

# Synchronous reference frame (SRF) PLL

TN103 | Posted on April 9, 2021 | Updated on May 7, 2025



**Antonin STAMPBACH**  
Development Engineer  
imperix • in



**Jessy ANÇAY**  
Sales & Project Engineer  
imperix • in

---

## Table of Contents

- [What is a synchronous reference frame PLL?](#)
- [SRF PLL implementation](#)
  - [Software implementation](#)
  - [Tuning of SRF PLL](#)
- [Going further : Double Decoupled Synchronous Reference Frame \(DDSRF\) PLL](#)
  - [Decoupling](#)
  - [Implementation of the DDSRF-PLL](#)
- [Academic references](#)
- [Use case examples](#)

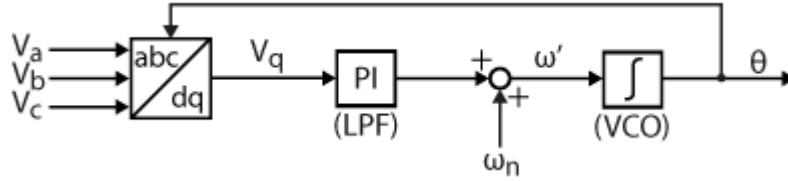
This note provides insights into the operating principle of a synchronous reference frame PLL (SRF PLL), also known as DQ-type PLL. An implementation of an SRF PLL is found in the [ACG SDK](#) as the [DQ-type PLL](#) block and in the [CPP SDK](#) in the API folder of the template project. For a different type of PLL, please refer to the page on the [SOGI PLL](#). Also, note that this page will focus on the three-phase implementation of the PLL. Single phase PLL are discussed in [Fictive axis emulation \(FAE\) for single-phase inverter](#). Finally, typical use cases for DQ-type PLLs are listed at the end of the page.

## What is a synchronous reference frame PLL?

Most of today's grid-tied power converters aim to control active and reactive power flow. Accurate phase tracking systems are then required to synchronize the controlled currents or voltages with the utility grid voltages. Many grid-synchronization techniques exist and an extensive review of the possible implementations can be found in the [Grid synchronization methods](#) page or in [1].

A synchronous reference frame PLL is a basic type of phase-locked loop based on the Park transform. In a nutshell, the SRF PLL is built using a Park transformation that acts as a phase detector, a low-pass filter (LPF) usually in the form of a PI regulator, and a voltage-controlled oscillator (VCO) typically made from an integrator [2]. The objective of this PLL is then to minimize either the direct or quadrature axis reference voltage. This will then ensure that the phase angle of the rotating reference frame of the park transformation matches the phase angle of the utility grid voltage vector [3].

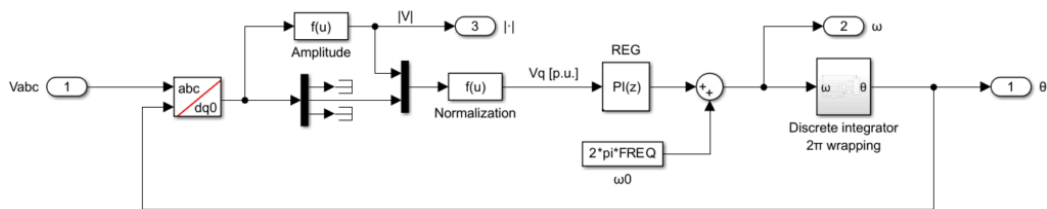
The general principle of an SRF PLL is given below:



The general principle of the SRF PLL

## SRF PLL implementation

As mentioned, the idea behind the synchronous reference frame PLL is to detect the phase angle by synchronizing the rotating frame reference of the PLL to the utility grid voltages, ensuring that  $V_q = 0$ . The structure proposed for the SRF-PLL is shown below. The park transform and the PI controller are very classic. Additionally,  $V_q$  is normalized so that the tuning of the PI controller does not depend on the amplitude of the grid voltage. Finally, the integrator is based on the forward Euler method, with the only specificity being that the output is wrapped between  $[0; 2\pi[$ .



