconexión de la red	33
5	Conclusiones.....	34
6	Referencias	34



1 Introducción

Este documento forma parte del proyecto I+D K-ER53-IDGFC, en el que están involucrados la Universidad Carlos III de Madrid (UC3M) y la Universitat Politècnica de Catalunya (UPC), coordinados por Red Eléctrica de España (REE). Uno de los objetivos de este proyecto es desarrollar controles *grid-forming* por parte de ambas universidades para ser utilizados por miembros de REE, pero también implementar otros modelos, como un convertidor *grid-following*, generadores síncronos o cargas.

En este documento se detalla el modelo de un convertidor con controles *grid-following* implementado en PSCAD. La operación *grid-following* es la más utilizada actualmente por los convertidores conectados a la red. Su funcionamiento se basa en la detección de la frecuencia y la fase de la tensión en su punto de conexión a la red para realizar la posterior sincronización del convertidor.

Esta estructura de control se ha evaluado en un sistema reducido, propuesto por la *FNN Guideline* [1]. Dicho sistema está compuesto por un equivalente de la red, una carga activa y el propio convertidor. Se han considerado diferentes eventos, como cambios de la potencia de referencia del convertidor, cambios del ángulo de la red o faltas, para evaluar todas las funcionalidades del control del convertidor. Este análisis se ha realizado mediante simulaciones EMT implementadas en PSCAD.

Por tanto, los objetivos de este documento son los siguientes:

- Describir una estructura de control *grid-following* y su implementación en PSCAD
- Describir la red de prueba y los eventos seleccionados para evaluar los controles del convertidor

El modelo de PSCAD *GFOL_model_v2.pscx* incluye el modelo del convertidor con los controles *grid-following*. La Figura 1 muestra la pantalla principal del modelo, donde se pueden identificar dos secciones fundamentales: el modelo físico y los resultados.



Red: equivalente Thevenin

Grid-following VSC

Carga activa
Fuente de corriente

Diagram illustrating the power system configuration for a Grid-following VSC (Voltage Source Converter) connected to a grid.

The system consists of three main components:

- Red: equivalente Thevenin**: A voltage source V_{grid} in series with impedance Z_1 , connected to a parallel combination of impedance Z_2 and a switch S_2 .
- Grid-following VSC**: A voltage source converter (VSC) connected to the grid through a series inductor L_f and resistor R_f . The VSC output voltage is V_{VSC} .
- Carga activa**: An active load represented by a current source.

The diagram shows the power flow and control variables:

- Power Flow**:
 - P_{grid} and Q_{grid} (red arrows) represent power flow from the grid to the VSC.
 - I_{VSC} , P_{VSC} , and Q_{VSC} (blue arrows) represent power flow from the VSC to the grid.
 - P_{load} and Q_{load} (green arrows) represent power flow from the VSC to the active load.
- Grid Connection**: The VSC is connected to the grid through a circuit breaker (CB) and a grid connection point.

Figura 2. Red de pruebas basada en la topología descrita en la Guía FNN

Esta red se ha implementado en PSCAD como se muestra en la Figura 3.

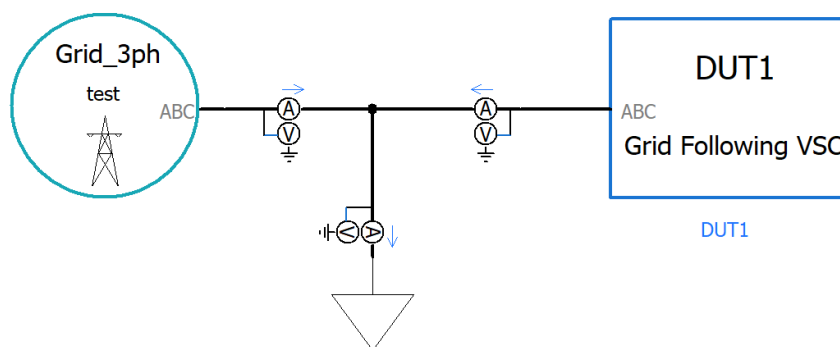


Figura 3. Implementación de la red de pruebas en PSCAD

2.1 Modelo de la red: equivalente Thévenin

El bloque de la red, mostrado en la Figura 4, consta de tres fuentes de tensión, una para cada fase, y dos impedancias en paralelo, $Z1$ y $Z2$, que permiten modificar el *Short-Circuit Ratio* (SCR) de la red. Estas impedancias se han diseñado de tal forma que cuando ambas están conectadas se tiene un $SCR = 3$, mientras que cuando se desconecta $Z2$, el SCR baja a 0,5. En ambos casos se ha utilizado una relación X/R igual a 3. El cálculo de estas impedancias se realiza en el código de Python descrito en la Sección 3.1, donde se puede modificar tanto el SCR como X/R , obteniendo los valores de $R1$, $L1$, $R2$ y $L2$ que finalmente se introducen en el modelo. La conexión y desconexión de estas impedancias se controla con las variables tS_1 y tS_2 , que accionan los interruptores $S1$ y $S2$.

Este bloque también permite ejecutar cambios del ángulo, frecuencia y magnitud de la tensión de la red. En particular, el modelo permite realizar cuatro cambios para cada magnitud indicando el tiempo en el que se quiere ejecutar el cambio y el valor final. Por ejemplo, si se quiere iniciar con una tensión de 1 pu, se indicará que la variable Ull_0 sea 1 (pu). Si en $t = 2$ s se quiere que esta tensión varíe hasta 1,1 pu, se indicará $tu_1 = 2$ (s) y $Ull_1 = 1,1$ (pu), lo que provocará un cambio en escalón de la tensión de la red desde 1 pu hasta 1,1 pu en $t = 2$ s. Los cambios de frecuencia se han limitado a una variación de 4 Hz/s.

Por último, se ha añadido la posibilidad de añadir faltas en el punto de conexión. Se han incluido cuatro faltas a tierra, dos trifásicas y dos monofásicas. Los parámetros que definen su funcionamiento son el tiempo de inicio de la falta, su duración y la resistencia.

Todos los parámetros anteriores se definen en el menú de parámetros del bloque de la red, mostrada en la Figura 4. Se puede emplear el código de Python descrito en la Sección 3.1 para ejecutar los escenarios descritos en la Sección 3.



2.2 Carga activa: fuente de corriente

La carga activa se ha implementado como fuente de corriente, tal y como se muestra en la Figura 5. Este bloque se ha diseñado para permitir cuatro cambios de carga durante la simulación, por lo que sus entradas son los tiempos en el que se desea realizar los cambios, los diferentes valores de la carga en pu y la potencia base. Se ha seleccionado un factor de potencia unitario, por lo que sólo se consume potencia activa. Se ha incluido una PLL, que permite sincronizar la fuente de corriente con la red.

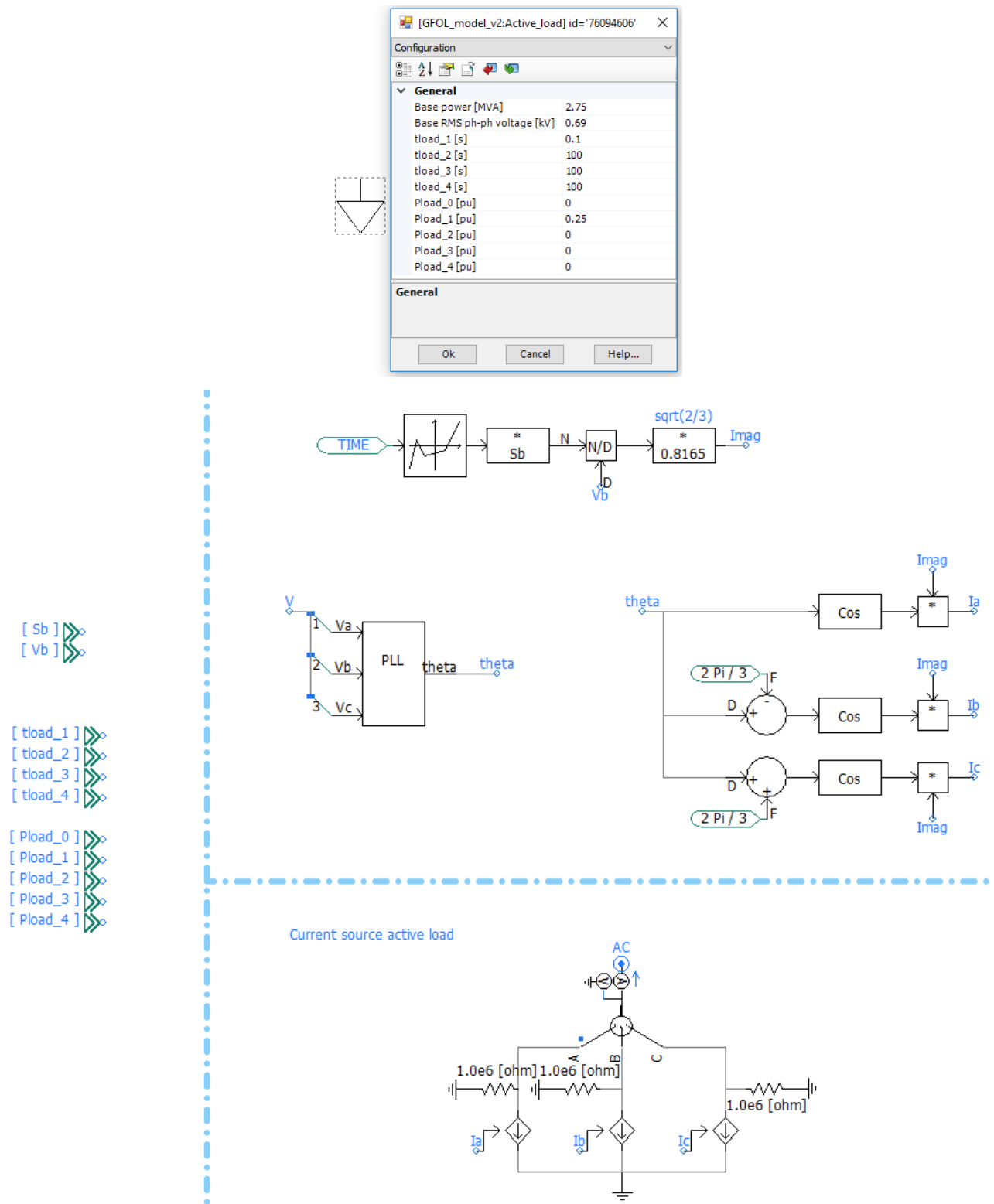


Figura 5. Implementación de la carga activa en PSCAD

2.3 Convertidor con control *grid-following*

Si se accede al bloque del convertidor, se llega a la pantalla mostrada en la Figura 7, donde se encuentran el modelo del convertidor y los diferentes bloques del control. El convertidor se ha implementado como un modelo promediado, representado por fuentes de tensión controladas, mientras que el filtro de conexión a la red es de tipo RL. El control está basado en una estructura en cascada, en la que el control interno corresponde con el control de corriente, cuyas referencias se generan en los lazos externos. El control de corriente se ha implementado en *Double Synchronous Reference Frame*, lo que permite controlar las secuencias positiva y negativa de la corriente. En este documento se ha empleado la siguiente transformada de Park $T(\theta)$ para obtener las componentes qd empleadas en el control del convertidor:

$$T(\theta) = \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \end{bmatrix}. \quad (1)$$

Entonces, las componentes qd se pueden obtener aplicando (1) a las componentes trifásicas abc :

$$x_{qd} = T(\theta)x_{abc} \quad (2)$$

La Figura 6 muestra la representación de las componentes qd en el plano abc .

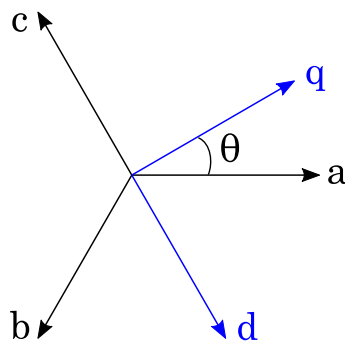
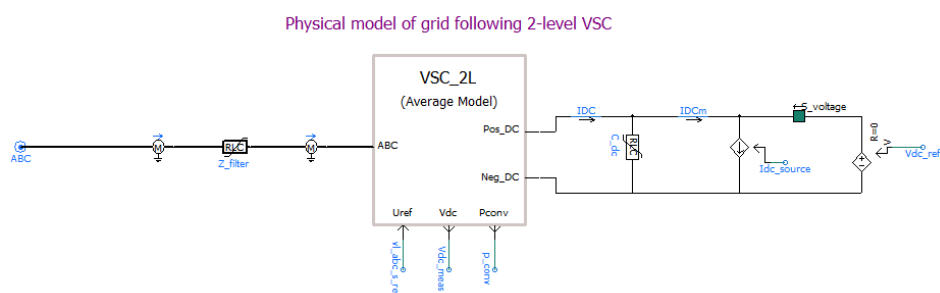


Figura 6. Representación del plano qd empleado para en el control

En este modelo se pueden seleccionar lazos de control de potencia activa, tensión DC y potencia reactiva, junto con los droops de frecuencia y tensión AC. De esta manera, los bloques que se pueden encontrar en el modelo son: Phase-Locked Loop (PLL), el control de corriente (Current Controller), el limitador de corriente (Current Limiter) y cálculo de referencias de corriente (Current Reference Calculation). También se ha incluido la detección de subtensiones y sobretensiones, lo que modifica el modo de operación del convertidor. En las siguientes secciones se detallan los bloques de control anteriores y sus características principales.



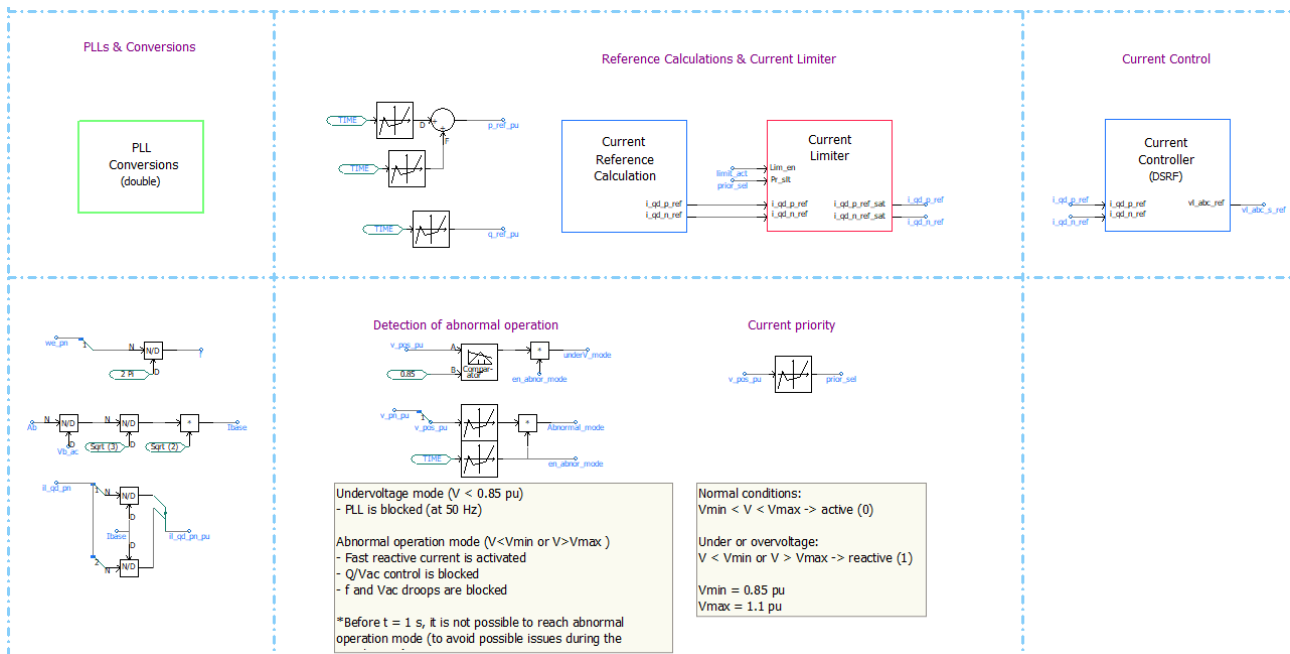


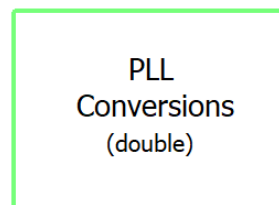
Figura 7. Modelo del convertidor con control *grid-following* en PSCAD

2.3.1 Phase Locked Loop (PLL)

Las estructuras de control *grid-following* basan su funcionamiento en la existencia de una red AC externa, con la que el convertidor tiene que sincronizarse. Para ello típicamente se utiliza una *Phase Locked Loop* (PLL), que estima la frecuencia y la fase de la red AC midiendo la tensión en el Punto de Conexión (Point Of Connection, POC). Esta PLL se ha diseñado siguiendo el método propuesto en [2] para una respuesta con un tiempo de establecimiento de 100 ms y un amortiguamiento de 0,707. El cálculo de los parámetros de la PLL (k_{p-pll} y k_{i-pll}) se ejecutan en el código de Python.

