Clarke's and Park's Transformations

BPRA047, BPRA048

Content

Introduction

- The performance of three-phase AC machines are described by their voltage equations and inductances.
  - It is well known that some machine inductances are functions of rotor speed.

$$(V_S) = \begin{bmatrix} v_{s1} \\ v_{s2} \\ v_{s3} \end{bmatrix}$$

- The coefficients of the differential equations, which describe the behavior of these machines, are time varying except when the rotor is stalled.

Introduction

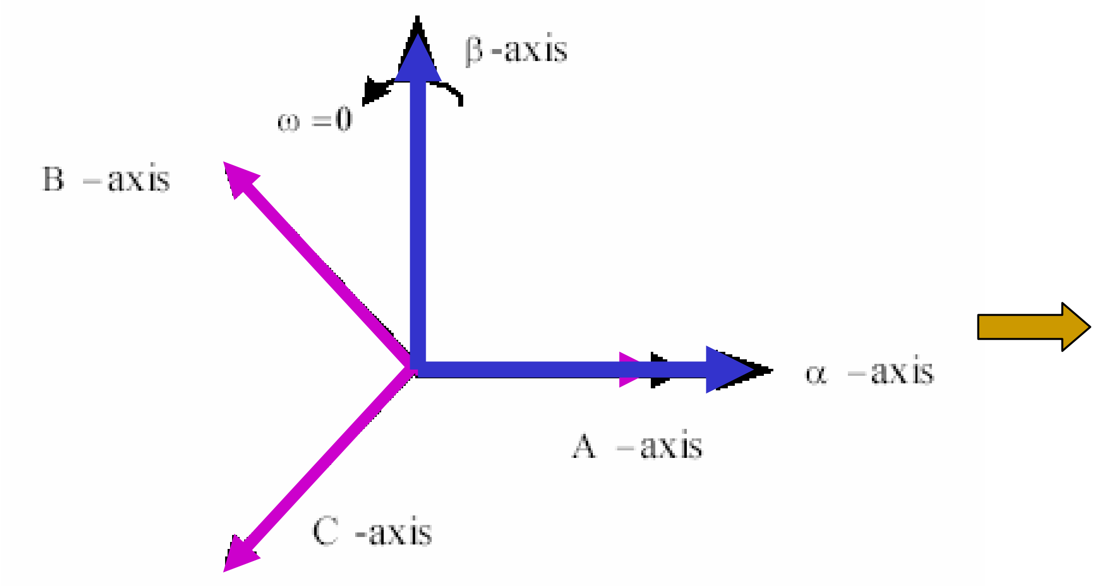
- A change of variables is often used to reduce the complexity of these differential equations.

- In this chapter, the well-known Clarke and Park transformations are introduced, modeled, and implemented on the LF2407 DSP.

- Using these transformations, many properties of electric machines can be studied without complexities in the voltage equations.

Clarke's Transformation

- The transformation of stationary circuits to a stationary reference frame was developed by E. Clarke.
- The stationary two-phase variables of Clarke's transformation are denoted as α and β, α-axis and β-axis are orthogonal.



$$v_\alpha = v_a$$

$$v_\beta = \frac{2v_b + v_a}{\sqrt{3}}$$

Clarke's Transformation

- In order for the transformation to be invertible, a third variable, known as the zero-sequence component, is added.
- The resulting transformation is

$$\left[f_{\alpha\beta 0}\right] = T_{\alpha\beta 0}\left[f_{abc}\right]$$

where *f* represents voltage, current, flux linkages, or electric charge; the transformation matrix T:

$$T_{\alpha\beta 0} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Inverse Clarke's Transformation

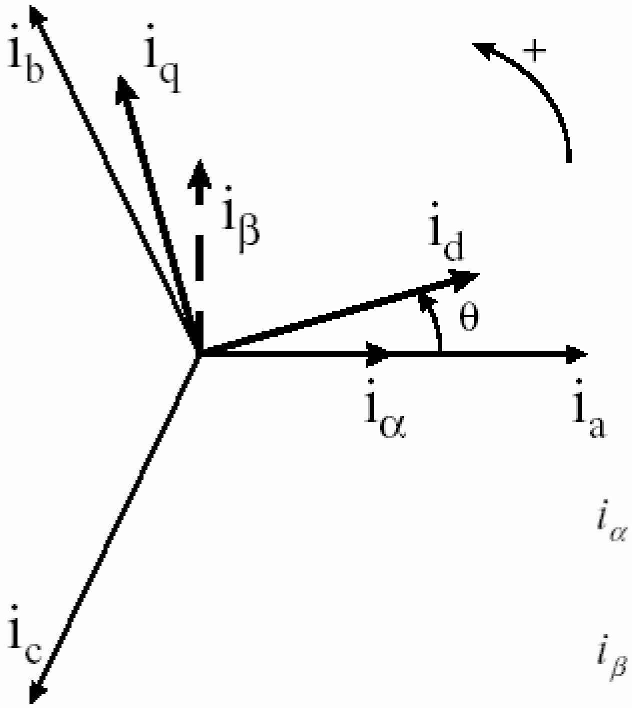
- The inverse transformation is given by

$$[f_{abc}] = T_{\alpha\beta 0}{}^{-1}[f_{\alpha\beta 0}]$$

where the inverse transformation matrix T^{-1} is presented by

$$T_{\alpha\beta 0}{}^{-1} = \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix}$$