Implementation of the three-phase SRF PLL

Moreover, when the estimated frequency from the PLL is used for applications beyond the PLL itself, it is highly recommended to filter the frequency estimation using a low-pass filter with a cutoff frequency of around 15 Hz. This ensures a smooth and reliable frequency estimation for stable performance in different applications.

Regarding the dynamic performances of the SRF PLL, it is difficult to get both fast-tracking and good filtering characteristics. Indeed, as stated in [2], a trade-off

between these two characteristics is necessary when tuning the control loop parameters. For a PLL with better dynamic performance, one can look into the page: [SOGI PLL](#).

## Software implementation

*Simulink model*

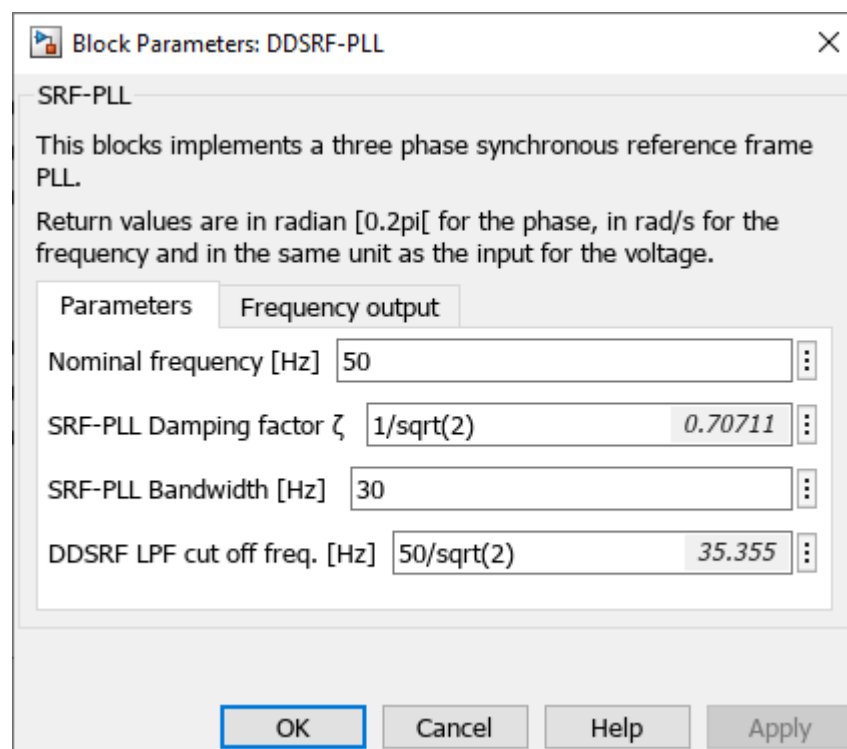
[SRF\\_DDSRF\\_SimulinkDownload](#)

*PLECS model*

[SRF\\_DDSRF\\_PLECSDownload](#)

The provided Simulink and PLECS model implements both the SRF-PLL and DDSRF-PLL using the [ACG SDK](#) for simulation and code generation.

### Simulink



DDSRF-PLL Simulink block

### PLECS

Block Parameters: SRF\_DDSRF\_PLECS/DDSRF-PLL

×

DDSRF-PLL (mask)

This block implements a standardized synchronous frame PLL. This means that the gain does not depend on the input voltage norm.

Return values are in radian [0.2pi[ for the phase, in rad/s for the frequency and in the same unit as the input for the voltage.

More informations are available in the Help section.

Parameters

Frequency Output

Nominal frequency [Hz] :

50

SRF-PLL Damping factor  $\zeta$  :

1/sqrt(2)

SRF-PLL Bandwidth [Hz] :

30

DDSRF LPF cut off freq. [Hz]:

50/sqrt(2)

Sample time [s] :