En este modelo se ha implementado una *Double Synchronous Reference Frame PLL* (DSRF-PLL), que mide tanto la secuencia positiva como la negativa de la tensión en el POC y la corriente intercambiada por el convertidor. Esto permite implementar controles independientes en el control para ambas secuencias, por lo que el convertidor podría proporcionar soporte a la red durante desequilibrios.

La Figura 8 muestra cómo se ha implementado la DSRF-PLL en PSCAD. Además de las componentes qd de las tensiones y corrientes en secuencia negativa y positiva, este bloque proporciona la magnitud de las secuencias positiva y negativa de la tensión. La secuencia positiva se emplea para detectar la operación transitoria durante subtensiones y sobretensiones y para calcular la referencia para la inyección rápida de corriente reactiva. La secuencia negativa de la tensión únicamente se emplea para generar la referencia de inyección rápida de corriente reactiva de secuencia negativa. Se pueden encontrar más detalles sobre la operación transitoria del convertidor durante subtensiones o sobretensiones en la Sección 2.3.4.



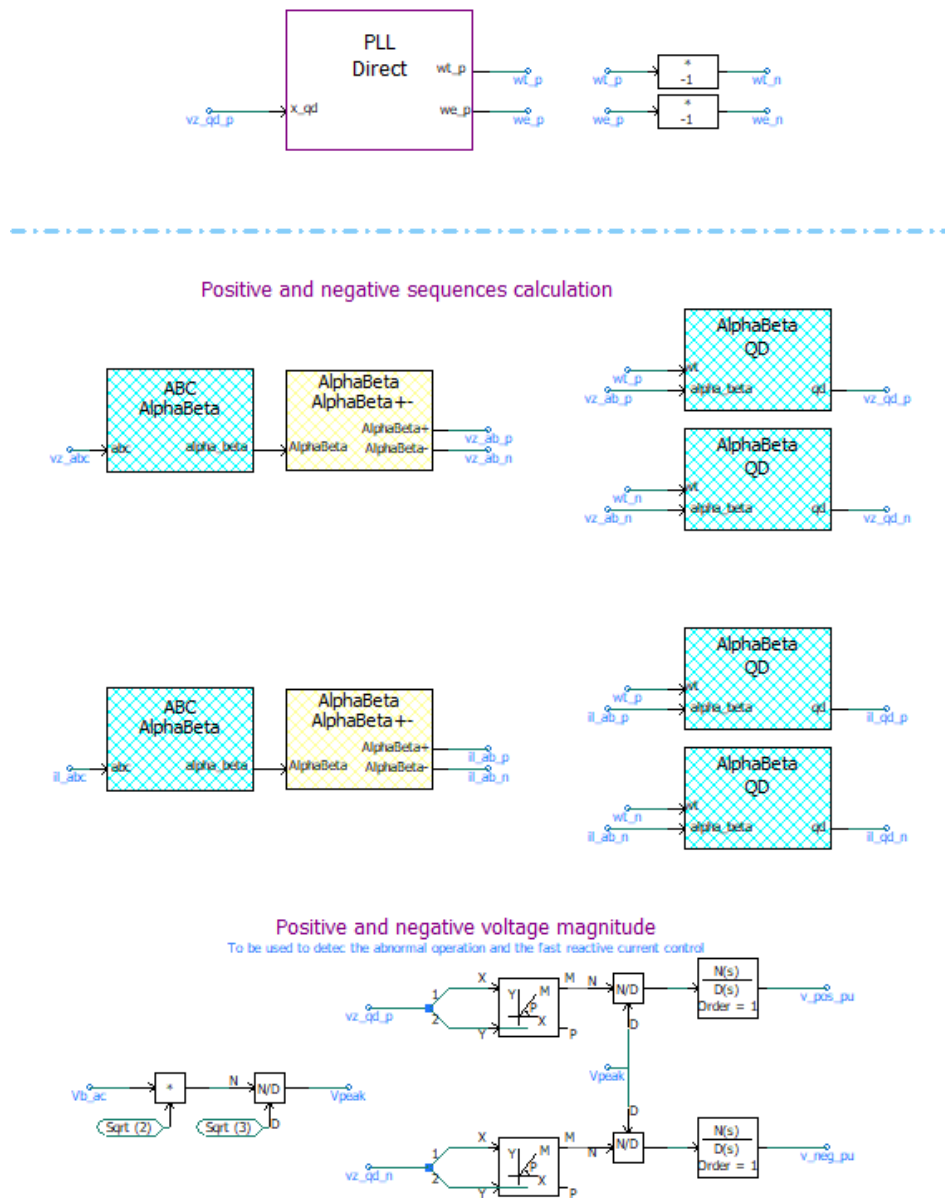


Figura 8. Implementación de la DSRF-PLL en PSCAD

2.3.2 Lazo de corriente

El lazo de control de corriente se ha implementado en *Double Synchronous Reference Frame*, lo que permite controlar tanto la corriente en secuencia positiva como negativa. Se ha utilizado la misma estructura para ambas secuencias, que está basada en dos controladores PI tal y como se muestra en la Figura 9.

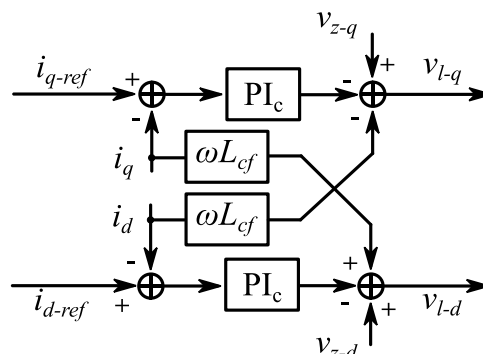


Figura 9. Esquema del control de corriente

La Figura 10 muestra la implementación del control de corriente en PSCAD. No es necesario modificar ningún parámetro del control de corriente, ya éstos están definidos localmente.

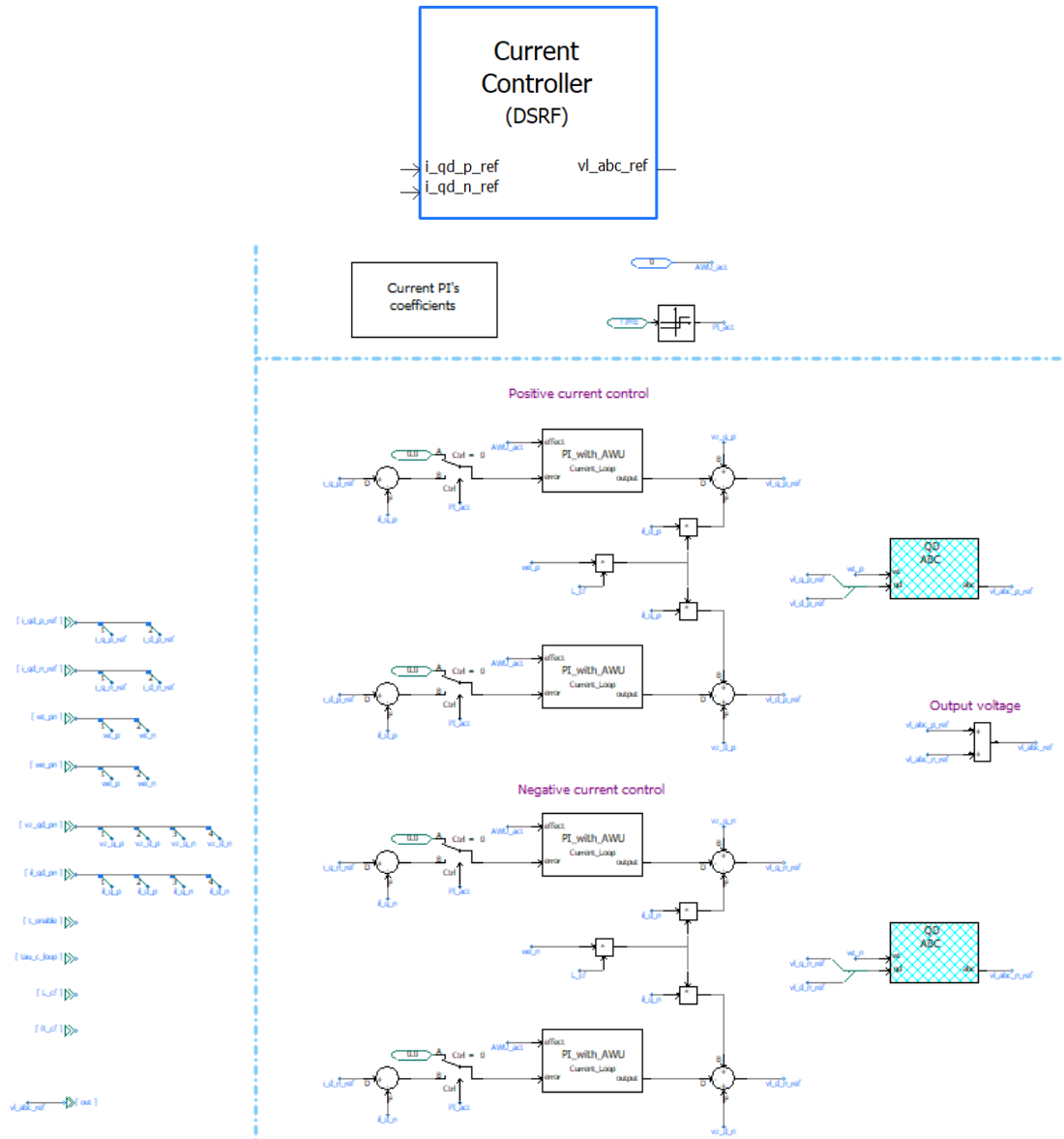


Figura 10. Implementación del lazo de corriente en PSCAD

Los parámetros de los controladores se calculan internamente en el bloque de control de corriente, diseñados siguiendo el método *Internal Model Control* (IMC) [3]:

$$k_{p-i} = \frac{L_{cf}}{\tau_c} \quad k_{i-i} = \frac{R_{cf}}{\tau_c} \quad (3)$$

Donde R_{cf} y L_{cf} son la resistencia e inductancia del filtro del convertidor y τ_c es la constante de tiempo deseada. Este cálculo se muestra en la Figura 11. Este diseño permite obtener una respuesta de corriente de primer orden con la constante de tiempo τ_c definida por el usuario.

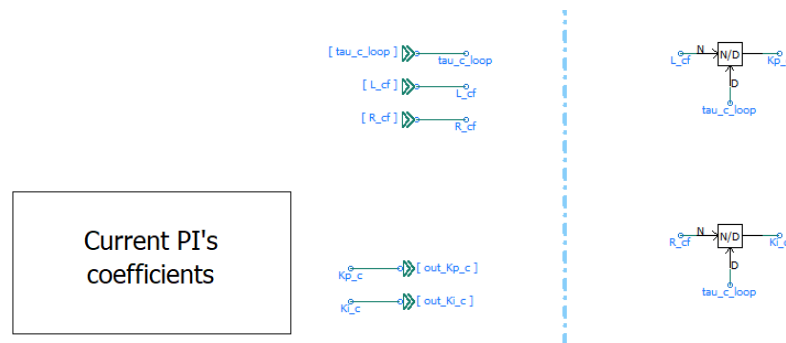


Figura 11. Cálculo de los parámetros del control de corriente

Únicamente es necesario indicar la constante de tiempo deseada para este lazo de control, que se define en el menú de parámetros del bloque general del convertidor. En la Figura 12 se ha escogido una constante de tiempo de 1 ms.

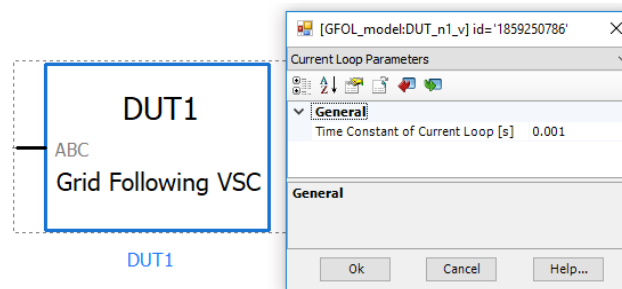


Figura 12. Asignar constante de tiempo para el lazo de control de corriente

2.3.3 Cálculo de referencias de corriente – controles externos

El bloque de cálculo de referencias de corriente incluye los siguientes controles:

- Referencia de corriente activa
 - Control de tensión DC
 - Control de potencia activa
 - Droop de frecuencia

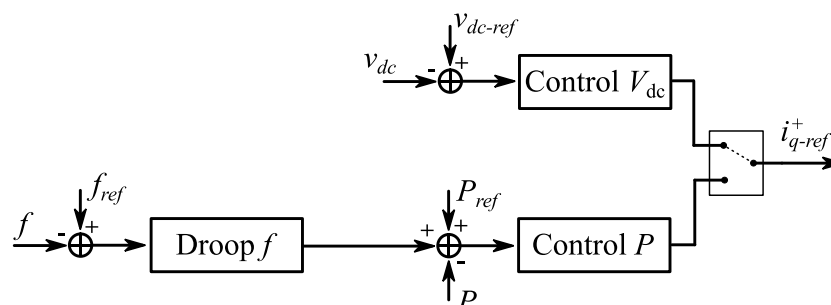


Figura 13. Controles externos para la referencia de corriente activa

- Referencia de corriente reactiva
 - Referencia de potencia reactiva en lazo abierto
 - Control de potencia reactiva
 - Droop de tensión AC
 - Control de inyección rápida de corriente reactiva

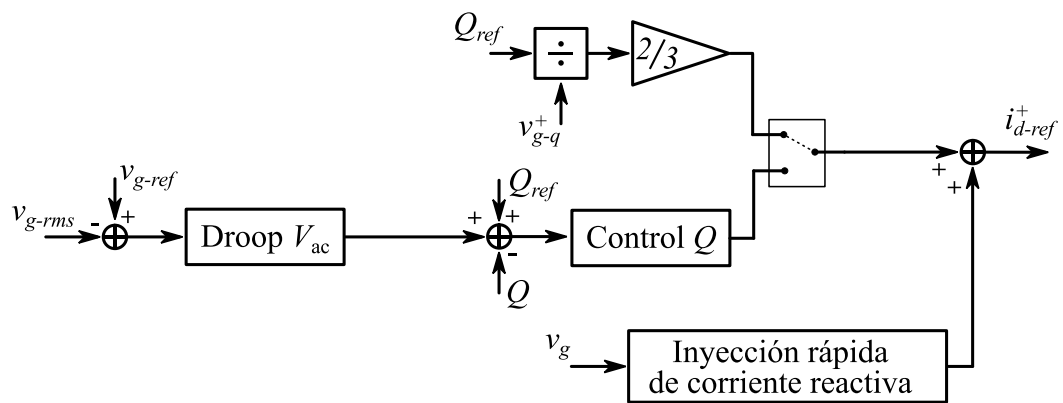


Figura 14. Controles externos para la referencia de corriente reactiva

Los controles anteriores se han implementado en PSCAD como se muestra en la Figura 15.

