dq0 to Alpha-Beta-Zero

SD034 | Posted on June 24, 2021 | Updated on May 27, 2025

Julien ORSINGER
Power Applications Specialist
imperix • in

Table of Contents

The “dq0 to Alpha-Beta-Zero” converts a space vector from a rotating (*dq0*) to a stationary (*αβ0*) reference frame. The angle of the rotating reference frame is given by the second input $\theta = \omega t$.

The transformation is performed using the following rotation:

$$\begin{bmatrix} V_\alpha \\ V_\beta \\ V_0 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix}$$

Simulink block

Signal specification

- The first input is a vector of dimension 3, containing the *dq0* components of the space vector in the rotating reference frame.
- The second input is the angle θ of the rotating reference frame, in radians.
- The output is a vector of dimension 3, containing the *αβ0* components of the space vector in the stationary reference frame.

Parameters

None.

PLECS block

None. The PLECS block *Transformation RRF->SRF* can be used instead.

C++ functions

The user template located in the installation folder of CPP SDK contains an API folder with implementations of the coordinate transformation functions. The *dq0 to αβ0* function is the following:

```
void DQ02ABG(SpaceVector *fixed, const SpaceVector *rotating, const float theta);
```
Code language: C++ (cpp)

Parameters

- `fixed`: pointer on the *αβ0* space vector that will be updated. The `SpaceVector` structure is defined below.
- `rotating`: pointer on the *dq0* space vector that will be transformed. The `SpaceVector` structure is defined below.
- theta: the angle of the rotating reference frame, in radians.

```
typedef struct{
        float real;       // d- or alpha-axis component
        float imaginary;  // q- or beta-axis component
        float offset;     // homopolar component
} SpaceVector;
```
Code language: C++ (cpp)