Ts

OK

Cancel

Apply

Help

DDSRF-PLL PLECS block

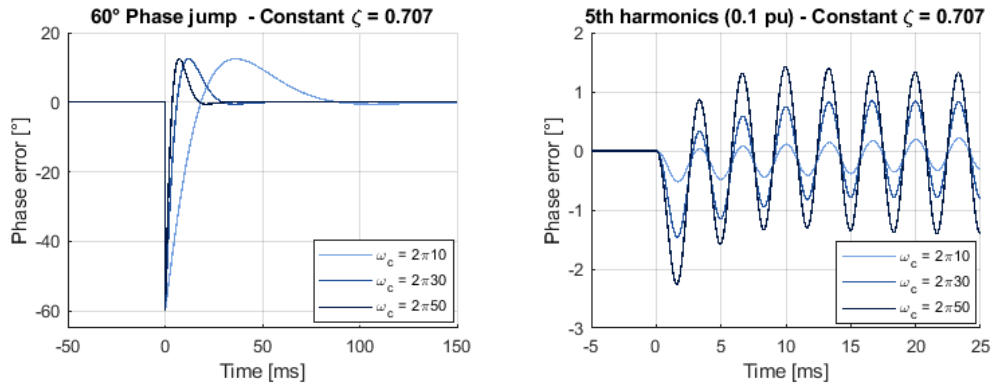
## Tuning of SRF PLL

To complete the implementation of the SRF PLL, the PI controller gains have to be determined. This can be achieved by linearizing the control loop [4] and deriving the following transfer function.

$$P(s) = \frac{\hat{\theta}}{\psi}(s) = \frac{2\zeta\omega_c s + \omega_c^2}{s^2 + 2\zeta\omega_c s + \omega_c^2} \quad \text{with} \quad K_p = 2\zeta\omega_c \quad \text{and} \quad K_i = \omega_c^2$$

Where  $\psi$  represents a phase disturbance,  $\hat{\theta}$  is the estimated phase,  $\zeta$  is the damping factor, and  $\omega_c$  is the PLL bandwidth. To achieve a good trade-off between settling time and overshoot, a commonly used damping factor is  $\zeta = \frac{1}{\sqrt{2}}$ . On the other hand, the bandwidth parameter,  $\omega_c$ , determines the settling time of the PLL and its disturbance rejection capability. A higher  $\omega_c$  results in a shorter settling time but increases sensitivity to disturbances. Therefore,  $\omega_c$  should be chosen based on the quality of the grid voltage. A typical value for real grids is approximately  $\omega_c \approx 2\pi 30$  rad/s.

The figures below illustrate the impact of  $\omega_c$  during a phase jump and when 5th harmonic distortion is present in the grid voltage.



Tuning effects of the SRF-PLL

## Experimental results

Experimental results and detailed comparisons with other PLLs are available on the [Grid synchronization methods](#) page.

## Going further : Double Decoupled Synchronous Reference Frame (DDSRF) PLL

Thanks to its simplicity and performance under nominal conditions, the SRF-PLL has become the PLL of choice for applications where robustness against disturbances is not required. However, in the presence of unbalance in the grid voltage, an oscillating term at twice the fundamental frequency appears after the Park transform. The DDSRF-PLL effectively addresses this issue by using a decoupling network to separate the positive and negative sequence components, ensuring accurate detection even under unbalanced voltages.

## Decoupling

The idea behind this decoupling is to generate a signal that mimics the parasitic signal at twice the fundamental frequency, using known parameters or those obtained through filtering. This generated signal is then subtracted from the main signal to achieve decoupling of the positive and negative sequence voltages. The mathematical development of this decoupling process is shown below.

Mathematical development of the decoupling

In the presence of unbalanced voltage, both positive and negative sequence components exist and phase voltages can be expressed as in the dq reference frame as follow.

$$V_{dq^+} = V^+ \begin{bmatrix} \cos(\omega t - \theta') \\ \sin(\omega t - \theta') \end{bmatrix} + V^- \begin{bmatrix} \cos(-\omega t - \theta' + \phi^-) \\ \sin(-\omega t - \theta' + \phi^-) \end{bmatrix}$$

$$V_{dq^-} = V^+ \begin{bmatrix} \cos(\omega t + \theta') \\ \sin(\omega t + \theta') \end{bmatrix} + V^- \begin{bmatrix} \cos(-\omega t + \theta' + \phi^-) \\ \sin(-\omega t + \theta' + \phi^-) \end{bmatrix}$$