Current Reference Calculation

$i_{qd_p_ref}$
 $i_{qd_n_ref}$

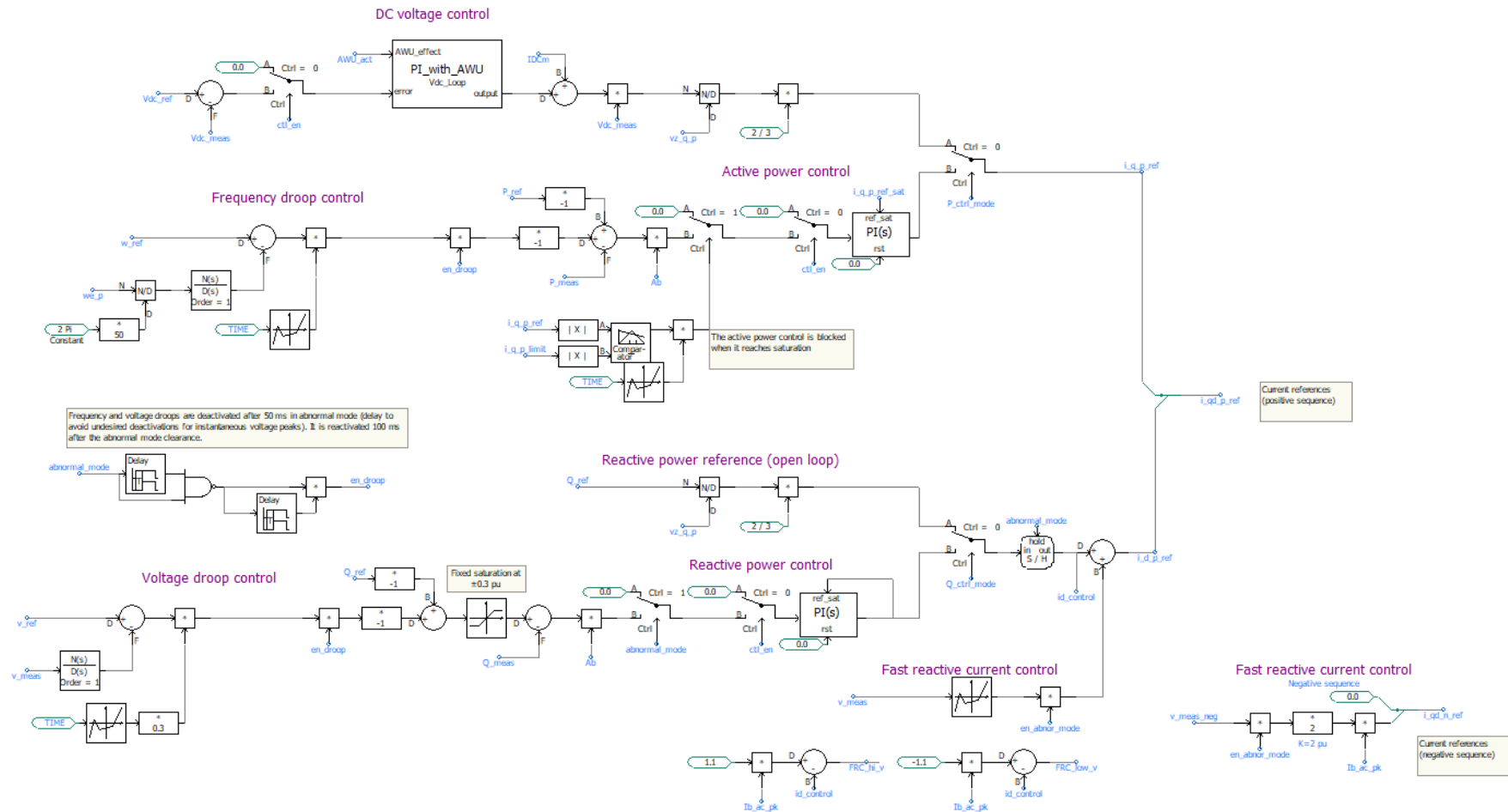


Figura 15. Lazos externos que generan las referencias de corriente

La selección del modo de control, por ejemplo control de tensión DC o control de potencia activa, y los parámetros se definen en el menú de parámetros del bloque general del convertidor, como se muestra en la Figura 16.

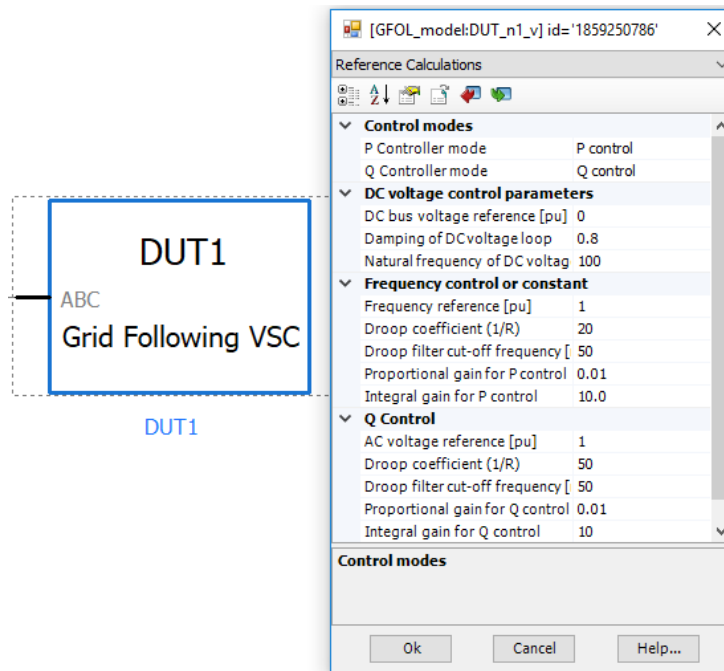


Figura 16. Selección del modo de control e introducción de parámetros

Los lazos de P y Q se han implementado con controladores PI, cuyos parámetros se han diseñado para tener una respuesta de primer orden con una constante de tiempo de aproximadamente 100 ms. Para ello se ha empleado el mismo método que para el lazo de corriente, el método IMC, dando lugar a las siguientes constantes:

$$k_{p-p} = \frac{\tau_c}{\tau_p} \quad k_{i-p} = \frac{1}{\tau_p} \quad (4)$$

$$k_{p-q} = \frac{\tau_c}{\tau_q} \quad k_{i-q} = \frac{1}{\tau_q} \quad (5)$$

Donde τ_p y τ_q son las constantes de tiempo del lazo de P y de Q respectivamente, asignándoles un valor de 100 ms. Este cálculo se realiza en el código de Python, introduciendo los valores de los parámetros de control directamente en el modelo de PSCAD. Los droops de frecuencia y de tensión requieren como parámetros de entrada la propia ganancia de droop y la frecuencia del filtro de la medida de frecuencia y tensión. La ganancia de droop está expresada en pu (respecto a la potencia aparente nominal [MVA]) y es la inversa del estatismo, de tal forma que, si se quiere un estatismo del 5%, se debe introducir una ganancia de $1/0,05 = 20$. Por defecto se han implementado un estatismo del 5% para la frecuencia y del 2% para la tensión.

2.3.4 Control para operación transitoria – inyección rápida de corriente reactiva

El control implementado en el modelo incluye un modo de operación transitoria que se detecta automáticamente según la tensión en el punto de conexión. En particular, esta operación transitoria tiene las siguientes características:

- Operación durante subtensiones ($V < 0,85$ pu)
 - Bloqueo de la PLL: la frecuencia de la PLL se bloquea a 50 Hz cuando la tensión se sitúa por debajo de 0,85 pu. Esta característica es útil durante faltas, cuando la frecuencia de la PLL puede sufrir transitorios abruptos.
- Operación durante subtensiones ($V < 0,85$ pu) y sobretensiones ($V > 1,1$ pu)

- Activación de la inyección rápida de corriente reactiva: el valor de la referencia de corriente depende de la desviación de tensión.
- Bloqueo de la referencia de corriente reactiva proveniente del control: para evitar interacciones con el control de inyección rápida de corriente reactiva.
- Bloqueo de los droops de frecuencia y tensión: si el evento de sub tensión o sobretensión excede los 50 ms, se bloquean los droops de frecuencia y tensión. Estos controles se rehabilitan de nuevo 100 ms después de que la tensión haya recuperado valores de operación normal.
- Cambio de prioridad de corriente: en operación normal, se prioriza la corriente activa (i_q), mientras que en operación transitoria la corriente reactiva (i_d) es preferente.

La Figura 17 muestra los bloques utilizados en PSCAD para la detección de la operación transitoria.

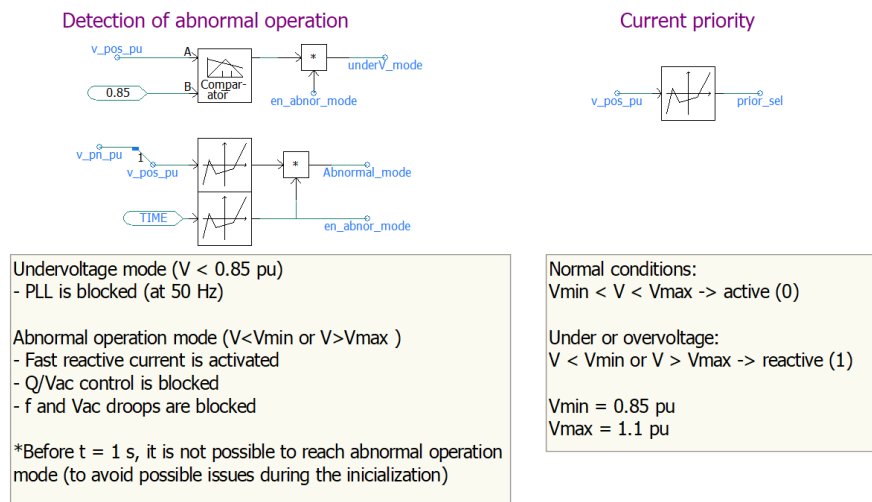
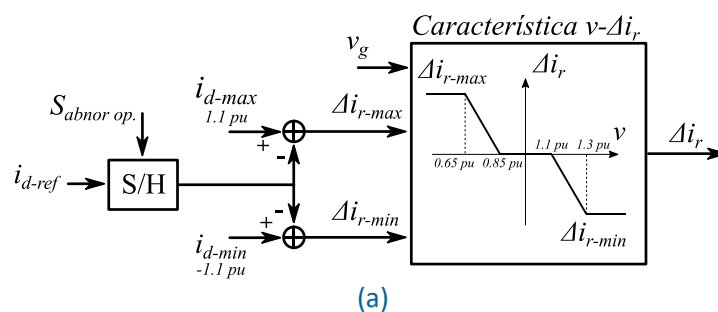


Figura 17. Detección de sub tensiones y sobretensiones en el POC y activación del control transitorio

Como se ha indicado anteriormente, la inyección rápida de corriente reactiva se activa cuando la tensión en el POC excede los límites establecidos ($V_{min} = 0,85 \text{ pu}$ y $V_{max} = 1,1 \text{ pu}$). Este control de corriente de falta se ha diseñado acorde con lo dispuesto en los artículos 7 (para la secuencia positiva) y 2.3 (para la secuencia negativa) de la Orden TED/749/2020 [4]. Se ha diseñado una característica $v - \Delta i_r$ que asegure que el convertidor inyecta o absorbe la corriente máxima (1,1 pu) cuando la tensión alcanza la tensión mínima (0,65 pu) o máxima (1,3 pu) establecida. Para ello se tiene en cuenta la referencia de corriente reactiva anterior a la activación del control rápido de reactiva, cuyo valor queda fijado durante la operación transitoria. De esta manera, los valores de Δi_{r-max} e Δi_{r-min} dependen del valor de i_{d-ref} en el momento de la activación del control transitorio. Por ejemplo, si en el momento de la activación del control transitorio $i_{d-ref} = 0,1 \text{ pu}$, entonces $\Delta i_{r-max} = 1,0 \text{ pu}$ y $\Delta i_{r-min} = -1,2 \text{ pu}$. Para la secuencia negativa, la inyección rápida de corriente reactiva es proporcional a la secuencia negativa de la tensión medida. La Figura 18 muestra el esquema del control de la secuencia positiva y la implementación en PSCAD.



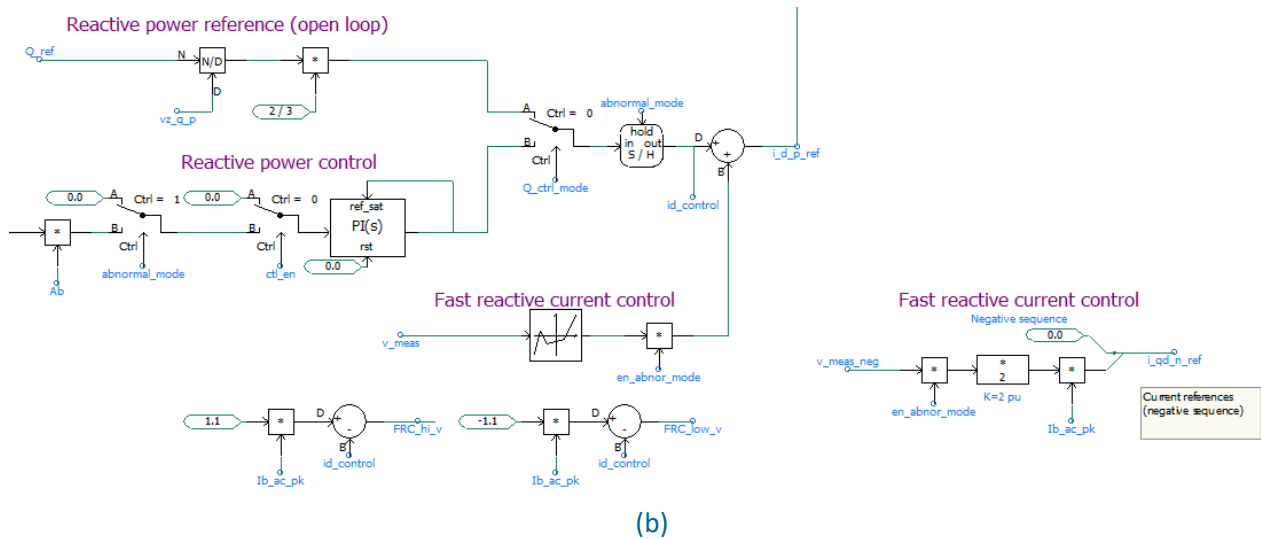


Figura 18. (a) Esquema control de inyección rápida de corriente reactiva; (b) su implementación en PSCAD

2.3.5 Limitación de corriente

El convertidor permite añadir una limitación de corriente. La activación o desactivación se realiza en el menú principal del convertidor, donde se indica si se quiere limitación o no, así como el límite máximo de corriente (valor RMS en kA), tal y como se muestra en Figura 19.

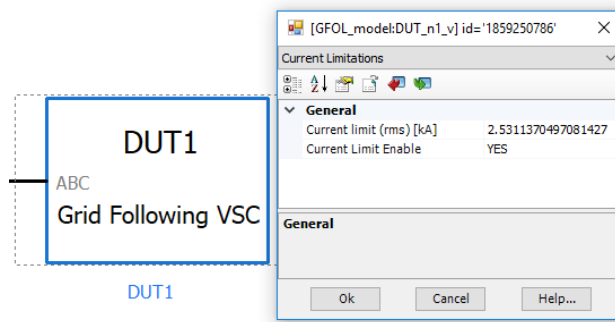


Figura 19. Activación de la limitación de corriente

Dependiendo del modo de operación del convertidor (normal o transitorio), los límites para cada componente de corriente son diferentes dependiendo de la prioridad. La regla de prioridad se ha implementado siguiendo el artículo 11 de la Orden TED/749/2020 [4]:

- Operación normal: la prioridad es la corriente activa i_q . Los límites se calculan como:

- Límite de corriente activa (i_q): es la corriente máxima.

$$i_{q-max} = i_{max}$$

- Límite de corriente reactiva positiva (i_d^+): se calcula teniendo en cuenta la corriente activa (i_q). Como cálculo conservador, se ha seleccionado el valor máximo entre la referencia i_{q-ref} y el valor real de la corriente i_q .

$$i_{d-max}^+ = \sqrt{i_{max}^2 - \max(i_{q-ref}, i_q)^2}$$

- Límite de corriente reactiva negativa (i_d^-): se calcula como la resta entre el límite de corriente reactiva positiva (i_{d-max}^+) menos el máximo entre la referencia de corriente reactiva positiva (i_{d-ref}^+) y la corriente reactiva positiva real (i_d^+).

$$i_{d-max}^- = i_{d-max}^+ - \max(i_{d-ref}^+, i_d^+)$$

- Operación transitoria: la prioridad es la corriente reactiva i_d .