Clarke's Transformation



$$i_\alpha = \frac{2}{3}\cdot i_a - \frac{1}{3}(i_b - i_c)$$

$$i_\beta = \frac{2}{\sqrt{3}}(i_b - i_c)$$

$$i_o = \frac{2}{3}(i_a + i_b + i_c)$$

$$i_\alpha = i_a$$

$$i_\beta = \frac{1}{\sqrt{3}}\cdot i_a + \frac{2}{\sqrt{3}} i_b$$

$$i_a + i_b + i_c = 0$$

$$i_a = i_\alpha$$

$$i_b = -\frac{1}{2}\cdot i_\alpha + \frac{\sqrt{3}}{2}\cdot i_\beta$$

$$i_c = -\frac{1}{2}\cdot i_\alpha - \frac{\sqrt{3}}{2}\cdot i_\beta$$

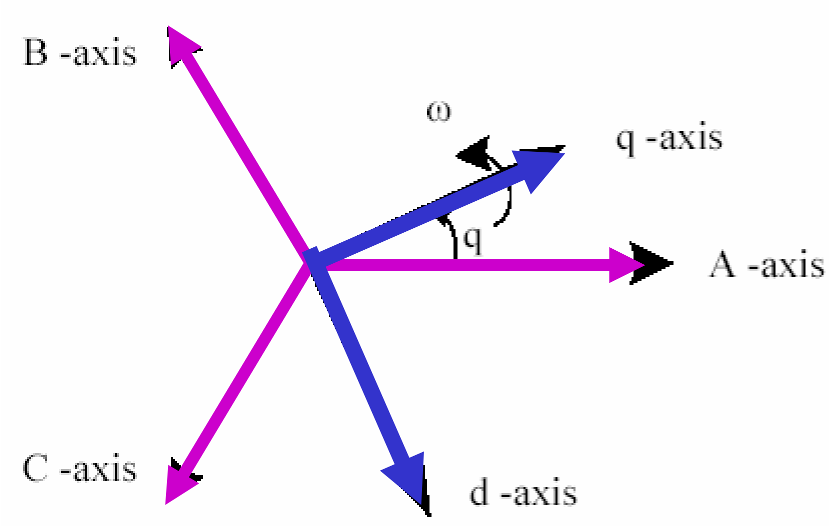
Park's Transformation

- In the late 1920s, R.H. Park introduced a new approach to electric machine analysis.
- He formulated a change of variables associated with fictitious windings rotating with the rotor.
- He referred the stator and rotor variables to a reference frame fixed on the rotor.
- From the rotor point of view, all the variables can be observed as constant values.

- Park's transformation, a revolution in machine analysis, has the unique property of eliminating all time varying inductances from the voltage equations of three-phase ac machines due to the rotor spinning.

Park's Transformation

- Park's transformation is a well-known three-phase to two-phase transformation in synchronous machine analysis.



$$\begin{bmatrix} v_{sa} \\ v_{sb} \\ v_{sc} \end{bmatrix} = \begin{bmatrix} \cos\theta_S & -\sin\theta_S & 1 \\ \cos(\theta_S - \frac{2\pi}{3}) & -\sin(\theta_S - \frac{2\pi}{3}) & 1 \\ \cos(\theta_S - \frac{4\pi}{3}) & -\sin(\theta_S - \frac{4\pi}{3}) & 1 \end{bmatrix} \begin{bmatrix} v_{sd} \\ v_{Sq} \\ v_{So} \end{bmatrix}$$

$$\begin{bmatrix} v_{sa} \\ v_{sb} \\ v_{sc} \end{bmatrix} = [P(\theta_S)] \cdot \begin{bmatrix} v_{sd} \\ v_{sq} \\ v_{so} \end{bmatrix} \text{ and } \begin{bmatrix} v_{sd} \\ v_{Sq} \\ v_{So} \end{bmatrix} = [P(\theta_S)]^{-1} \begin{bmatrix} v_{s1} \\ v_{S2} \\ v_{S3} \end{bmatrix}$$

Park's Transformation

- The Park's transformation equation is of the form

$$\left[f_{qd0s}\right] = T_{qd0}(\theta)\left[f_{abcs}\right]$$

$$T_{qd0s}(\theta) = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ \sin(\theta) & \sin(\theta - \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