With  $V^\pm$  the positive and negative sequence voltage,  $\omega t$  the real grid phase,  $\theta'$  the estimated grid phase and  $\phi^-$  the negative sequence phase offset. Assuming a reasonably well-parameterized PLL is used, it can be assumed that  $\omega t \approx \theta'$  resulting in the following voltages.

$$V_{dq^+} \approx V^+ \begin{bmatrix} 1 \\ 0 \end{bmatrix} + V^- \begin{bmatrix} \cos(-2\omega t + \phi^-) \\ \sin(-2\omega t + \phi^-) \end{bmatrix}$$

$$V_{dq^-} \approx V^+ \begin{bmatrix} \cos(2\omega t) \\ \sin(2\omega t) \end{bmatrix} + V^- \begin{bmatrix} \cos(\phi^-) \\ \sin(\phi^-) \end{bmatrix}$$

Indeed, the positive and negative sequences are coupled. To synchronize the PLL on  $V_q^+$  the oscillating term at  $2\omega$  must be removed. This is achieved by reconstructing the perturbation signal using trigonometry and filtering the negative sequence voltage. The reconstructed signal is then subtracted from the positive sequence voltage to obtain a pure positive sequence voltage. The following equation illustrates the reconstruction of the parasitic signal using the estimated phase and the filtered negative sequence.

$$V^- \begin{bmatrix} \cos(-2\omega t + \phi^-) \\ \sin(-2\omega t + \phi^-) \end{bmatrix} \stackrel{\theta' \approx \omega t}{=} \begin{bmatrix} \overbrace{V^- \cos(\phi^-) \cos(2\theta')}^{\overline{v_{d-}'}} + \overbrace{V^- \sin(\phi^-) \sin(2\theta')}^{\overline{v_{q-}'}} \\ -\underbrace{V^- \cos(\phi^-) \sin(2\theta')}_{\overline{v_{d-}'}} + \underbrace{V^- \sin(\phi^-) \cos(2\theta')}_{\overline{v_{q-}'}} \end{bmatrix}$$

The above  $\overline{v_{dq-}'}$  components are obtained by filtering the negative sequence voltage with a low-pass filter (LPF) whose cutoff frequency is  $\leq 2\omega$ . [4] recommends using a first-order Butterworth filter with a cutoff frequency of  $\omega_0 = \frac{\omega}{\sqrt{2}}$ .

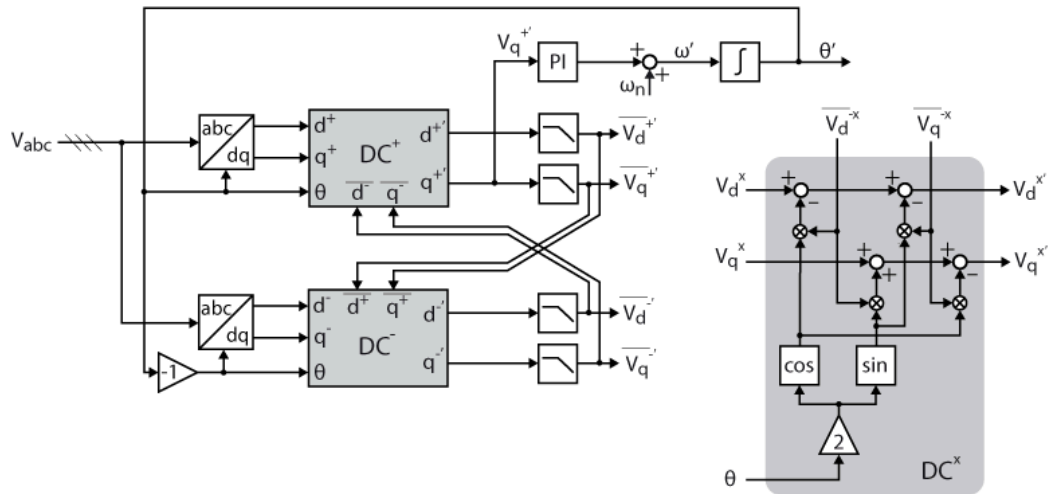
$$LPF(V_{dq^-}) = \overline{v_{dq-}'} \approx V^- \begin{bmatrix} \cos(\phi^-) \\ \sin(\phi^-) \end{bmatrix}$$