- Límite de corriente reactiva positiva (i_d^+):

$$i_{d-max} = i_{max}$$

2. Límite de corriente reactiva negativa (i_d^-):

$$i_{d-max}^- = i_{max} - \max(i_{d-ref}^+, i_d^+)$$

3. Límite de corriente activa (i_q):

$$i_{q-max} = \sqrt{i_{max}^2 - (\max(i_{d-ref}^+, i_d^+) + \max(i_{d-ref}^-, i_d^-))^2}$$

Los límites anteriores se han implementado en PSCAD como se muestra en la Figura 20.

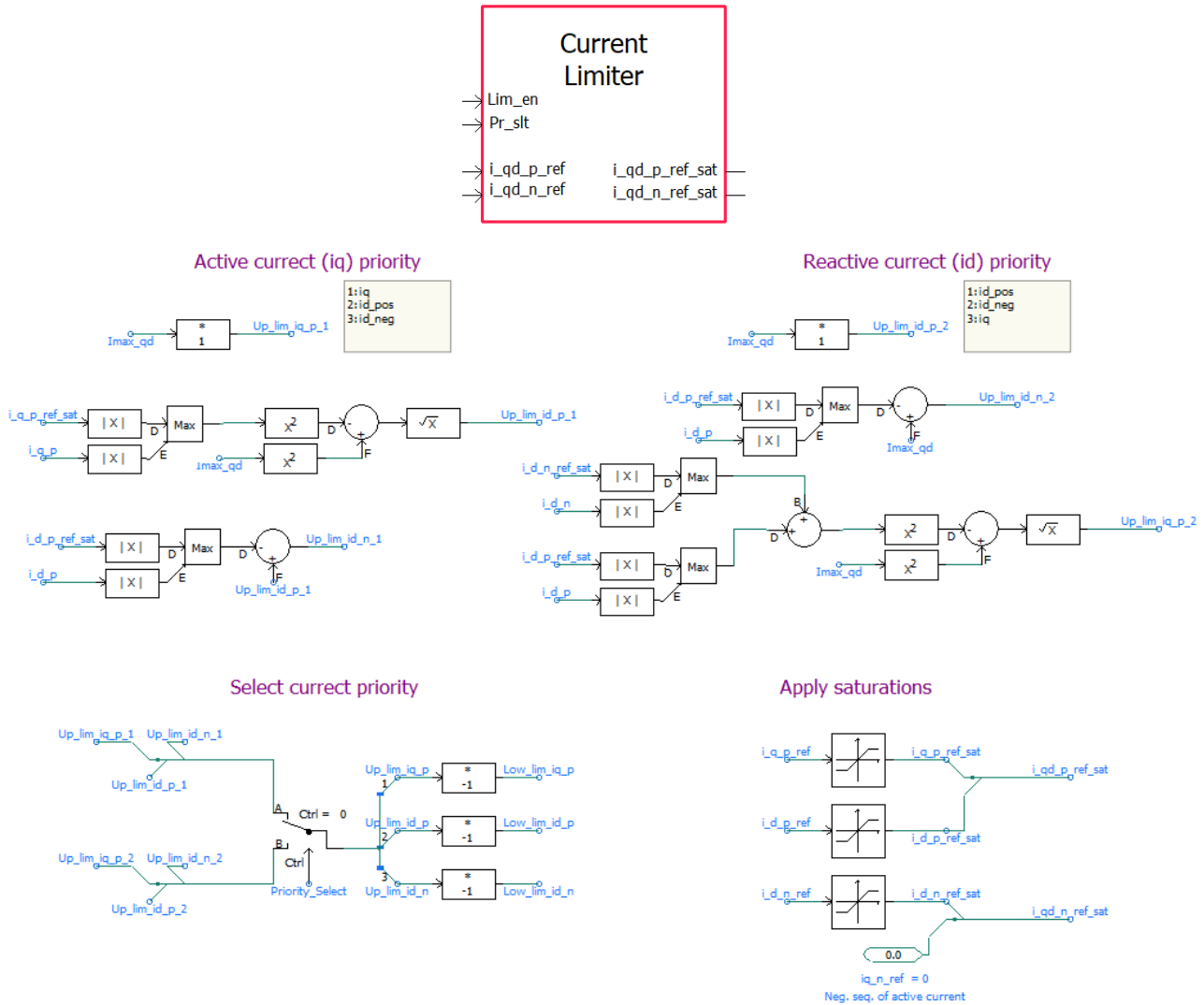


Figura 20. Limitación de corriente implementada en PSCAD

3 Descripción de las simulaciones

Las simulaciones se han ejecutado utilizando la versión de PSCAD 4.6.2 y el compilador “Intel Visual Fortran Compiler XE 14.0.1.139”. Los modelos se deberían poder ejecutar con cualquier otra versión de compilador, ya que sólo se han empleado modelos de la librería original de PSCAD.

Se han simulado los escenarios básicos para comprobar el funcionamiento del control *grid-following* descrito en la Sección 2.3. Las principales características de cada uno de estos escenarios se muestran en la Tabla 1.

Tabla 1. Descripción de los escenarios simulados

Escenario	Eventos simulados	Pref	Qref
1. Sin perturbaciones	-	0,5 pu	0,1 pu
2. Cambio de Pref	Pref: de 1 pu a 0 en escalones de -0,2 pu	Variable	0 pu
3. Cambios de Qref	Qref: cambios $\pm 0,3$ pu	0 pu	Variable
4. Cambio de ángulo de la red	Cambios de $\pm 20^\circ$	0,5 pu	0,1 pu
5. Cambio de la frecuencia de la red	Rampa ± 4 Hz/s desde 50 Hz a 51/49 Hz	0,5 pu	0,1 pu
6. Cambio de tensión de la red	Cambios de $\pm 0,1$ pu	0,5 pu	0,1 pu
7. Faltas simétricas	V = 0 pu durante 100 ms V = 0,7 pu durante 500 ms	0,5 pu	0,1 pu
8. Faltas asimétricas	V = 0 pu durante 100 ms V = 0,7 pu durante 500 ms	0,5 pu	0,1 pu
9. Cambio de SCR	Desde 3 a 0,5 (apertura S2)	0,5 pu	0,1 pu
10. Desconexión de la red	Apertura de S1 y S2	0,5 pu	0,1 pu

3.1 Código de Python para ejecutar el modelo

El modelo *GFOL_model_v2.pscx* descrito en las secciones anteriores se debe ejecutar mediante el código incluido en el archivo de Python *pscad_run_gfol_v2.py*, ya que define los parámetros a utilizar así como el escenario a simular. Si no se utiliza dicho código, únicamente se podría simular un solo escenario (el que se haya ejecutado y guardado por última vez) o cambiando manualmente los parámetros adecuados. Este código está dividido en varias partes:

1. Importar librerías
2. Seleccionar versión de PSCAD y de compilador
3. Seleccionar proyecto

```
# run the PSCAD model with python
# project: GFOL_model_v2
# test cases updated
# Author: CITCEA-UPC
# Date: March 2021

import os
import mhrc.automation
from mhrc.automation.utilities.file import File
import shutil
from math import pi, sqrt

#####
##### Parameters to be updated by user #####
#####

# PSCAD and project
pscad_version = 'PSCAD 4.6.2 (x64)'
fortran_version = 'Intel 14.0.139'
fortran_ext = '.if12'
project_name = 'GFOL_model_v2'
```



4. Parámetros de la simulación

```
# simulation setup
sim_duration = 10 # in seconds
sim_step = 10 # in micro seconds

# test setup
test_case = 2 # 1: Normal operation, 2: P-ref variation, 3: Q-ref variation,
              # 4: Grid angle variation, 5: Grid frequency variation, 6: Grid voltage variation,
              # 7: Symmetrical fault, 8: Asymmetrical fault,
              # 9: Change of grid SCR, 10: Grid disconnection
```

5. Definir parámetros de cada bloque, por ejemplo, del control del convertidor

```
#-----
# CONVERTER CONTROL
#-----
#PLL
ts=0.1 # Settling time
xi_pll = 0.707 # Damping
wn_pll = 4/(ts*xi_pll)
kp_pll = 2*wn_pll*xi_pll/(base_ACvoltage*sqrt(2/3))
tau_pll = 2*xi_pll/wn_pll
ki_pll = kp_pll/tau_pll

# P controller mode
P_ctrl_mode = 1 # 1 = P control (+frequency control, depending on mp value)
              # 0 = DC voltage control

# Q controller mode
Q_ctrl_mode = 1 # 1 = Q control (+voltage control, depending on mq value)
              # 0 = open loop (direct Q reference)

# current controller (kp and ki are calculated in the PSCAD model according to the parameters defined here)
tau_c = 0.001 # time constant

# Current saturation
Imax = 1.1 #pu
en_current_saturation = 1 # 1 = saturation enabled
                      # 0 = saturation disabled

# Active power control
tau_P = 0.1 # s
kp_P = tau_c/tau_P
ki_P = 1/tau_P

# Reactive power control
tau_Q = 0.1 # s
kp_Q = tau_c/tau_Q
ki_Q = 1/tau_Q

# Frequency droop control
mp = 20 # mp = 1/Rp (Rp=5%)
w_filtP = 50 # Hz

# Voltage droop control
mq = 50 # mq = 1/Rq (Rq=2%)
w_filtQ = 50 # Hz
```

6. A continuación, se encuentran las clases creadas que definen todos los parámetros de la red, del convertidor y de la carga que se modifican en alguno de los escenarios. Ejemplos de estos parámetros son el ángulo o la frecuencia de la red, la potencia activa o reactiva del convertidor o la potencia de la carga. Estos permiten ejecutar cambios en escalón en el tiempo y los valores especificados. Por ejemplo: si se indica que $t_phi1 = 2$, se producirá un cambio del ángulo de la red en 2 segundos. Puesto que es el primer escalón, el valor del ángulo cambiará desde phi_0 (valor inicial) a phi_1 . Este valor phi_1 se mantendrá hasta t_phi2 . Por defecto se ha indicado que todos cambios se produzcan en $t = 100$ s, lo que significa que no se produce ningún cambio en la simulación, ya que su duración es de 10 s. Esto correspondería con el escenario 1.

```
#-----  
# VARIABLES TO SET DIFFERENT SCENARIOS  
#-----  
class param_grid:  
    tphi_1 = 100 # Changes for grid angle, time, in [s]  
    tphi_2 = 100  
    tphi_3 = 100  
    tphi_4 = 100  
    phi_0 = 0 # Changes for grid angle, angle, in [°]  
    phi_1 = 0  
    phi_2 = 0  
    phi_3 = 0  
    phi_4 = 0
```

```
class param_vsc:  
    tP_1 = 100 # Changes for P reference, time, in [s]  
    tP_2 = 100  
    tP_3 = 100  
    tP_4 = 100  
    tP_5 = 100  
    Pref_0 = 0.5 # Changes for P reference, P, in [pu]  
    Pref_1 = 0.5  
    Pref_2 = 0.5  
    Pref_3 = 0.5  
    Pref_4 = 0.5  
    Pref_5 = 0.5
```

```
class param_load:  
    tload_1 = 0.1 #Changes for load, time, in [s]  
    tload_2 = 100  
    tload_3 = 100  
    tload_4 = 100  
    Pload_0 = 0 # initial load in [pu]  
    Pload_1 = 0.25  
    Pload_2 = 0  
    Pload_3 = 0  
    Pload_4 = 0 # final load in [pu]
```

7. Tras la definición de las clases, se encuentra la definición de las funciones que modifican los parámetros para cada escenario. Por ejemplo: en el escenario 2 se modifica la referencia de P del convertidor.

```
def test_case_2(param_vsc): #Pref changes  
    param_vsc.tP_1 = 2  
    param_vsc.tP_2 = 3.5  
    param_vsc.tP_3 = 5  
    param_vsc.tP_4 = 6.5  
    param_vsc.tP_5 = 8  
    param_vsc.Pref_0 = 1  
    param_vsc.Pref_1 = 0.8  
    param_vsc.Pref_2 = 0.6  
    param_vsc.Pref_3 = 0.4  
    param_vsc.Pref_4 = 0.2  
    param_vsc.Pref_5 = 0  
    param_vsc.Qref_0 = 0  
    return param_vsc
```

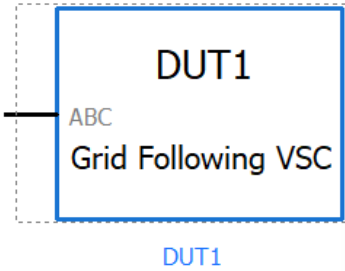


8. Definir la variable con la que se quiere iterar y los valores

```
##### define the name of the varying parameter #####  
  
varying_par_name = 'test_case' # name of the parameter  
varying_par_values = [2,3,4,5,6,7,8] # desired values of the parameter
```

9. Definir los Ids de los componentes que se quieren modificar

```
# Component IDs  
id_dut1 = 1859250786  
id_grid = 34364000  
id_load = 76094606
```



Attributes

General

Class	UserCmp
Id	1859250786
Name	GFOL_model:DUT_n1_v
Definition	GFOL_model:DUT_n1_v
Build Label	

Schematic

Ok Cancel Help...

10. Lanzar PSCAD, cargar el proyecto y seleccionar los parámetros para el modelo. En función del escenario seleccionado (con la variable test_case), se ejecuta la función adecuada para modificar los parámetros.

```
# Launch specific PSCAD and Fortran version  
pscad = mhrc.automation.launch_pscad(pscad_version=pscad_version, fortran_version=fortran_version,minimize=True)  
  
if pscad:  
    try:  
        # Load the project  
        working_dir = os.getcwd() + "\\ "  
        pscad.load([working_dir + project_name + ".pscx"])  
        project = pscad.project(project_name)  
        project.focus()  
        all_sim_names = []  
  
        for value in varying_par_values:  
  
            globals()[varying_par_name] = value  
            sim_name = varying_par_name + '_' + str(value)  
            all_sim_names.append(sim_name)  
  
            default_param(param_grid, param_vsc, param_load)  
            if test_case == 2:  
                test_case_2(param_vsc)  
            elif test_case == 3:  
                test_case_3(param_vsc)  
            elif test_case == 4:  
                test_case_4(param_grid)  
            elif test_case == 5:  
                test_case_5(param_grid)  
            elif test_case == 6:  
                test_case_6(param_grid)  
            elif test_case == 7:  
                test_case_7(param_grid)  
            elif test_case == 8:  
                test_case_8(param_grid)  
            elif test_case == 9:  
                test_case_9(param_grid)  
            elif test_case == 10:  
                test_case_10(param_grid)  
            project.set_parameters(time_duration=sim_duration, time_step=sim_step)
```


11. Actualizar el modelo con los parámetros seleccionados y ejecutar el modelo.

```
# Get the "Main" canvas
main = project.user_canvas('Main')

# Get components
dut1 = main.user_cmp(id_dut1)
grid = main.user_cmp(id_grid)
load = main.user_cmp(id_load)

# parameters update
dut1.set_parameters(f_l=base_freq, Ab=base_power, Vb_ac=base_ACvoltage, Vb_dc=base_DCvoltage,
    tau_c_loop=tau_c, R_cf=filter_res, L_cf=filter_induct,
    I_max_rms = I_max*base_current, limit_act = en_current_saturation,
    P_ctrl_mode = P_ctrl_mode, Q_ctrl_mode = Q_ctrl_mode,
    kp_activepower_PI = kp_P, ki_activepower_PI = ki_P,
    mp_droop = mp, wp_droop = w_filtP, mq_droop = mq, wq_droop = w_filtQ,
    tP1=param_vsc.tP_1, tP2=param_vsc.tP_2, tP3=param_vsc.tP_3, tP4=param_vsc.tP_4, tP5=param_vsc.tP_5,
    Pref0=param_vsc.Pref_0, Pref1=param_vsc.Pref_1, Pref2=param_vsc.Pref_2, Pref3=param_vsc.Pref_3, Pref4=param_vsc.Pref_4, Pref5=param_vsc.Pref_5,
    tQ1=param_vsc.tQ_1, tQ2=param_vsc.tQ_2, tQ3=param_vsc.tQ_3, tQ4=param_vsc.tQ_4,
    Qref0=param_vsc.Qref_0, Qref1=param_vsc.Qref_1, Qref2=param_vsc.Qref_2, Qref3=param_vsc.Qref_3, Qref4=param_vsc.Qref_4)

grid.set_parameters(L1=z1_induct, R1=z1_res, L2=z2_induct, R2=z2_res,
    tphi_1=param_grid.tphi_1, tphi_2=param_grid.tphi_2, tphi_3=param_grid.tphi_3, tphi_4=param_grid.tphi_4,
    phi_0=param_grid.phi_0, phi_1=param_grid.phi_1, phi_2=param_grid.phi_2, phi_3=param_grid.phi_3, phi_4=param_grid.phi_4,
    tf_1=param_grid.tf_1, tf_2=param_grid.tf_2, tf_3=param_grid.tf_3, tf_4=param_grid.tf_4,
    f_0=param_grid.f_0, f_1=param_grid.f_1, f_2=param_grid.f_2, f_3=param_grid.f_3, f_4=param_grid.f_4,
    tu_1=param_grid.tu_1, tu_2=param_grid.tu_2, tu_3=param_grid.tu_3, tu_4=param_grid.tu_4, Vb_ac=base_ACvoltage,
    Ull_0=param_grid.Ull_0, Ull_1=param_grid.Ull_1, Ull_2=param_grid.Ull_2, Ull_3=param_grid.Ull_3, Ull_4=param_grid.Ull_4,
    ts_1=param_grid.ts_1, ts_2=param_grid.ts_2,
    tf_3p_1=param_grid.tf_3p_1, df_3p_1=param_grid.df_3p_1, rf_3p_1=param_grid.rf_3p_1,
    tf_3p_2=param_grid.tf_3p_2, df_3p_2=param_grid.df_3p_2, rf_3p_2=param_grid.rf_3p_2,
    tf_lp_1=param_grid.tf_lp_1, df_lp_1=param_grid.df_lp_1, rf_lp_1=param_grid.rf_lp_1,
    tf_lp_2=param_grid.tf_lp_2, df_lp_2=param_grid.df_lp_2, rf_lp_2=param_grid.rf_lp_2)

load.set_parameters(tload_1=param_load.tload_1, tload_2=param_load.tload_2, tload_3=param_load.tload_3, tload_4=param_load.tload_4,
    Pload_0=param_load.Pload_0, Pload_1=param_load.Pload_1, Pload_2=param_load.Pload_2, Pload_3=param_load.Pload_3, Pload_4=param_load.Pload_4)

# run model
project.run()
```

12. Guardar resultados: en la carpeta output se crean directorios con el nombre *variable_modificada_valor*, donde se guardan los resultados en formato .txt.

» GFOL_model »

Nombre

- GFOL_model.if12
- output_files
- GFOL_model.bakx
- GFOL_model.pscx
- GFOL_model.psmx
- pscad_run_gfol.py
- save_txt_from_out.py

» GFOL_model » output_files »

Nombre

- test_case_1
- test_case_2
- test_case_3
- test_case_4
- test_case_5
- test_case_6
- test_case_7
- test_case_9
- test_case_10

» GFOL_model » output_files » test_case_2

Nombre

- sim_results.inf
- sim_results_01.txt
- sim_results_02.txt

4 Resultados de los escenarios 1-10

En las siguientes secciones se muestran los resultados obtenidos para los diferentes escenarios considerados para el modelo *grid-following*.

4.1 Escenario 1: sin perturbaciones

Este primer escenario permite evaluar el funcionamiento del modelo de *grid-following* y comprobar que el sistema es estable. Como se ha indicado anteriormente, las impedancias del equivalente de la red se han dimensionado para tener un SCR de 3. La carga consume una carga puramente resistiva iguala 0,25 pu, mientras que la referencia de potencia del convertidor es de 0,5 pu. La red absorbe la diferencia entre la potencia inyectada por el convertidor y la consumida por la carga, es decir, 0,25 pu. Estos resultados se observan en la Figura 21.

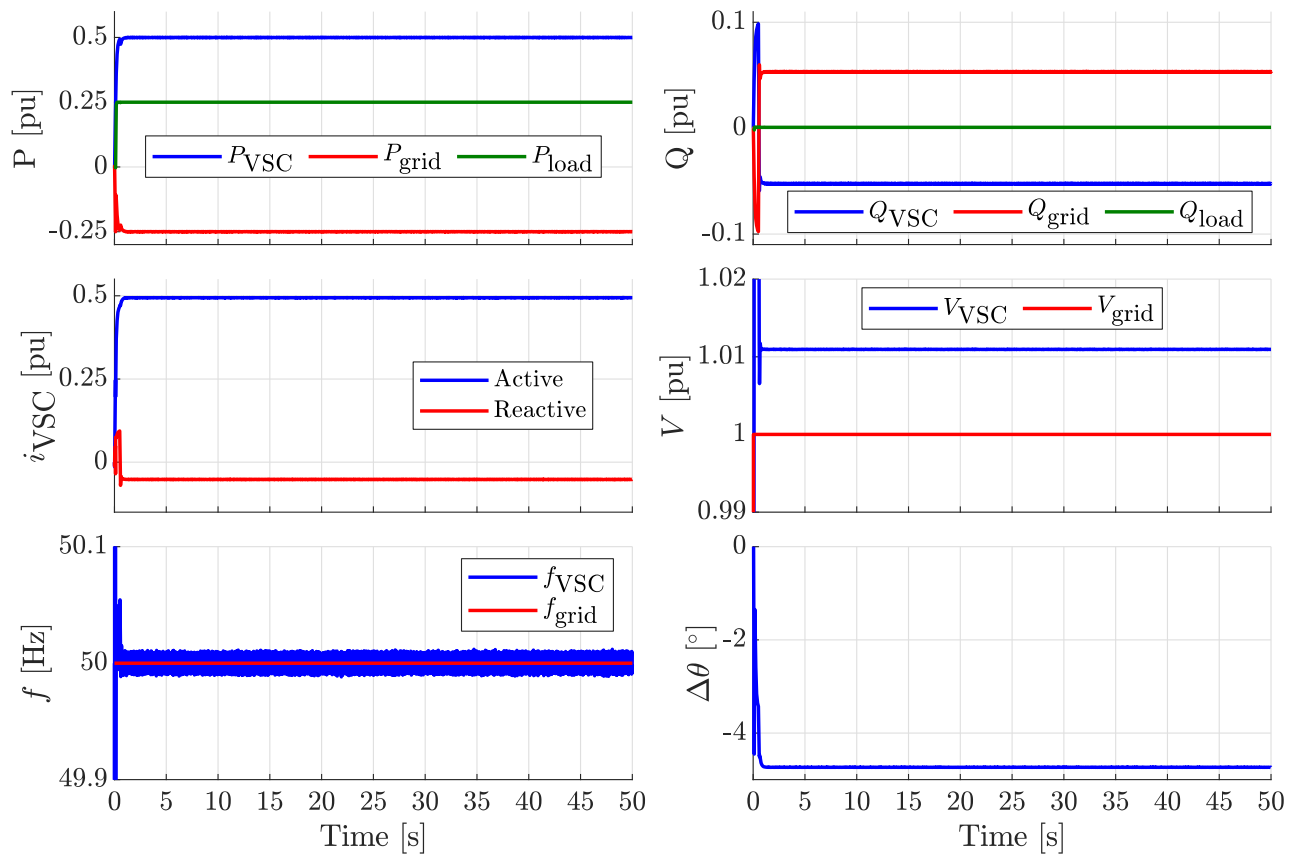


Figura 21. Escenario 1: sin perturbaciones

La simulación alcanza el régimen estacionario aproximadamente tras 1 segundo de simulación. Esto se observa en la Figura 22, que muestra en detalle el primer segundo de la simulación. Puesto que el modelo no se ha inicializado, las corrientes y tensiones son cero al inicio de la simulación. El control del convertidor se habilita en $t = 0,001$ s, cuando empieza a inyectar corriente siguiendo las referencias de potencia activa (0,5 pu) y reactiva (0,1 pu). En $t = 0,1$ s, se realiza un cambio de demanda, que varía desde 0 pu hasta 0,25 pu. Al tratarse de un cambio bastante abrupto, afecta a otras magnitudes del sistema, como la tensión. Tras aproximadamente 500 ms, tanto la P como la Q del convertidor alcanzan los valores de sus referencias. En ese instante, en $t = 0,5$ s, se activan los droops de frecuencia (5%) y tensión (2%). **Puesto que la frecuencia del sistema es 50 Hz, la activación del droop de frecuencia no tiene un gran efecto.** Sin embargo, la activación del droop de tensión modifica considerablemente la referencia de potencia reactiva, ya que la desviación de tensión del convertidor previa a la activación del droop es bastante elevada ($V_{VSC} > 1,05$ pu). Entonces, con la activación del droop de tensión se consigue reducir la desviación de tensión en bornes del convertidor mediante la variación de su potencia reactiva.

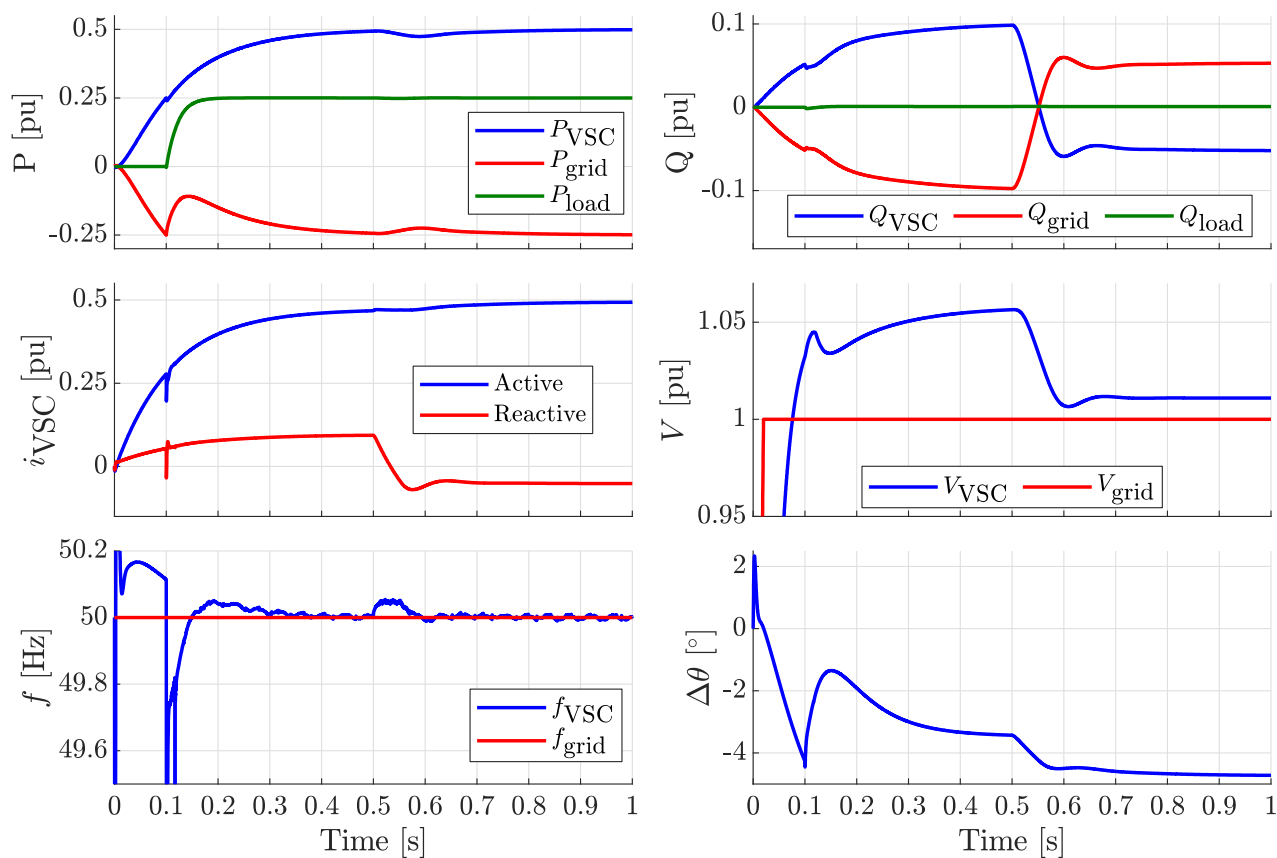


Figura 22. Inicialización del modelo *grid-following*

4.2 Escenario 2: cambio de Pref

Este escenario supone cambios de referencia de potencia activa del convertidor. Esta referencia se inicializa en 1 pu. Puesto que la carga es de 0,25 pu, la red absorbe 0,75 pu. En $t = 2$ s, se produce el primer cambio de referencia de potencia activa del convertidor, desde 1 pu hasta 0,8 pu. El resto de escalones de -0,2 pu se ejecutan cada 1,5 s. El perfil de la referencia de potencia activa se muestra en la Figura 23. El convertidor es capaz de seguir estos cambios en la referencia con una respuesta de primer orden, tal y como se muestra en la Figura 24. Conforme se va reduciendo la potencia intercambiada entre la red y el convertidor, la tensión del convertidor y la diferencia de ángulo $\Delta\theta$ entre ellos se ven modificadas. Como consecuencia del cambio de tensión, el convertidor también varía su potencia reactiva, ya que el *droop* de tensión está activo. El signo de $\Delta\theta$ se invierte cuando la red deja de absorber potencia y pasa a inyectarla, en $t = 6,5$ s aproximadamente. En ese instante, la referencia de potencia del convertidor es de 0,2 pu, mientras que la carga se mantiene en 0,25 pu, por lo que la red debe proveer 0,05 pu.

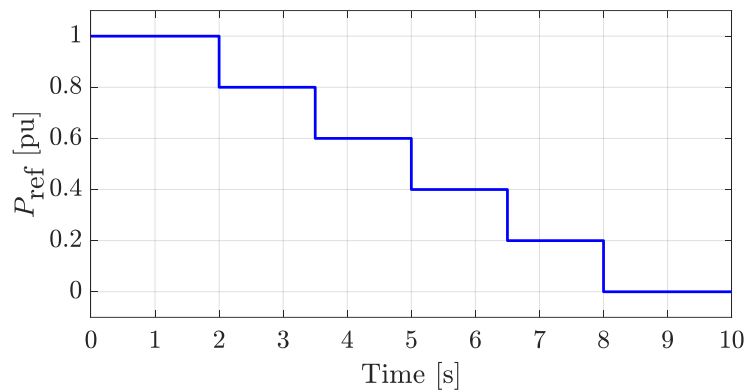


Figura 23. Perfil de potencia activa del escenario 2

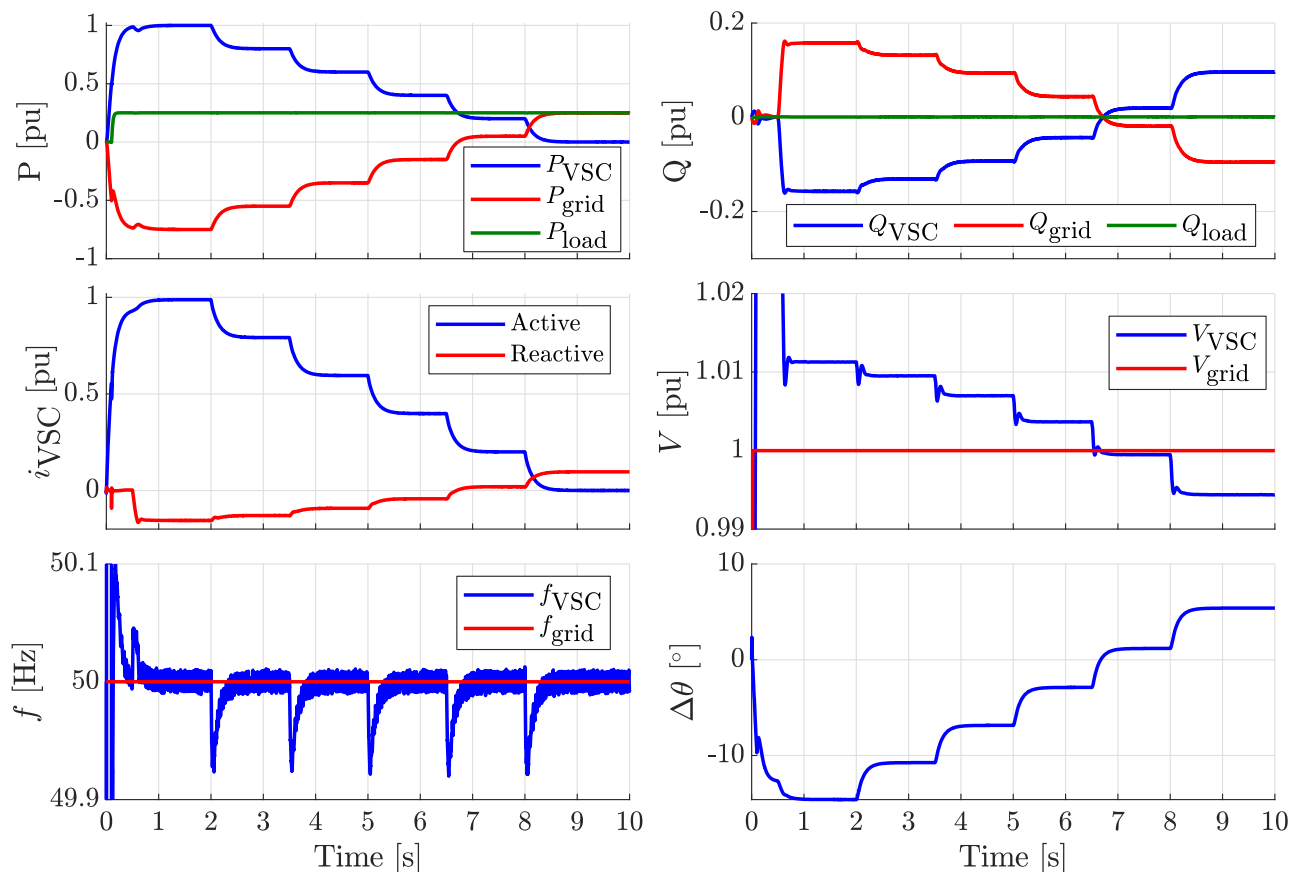


Figura 24. Escenario 2: cambio de referencia de Pref

4.3 Escenario 3: cambio de Q_{ref}

La referencia de potencia reactiva del convertidor es inicialmente 0 pu, como se muestra en la Figura 25. En los resultados de la Figura 26 se observa como la potencia reactiva inyectada por el convertidor no sigue la referencia, ya que el *droop* de tensión modifica dicha referencia, como se indicó en el escenario 1.