Finally, the reconstructed signal is subtracted from the perturbed signal, allowing for effective decoupling of the sequences, as shown below.

$$V_{dq^+} \approx V^+ \begin{bmatrix} 1 \\ 0 \end{bmatrix} + V^- \begin{bmatrix} \cos(-2\omega t + \phi^-) \\ \sin(-2\omega t + \phi^-) \end{bmatrix} - V^- \begin{bmatrix} \cos(-2\theta' + \phi^-) \\ \sin(-2\theta' + \phi^-) \end{bmatrix} \approx V^+ \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

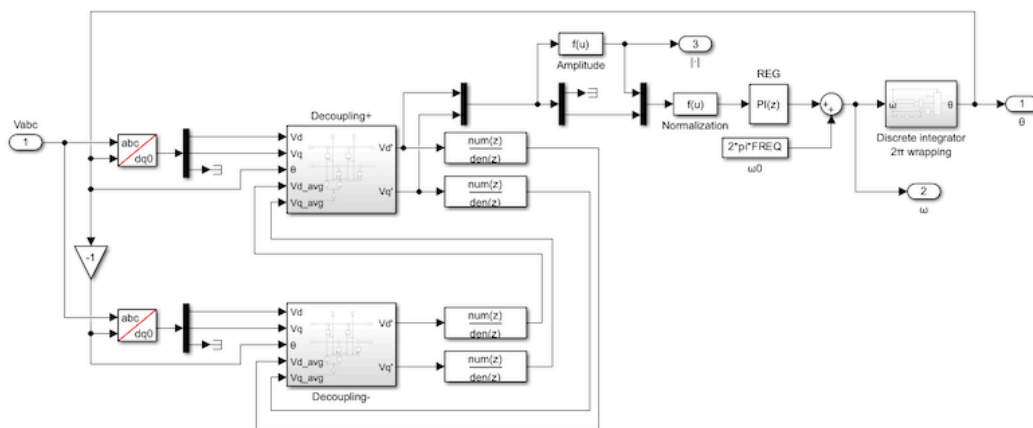
## Implementation of the DDSRF-PLL

The algorithm for decoupling the positive and negative sequences is implemented as shown below. A decoupling network for both sequences is added after the Park transform ensuring that only the positive sequence is used in the SRF-PLL.



Implementation of the DDSRF-PLL

Where the  $DC^x$  blocks implement the mathematics developed above to achieve decoupled voltages in the dq reference frame. The complete implementation of the DDSRF-PLL on Simulink is shown below



DDSRF-PLL Simulink implementation

The corresponding simulink file can be downloaded above, in the same file as the SRF-PLL. Experimental results and detailed comparisons with other PLLs are available on the [Grid synchronization methods](#) page.

## Academic references

[1] M. Boyra and J. Thomas, "A review on synchronization methods for grid-connected three-phase VSC under unbalanced and distorted conditions," in Proc. EPE Conf., Birmingham, 2011.

[2] S.-K. Chung, "A phase tracking system for three-phase utility interface inverters," IEEE Transactions on Power Electronics, vol. 15, no. 3, May 2000.

[3] L. Arruda and B. Silva, S.M. and Cardoso Filho, "PLL structures for utility connected systems," in 2001 IEEE Industry Applications Society 36th Annual Meeting (IAS'01), Chicago, USA, Sep./Oct. 30–4, 2001

[4] P. Rodriguez, J. Pou, J. Bergas, J. I. Candela, R. P. Burgos and D. Boroyevich, "Decoupled Double Synchronous Reference Frame PLL for Power Converters Control," in IEEE Transactions on Power Electronics, vol. 22, no. 2, March 2007.

## Use case examples

As introduced above, phase-locked loops are used in most of today's grid-tied power converters. Here are a few examples of such converters:

- [Three-phase PV inverter for grid-tied applications](#)
- [Fast electric vehicle charger with intermediate energy storage](#)
- [Multi-converter system for micro-grid](#)
- [Three-phase MMC converter](#)
- [Solid-state transformers](#)
- [Cascaded H-bridge converter](#)
- [Static synchronous compensator \(STATCOM\)](#)