La Figura 25 muestra los cambios de referencia de potencia reactiva implementados en este escenario. En $t = 2$ s, la referencia de potencia reactiva se incrementa hasta 0,3 pu, dando lugar a un incremento de la tensión. En $t = 4$ s, se reduce la referencia de nuevo a 0 pu, por lo que el sistema vuelve al punto de operación inicial. En $t = 4$ s, se implementa un cambio de -0,3 pu, obteniéndose resultados similares. Los cambios de referencia de potencia reactiva no afectan considerablemente a la potencia activa inyectada por el convertidor.

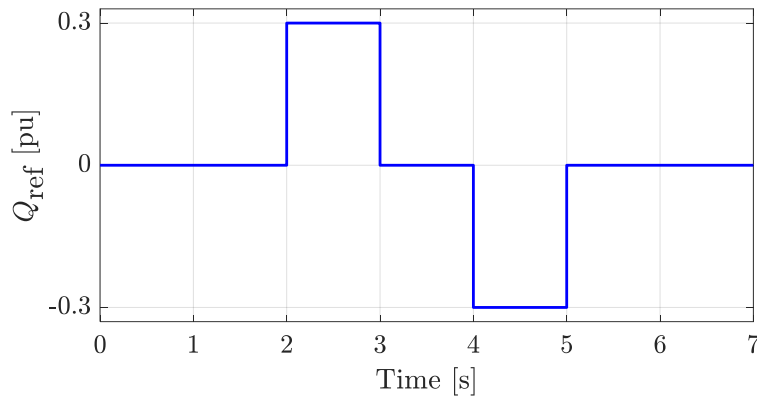


Figura 25. Perfil de referencia de potencia reactiva del escenario 3

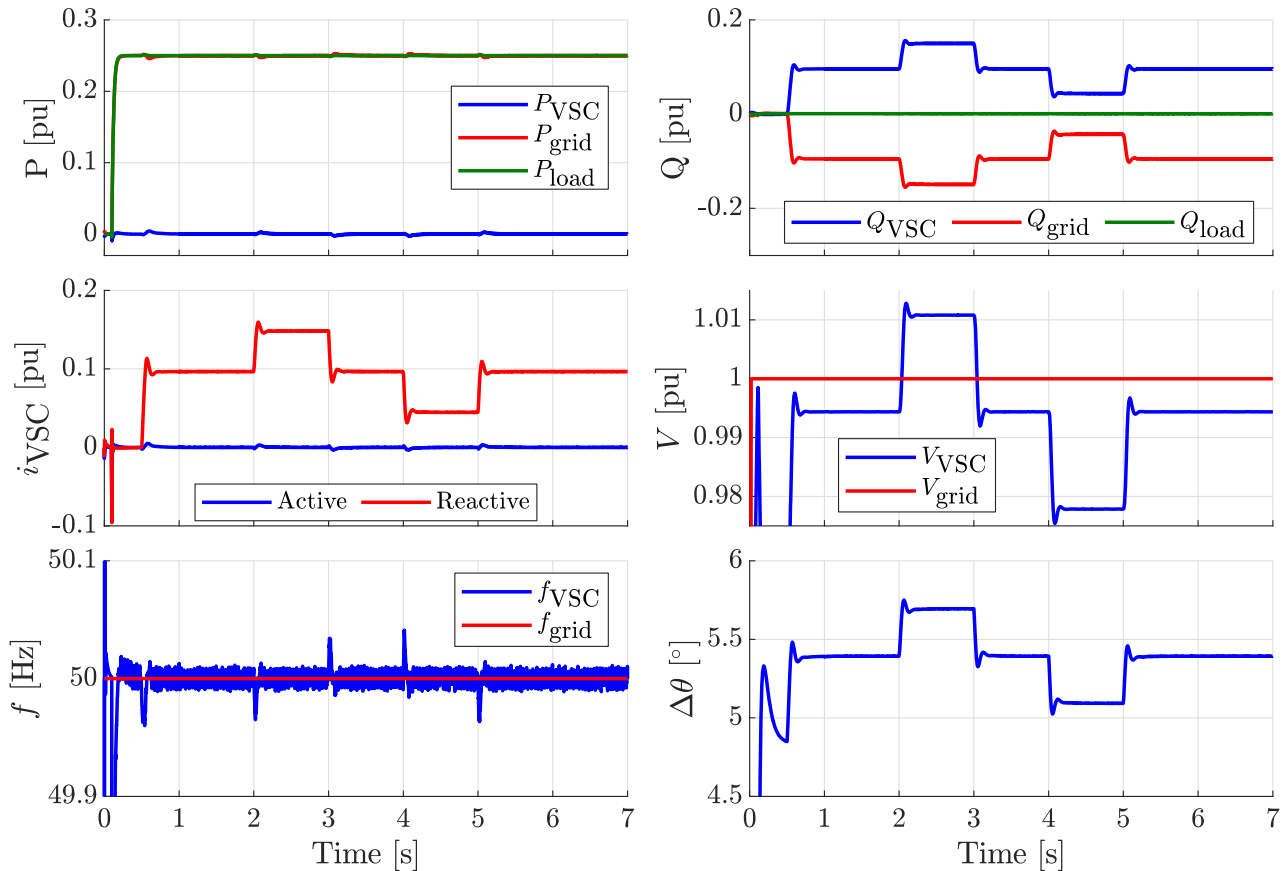


Figura 26. Escenario 3: cambio de Q_{ref}

4.4 Escenario 4: cambio de ángulo de la red

En este escenario se han implementado cambios de ángulo de la red de $\pm 20^\circ$ de la Figura 27. Todos los cambios tienen efectos similares en el sistema, como se muestra en la Figura 28. Cada cambio de ángulo provoca una desincronización transitoria del convertidor, provocando un cambio del flujo de potencia activa y reactiva. La PLL corrige rápidamente la frecuencia interna del convertidor, mostrada en la Figura 28, permitiendo al convertidor sincronizarse de nuevo con la red.

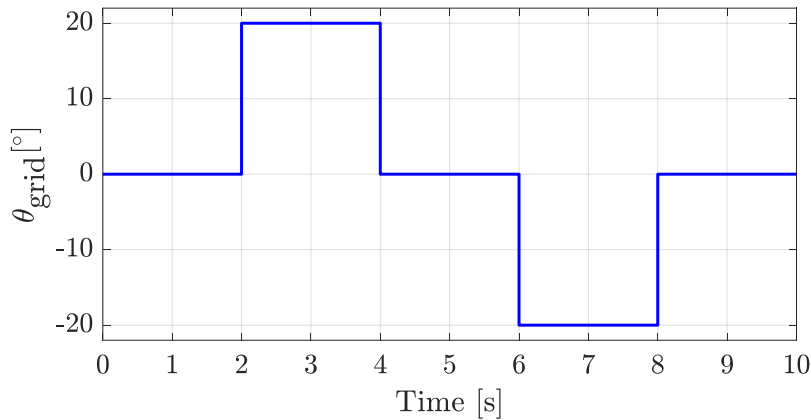


Figura 27. Cambios de ángulo de la red implementados en el escenario 4

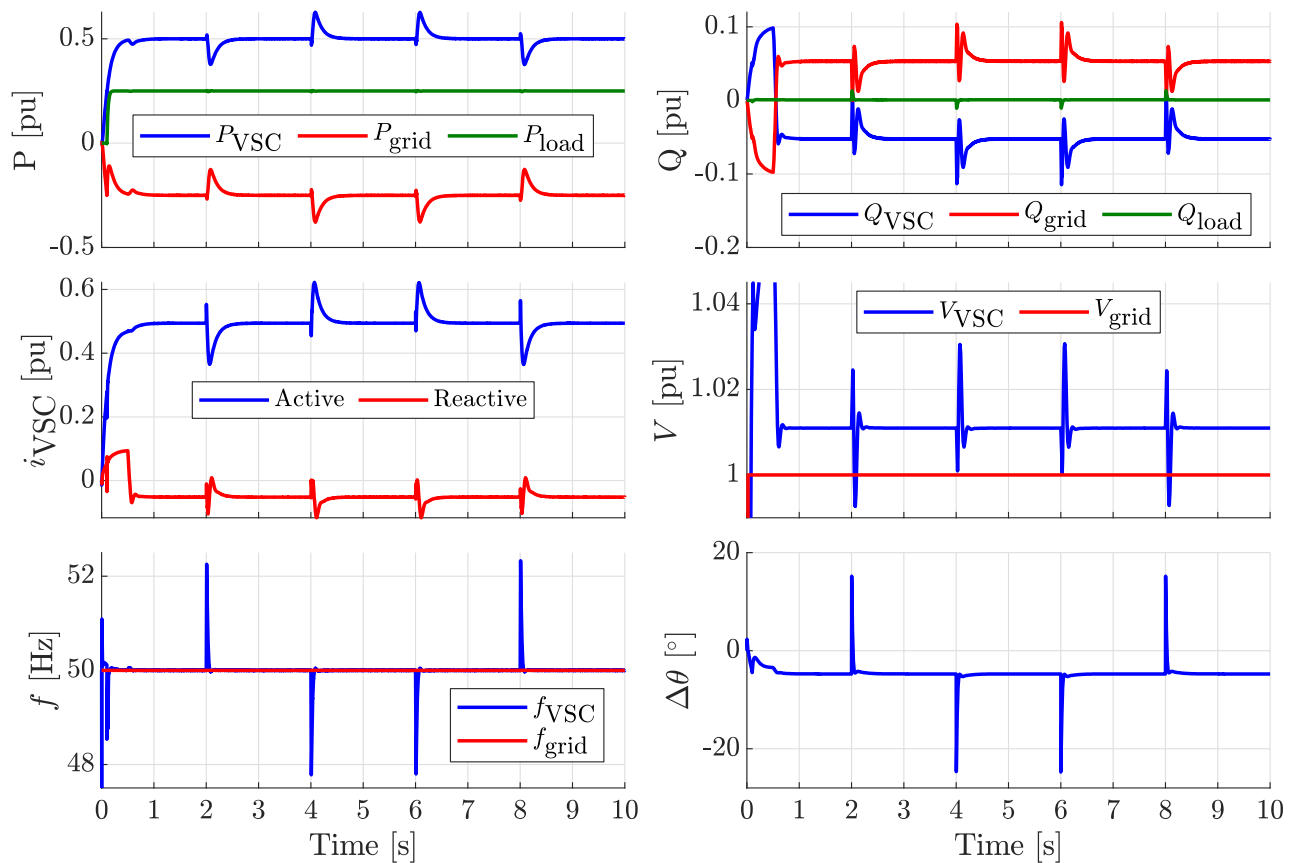


Figura 28. Escenario 4: cambio de ángulo de la red

4.5 Escenario 5: cambio de frecuencia de la red

En este escenario se han implementado cambios de frecuencia de ± 1 Hz con una rampa de ± 4 Hz/s, como se muestra en la Figura 29. Al inicio de la simulación, el convertidor inyecta 0,5 pu de potencia activa. Cuando se modifica la frecuencia de la red, la PLL es capaz de medir esta variación, por lo que el VSC reduce su potencia generada de acuerdo con el *droop* implementado (5%). Esta reducción de potencia activa es de -0,4 pu, desde 0,5 pu iniciales hasta 0,1 pu a 51 Hz. Cuando la frecuencia de la red vuelve a 50 Hz, el convertidor vuelve a inyectar 0,5 pu de potencia activa.

En $t = 6$ s, se reduce la frecuencia a 49 Hz. Esto provoca un aumento de la potencia del convertidor de 0,4 pu, pasando a inyectar 0,9 pu. Por tanto, el convertidor es capaz de dar soporte a la frecuencia gracias a la medida de la frecuencia de la PLL y del *droop* de frecuencia, que modifica la referencia de potencia activa.

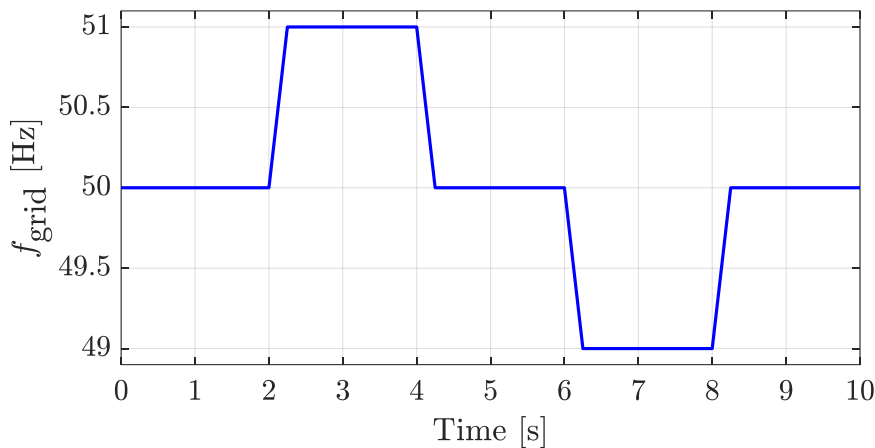


Figura 29. Cambios de frecuencia de la red implementados en el escenario 5

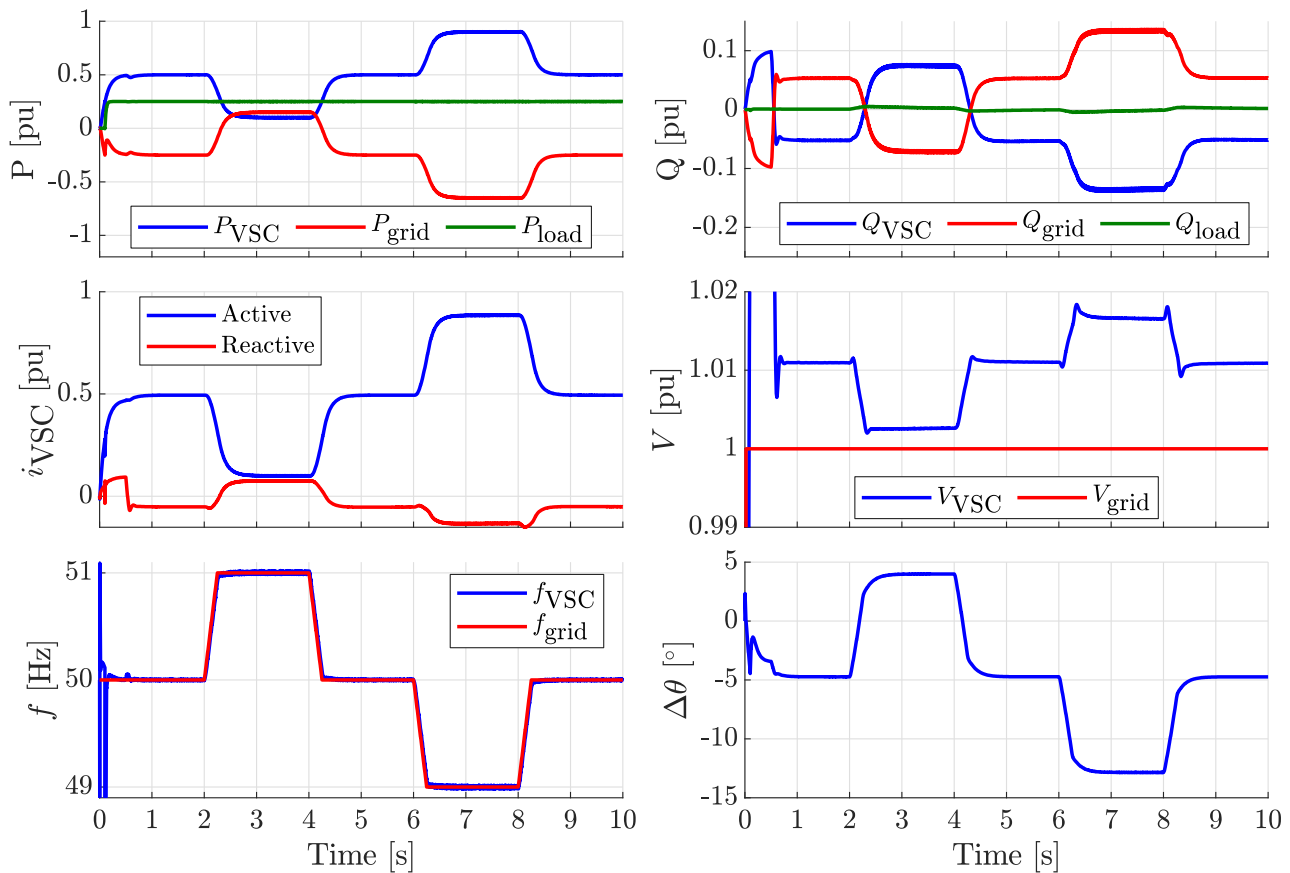


Figura 30. Escenario 5: cambio de frecuencia de la red

4.6 Escenario 6: cambio de tensión de la red

Los cambios de tensión de la red implementados en el escenario 6 se muestran en la Figura 31. Estas variaciones de tensión de la red provocan principalmente cambios en la reactiva intercambiada por el convertidor, como se muestra en la Figura 32. Los cambios de tensión de la red provocan grandes cambios en la tensión del convertidor, que es capaz de reducir la desviación de tensión modificando su inyección de potencia reactiva gracias al *droop* de tensión.

Debido a los cambios abruptos de tensión, la frecuencia de la PLL también se ve afectada, ya que la tensión es la variable de entrada de este lazo de control.

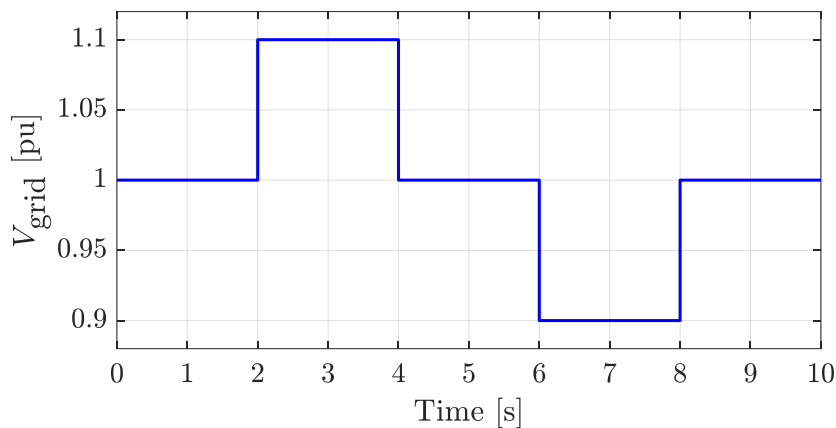


Figura 31. Cambios de tensión de la red implementados en el escenario 6

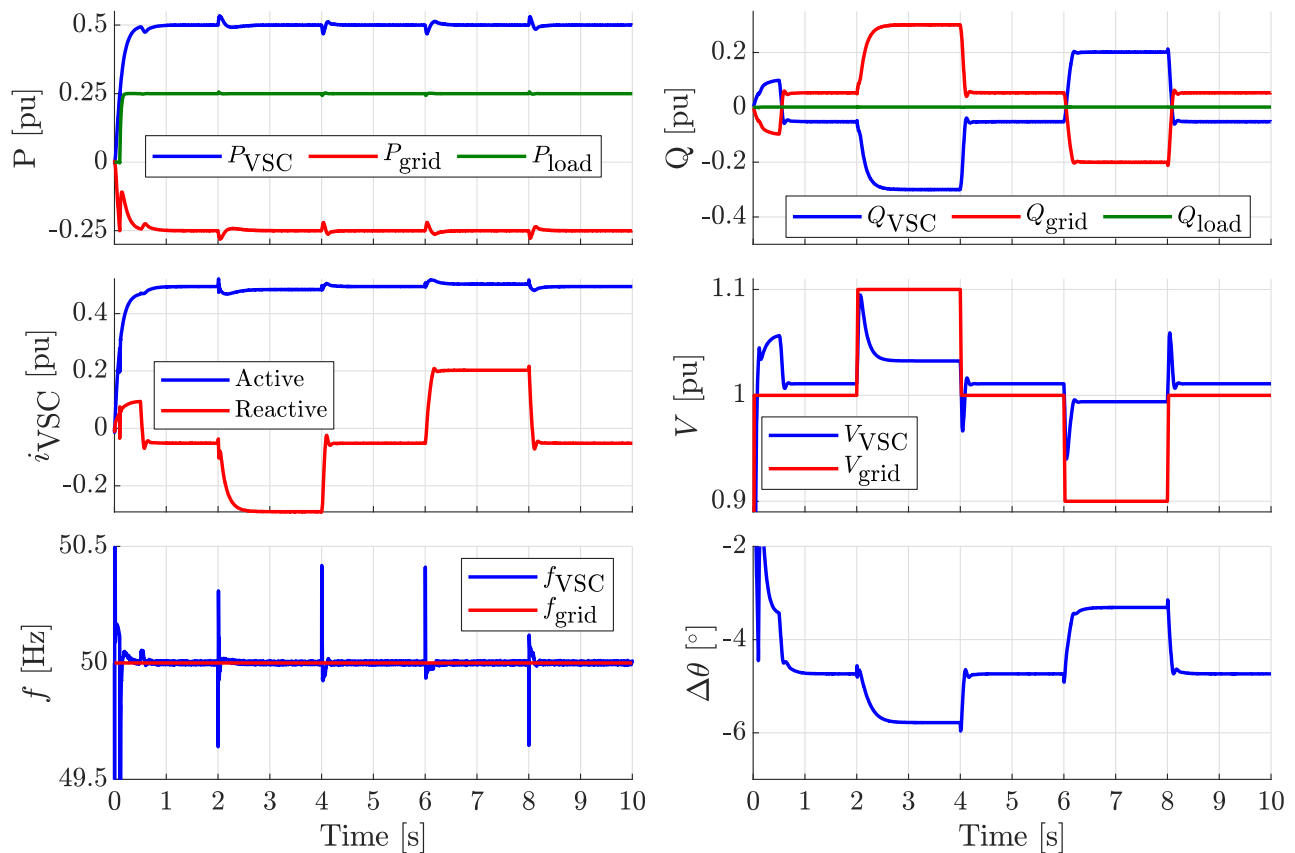


Figura 32. Escenario 6: cambio de tensión de la red

4.7 Escenario 7: faltas simétricas

En este escenario se muestra la respuesta del convertidor durante faltas simétricas (faltas trifásicas a tierra) en el punto de conexión. Se han examinado dos faltas: una falta profunda ($V \approx 0$ pu) durante 100 ms y otra falta de $V \approx 0,7$ pu durante 500 ms.

Como se muestra en la Figura 33, la primera falta ocurre en $t = 3$ s. Dado que la impedancia de falta es muy reducida, la tensión del VSC se reduce prácticamente a cero durante los 100 ms que está activa. Durante ese tiempo, se activa el control de inyección rápida de corriente reactiva. Debido a la profundidad de la falta, el convertidor inyecta la máxima corriente reactiva posible de acuerdo con sus limitaciones, en este caso, 1,1 pu. También se observa que la frecuencia medida por la PLL se ve afectada por los cambios abruptos de tensión, especialmente en el momento de la falta y su despeje. Tras este transitorio, el convertidor es capaz de continuar su operación normal. La Figura 34 muestra una ampliación de la corriente del convertidor, tensiones y frecuencia de la PLL de esta primera falta.

La segunda falta se produce en $t = 6$ s, cuya profundidad es menor que en la anterior, con una tensión de aproximadamente 0,7 pu. Esta tensión es suficientemente baja como para activar la inyección rápida de corriente reactiva, pero no como para saturar el convertidor. De esta forma, el convertidor inyecta aproximadamente 0,7 pu de corriente reactiva, mientras puede inyectar corriente activa.

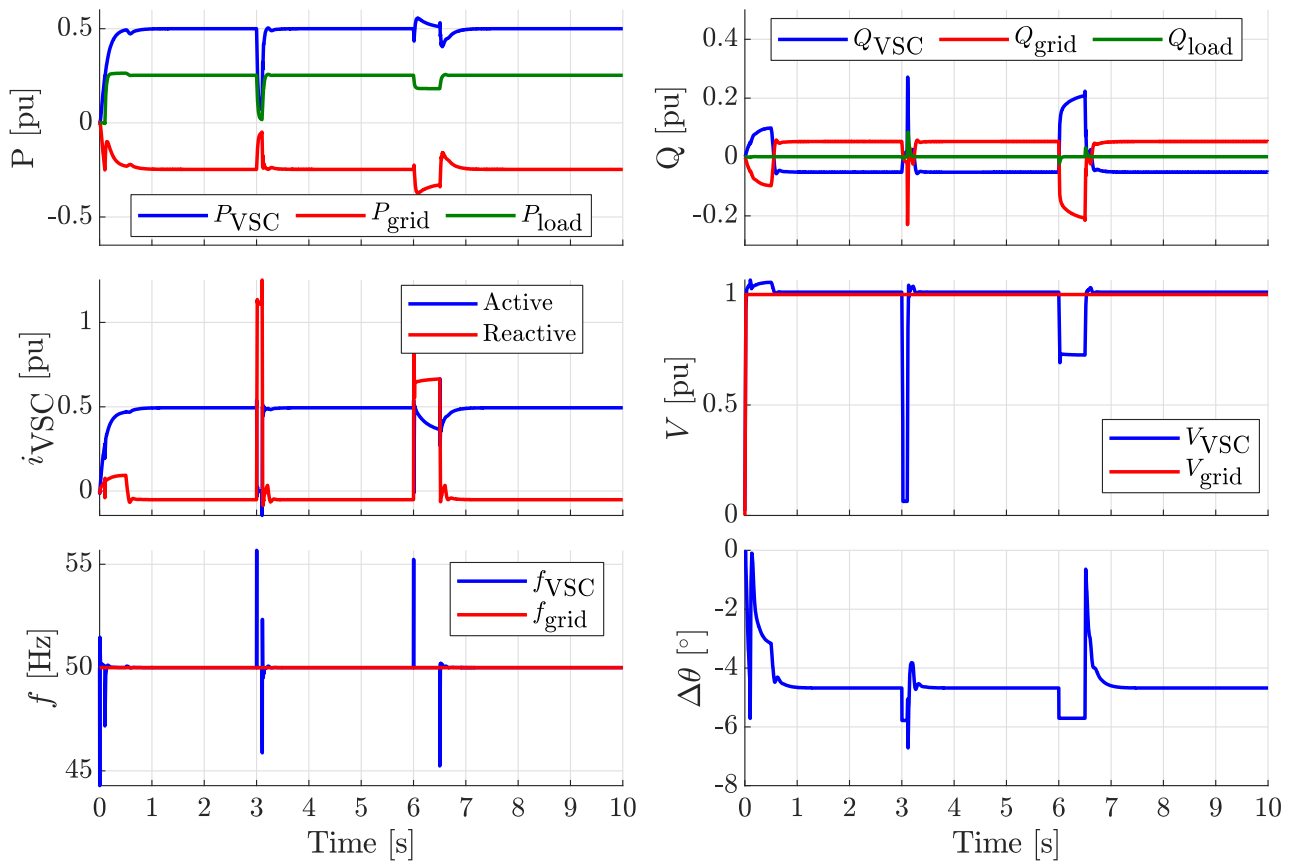


Figura 33. Escenario 7: faltas simétricas

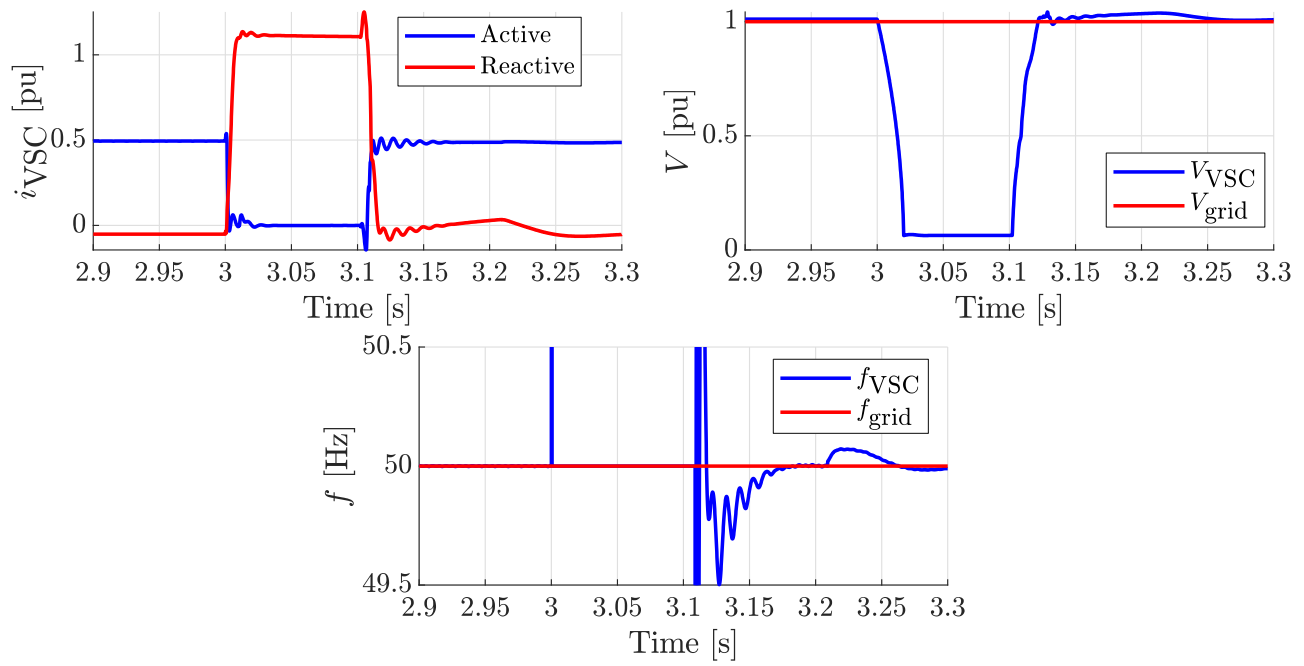


Figura 34. Escenario 7: ampliación de la corriente del VSC, tensiones y frecuencia de la PLL

4.8 Escenario 8: faltas asimétricas

En este escenario 8 se repite la secuencia de los eventos anteriores, pero se consideran faltas asimétricas (monofásicas a tierra). En $t = 3$ s, se produce una falta de la fase A a tierra con impedancia muy reducida. Los resultados se presentan en la Figura 35. Durante faltas asimétricas, el convertidor inyecta secuencia negativa a la red de acuerdo con el control de inyección rápida de corriente reactiva implementado para esta secuencia, como se muestra en la Figura 36. La referencia de corriente en secuencia negativa es proporcional a la tensión en secuencia negativa medida en el POC. La segunda falta se produce en $t = 6$ s, durante la cual la tensión de la fase A se mantiene aproximadamente alrededor de 0,7 pu debido a la impedancia de la falta. Al tratarse de un sistema desequilibrado, durante las faltas asimétricas se observan oscilaciones al doble de la frecuencia de la red (100 Hz) en la potencia del convertidor.

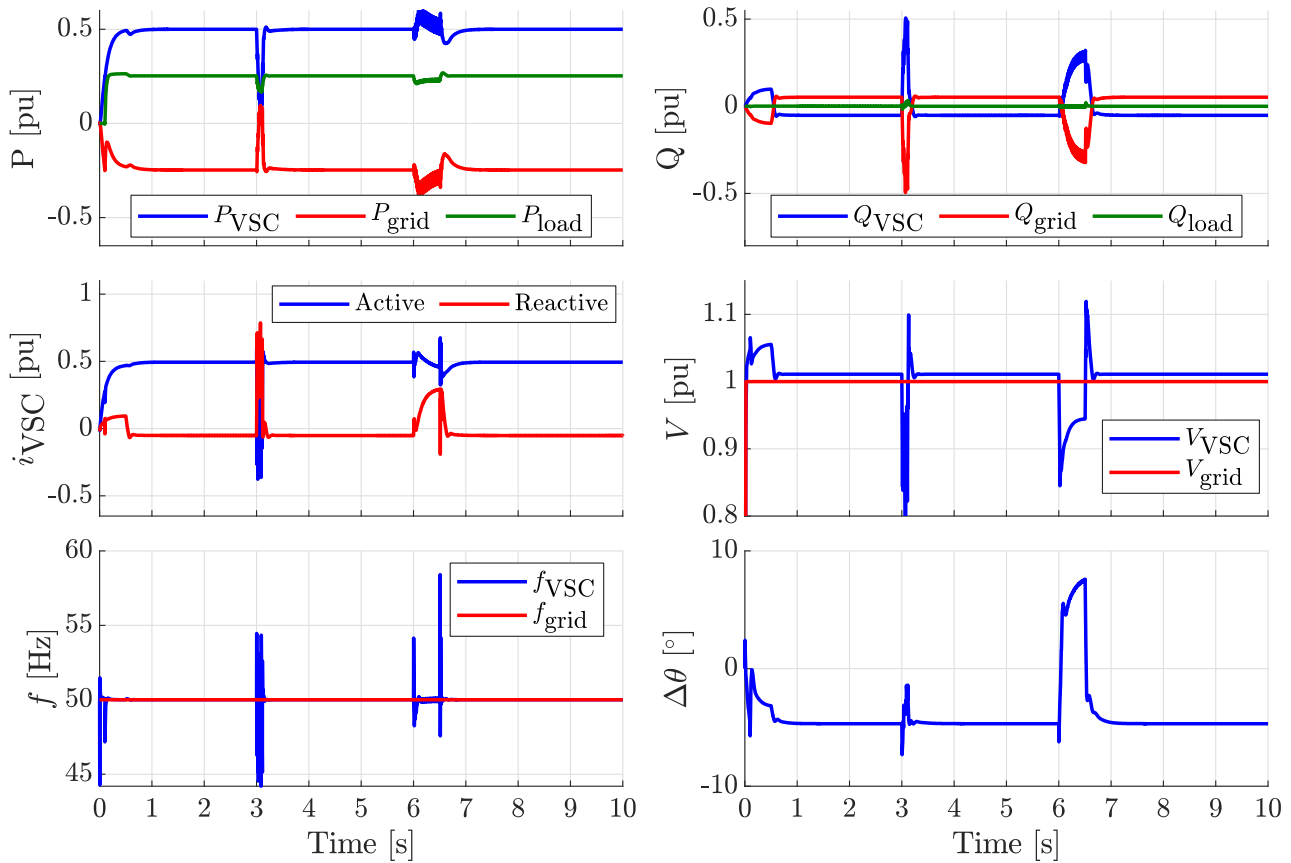


Figura 35. Escenario 8: faltas asimétricas

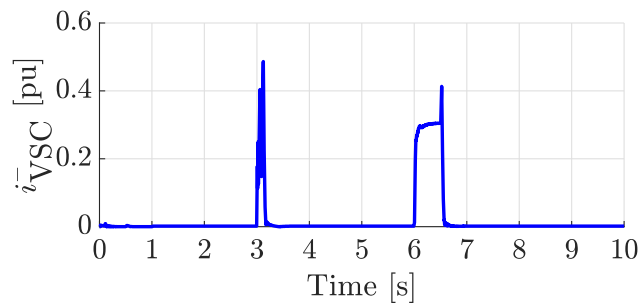


Figura 36. Escenario 8: corriente en secuencia negativa inyectada por el VSC

4.9 Escenario 9: cambio de SCR de la red

En este escenario se aplica un cambio de SCR de la red. Para ello se desconecta la impedancia Z2 de la Figura 4. El cambio se produce en $t = 2$ s. En la Figura 37 se muestra como el convertidor no es capaz de mantener sincronismo con la red tras el aumento de la impedancia de la red. Se observa grandes oscilaciones en la frecuencia de la PLL, que da lugar a una pérdida de sincronismo del convertidor y la inestabilidad del sistema.

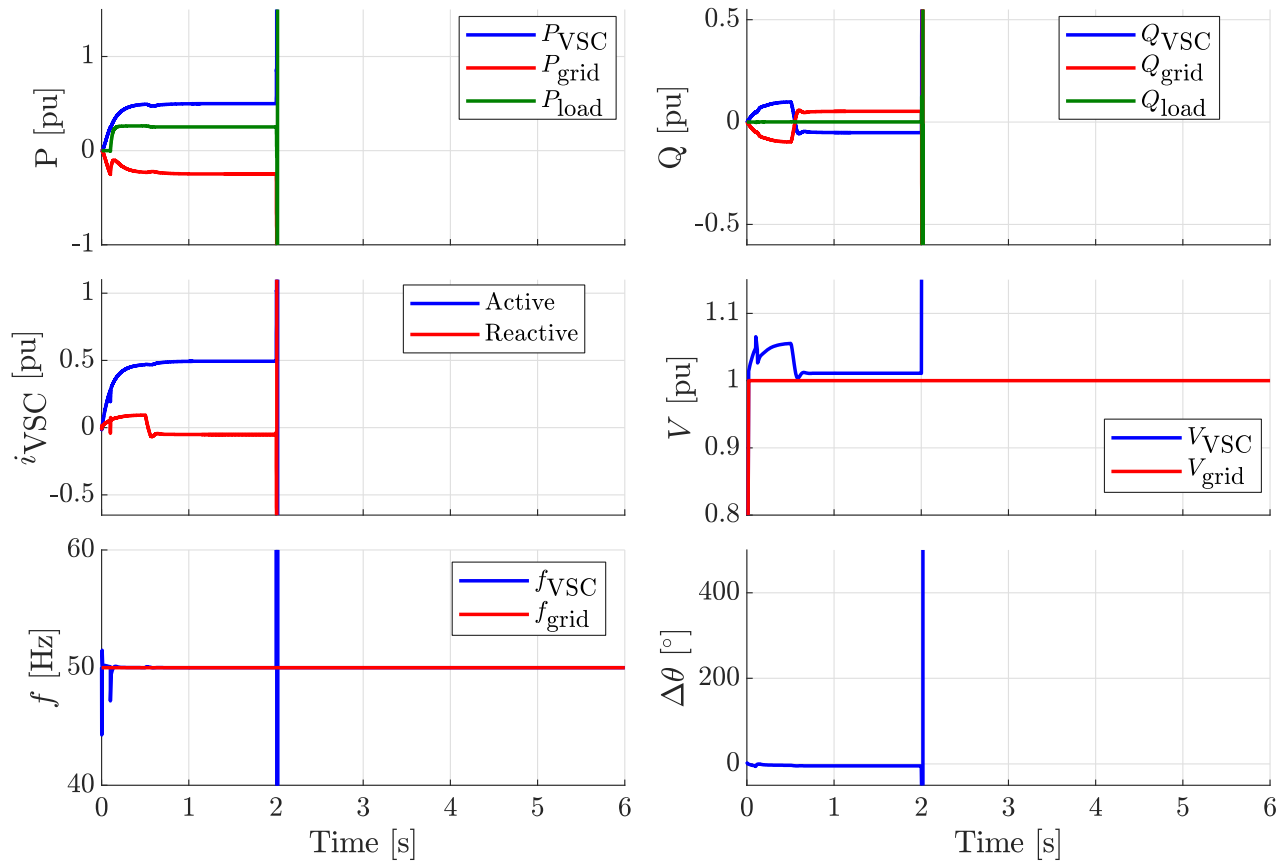


Figura 37. Escenario 9: cambio de SCR

4.10 Escenario 10: desconexión de la red

En este último se muestra la desconexión de la red, por lo que el convertidor está operando en modo isla. Esta desconexión se produce en $t = 2$ s. Dado que el control implementado es *grid-following*, el convertidor no puede mantener la tensión ni la frecuencia de la red aislada, dando lugar, de nuevo, a una inestabilidad. Estos dos últimos escenarios (9 y 10) muestran la necesidad de controles adicionales, por ejemplo, operación *grid-forming*, que permitan tanto la operación en isla como conectado a redes de SCR bajo. En otros informes de este proyecto se muestra la mejora que ofrece la operación en *grid-forming* en estos escenarios.

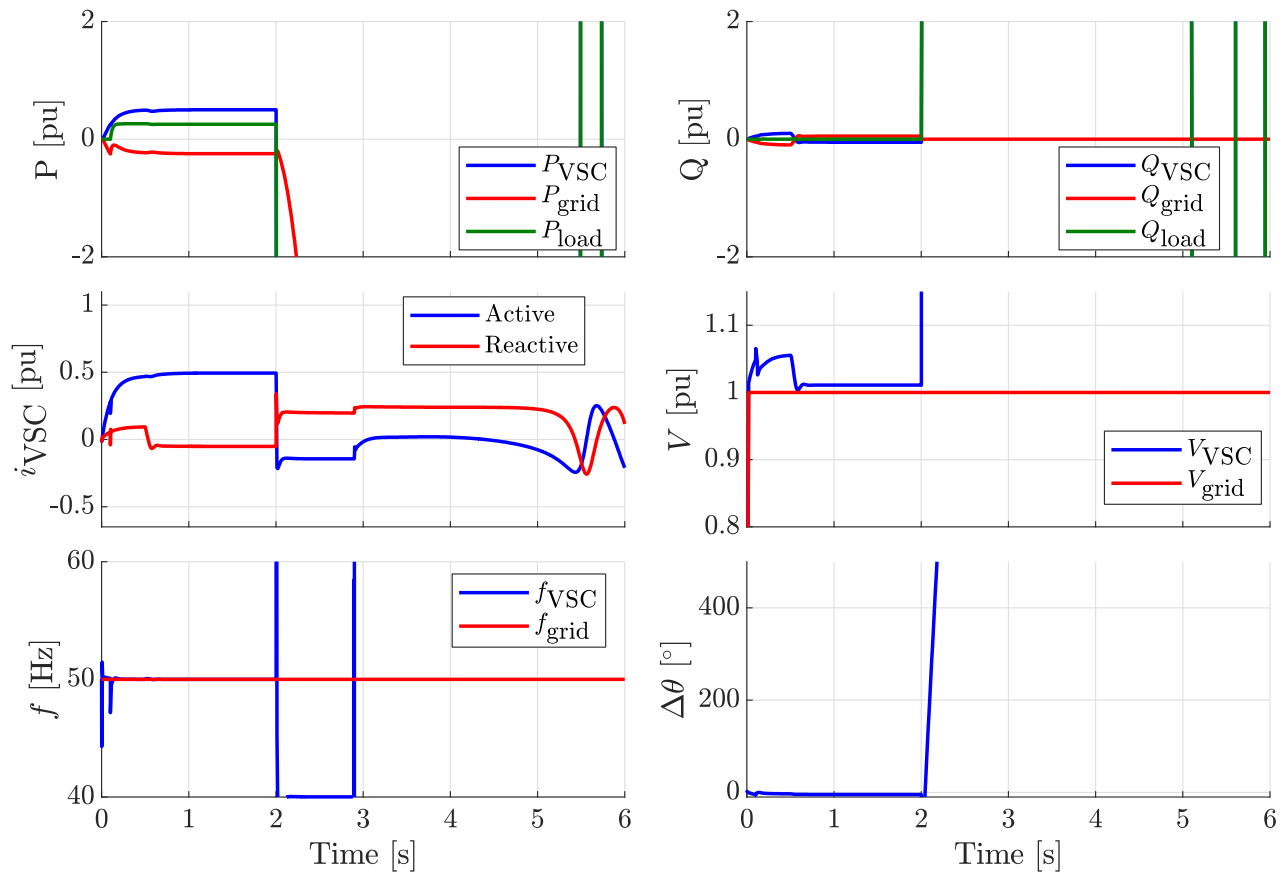


Figura 38. Escenario 10: desconexión de la red

5 Conclusiones

En el modelo *GROL_model.pscx* se ha implementado un convertidor con control *grid-following*. Este control se ha evaluado mediante varios escenarios para verificar las diferentes funcionalidades del convertidor. En particular, se ha implementado una red basada en la descrita en la Guía FNN, que está compuesta por una red Thévenin, una carga activa y el propio convertidor.

Las principales conclusiones de los escenarios anteriores son:

- Escenario 2: la potencia activa inyectada por el convertidor sigue la referencia y tiene una respuesta de primer orden con una constante de tiempo de unos 100 ms.
- Escenario 3: se puede variar la potencia reactiva intercambiada por el convertidor modificando su referencia. Sin embargo, si el *droop* de tensión está activo, la potencia reactiva real no será igual que la referencia especificada.
- Escenario 4: los cambios de ángulo de la red modifican transitoriamente el flujo de potencias intercambiado con el convertidor debido a la desincronización del convertidor con la red. La PLL es capaz de resincronizar el convertidor con la red rápidamente y corregir la desviación de potencia.
- Escenario 5: el convertidor puede dar soporte a la frecuencia gracias a la medida de frecuencia de la PLL y al *droop* de frecuencia.
- Escenario 6: el convertidor puede reducir la desviación de tensión en el punto de conexión a pesar de los cambios de tensión de la red modificando su inyección de potencia reactiva de acuerdo con el control de *droop* de tensión.
- Escenario 7: durante faltas el convertidor se mantiene conectado inyectando corriente reactiva proporcional a profundidad de la falta. La PLL se bloquea si la tensión es menor que un valor especificado (0,85 pu).
- Escenario 8: el convertidor inyecta corriente en secuencia negativa acorde con la secuencia negativa de la tensión medida en el punto de conexión.
- Escenarios 9 y 10: muestran la necesidad de controladores adicionales que permitan tanto la operación en isla como el convertidor conectado a redes débiles. Un ejemplo de estos controles puede ser la operación *grid-forming*.

Estos resultados muestran las principales funcionalidades de un convertidor operando en modo *grid-following*.

6 Referencias

- [1] VDE, «FNN Guideline: Grid forming behaviour of HVDC systems and DC-connected PPMs».
- [2] S.-K. Chung, «A Phase Tracking System for Three Phase Utility Interface Inverters,» *IEEE Transactions on Power Electronics*, vol. 15, nº 3, pp. 431-438, May 2000.
- [3] L. Harnefors y H.-P. Nee, «Model-Based Current Control of AC Machines Using the Internal Model Control Method,» *IEEE Transactions on Industry Applications*, vol. 34, nº 1, pp. 133-141, Jan/Feb 1998.
- [4] Orden TED/749/2020, «por la que se establecen los requisitos técnicos para la conexión a la red necesarios para la implementación de los códigos de red de conexión,» *Boletín Oficial del Estado*, pp. 62406-62458, 2020.