- θ is the angular displacement of Park's reference frame

Inverse Park's Transformation

- It can be shown that for the inverse transformation we can write

$$[f_{abcs}] = T_{qd0}(\theta)^{-1} \cdot [f_{qd0s}]$$

where the inverse of Park's transformation matrix is given by

$$T_{qd0}(\theta)^{-1} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 1 \\ \cos(\theta - \frac{2\pi}{3}) & \sin(\theta - \frac{2\pi}{3}) & 1 \\ \cos(\theta + \frac{2\pi}{3}) & \sin(\theta + \frac{2\pi}{3}) & 1 \end{bmatrix}$$

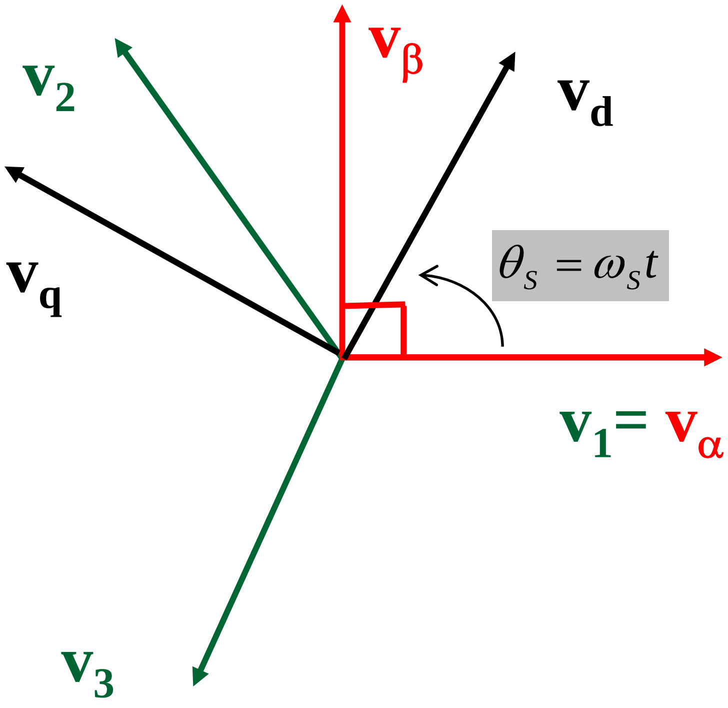
Park's Transformation

- The angular displacement θ must be continuous, but the angular velocity associated with the change of variables is unspecified.
- The frame of reference may rotate at any constant, varying angular velocity, or it may remain stationary.
- The angular velocity of the transformation can be chosen arbitrarily to best fit the system equation solution or to satisfy the system constraints.

- The change of variables may be applied to variables of any waveform and time sequence;
  - however, we will find that the transformation given above is particularly appropriate for an a-b-c sequence.

Park's Transformation

- Park's Transform is usually split into CLARKE transform and one rotation;
- CLARKE converts balanced three phase quantities into balanced two phase orthogonal quantities;



CLARKE

$$v_\alpha = v_1$$

$$v_\beta = \frac{2.v_2 + v_1}{\sqrt{3}}$$

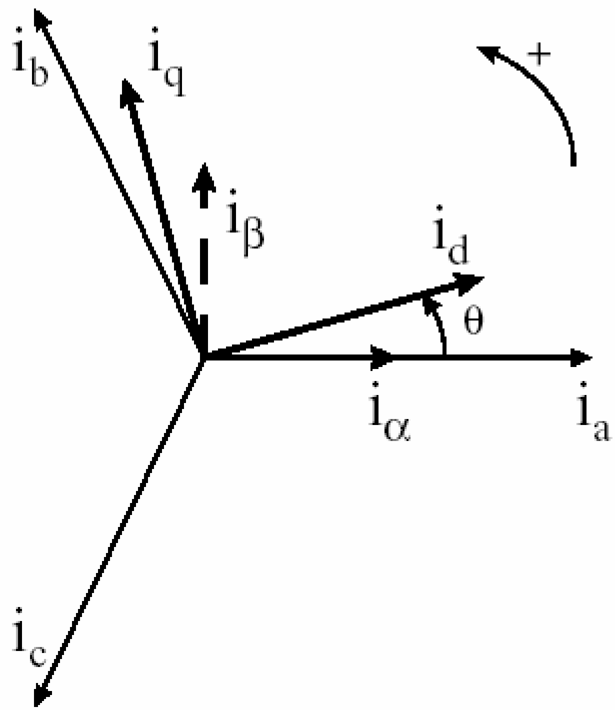
+

Rotation

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\theta_S) & \sin(\theta_S) \\ -\sin(\theta_S) & \cos(\theta_S) \end{bmatrix} \cdot \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}$$

PARK TRANSFORM

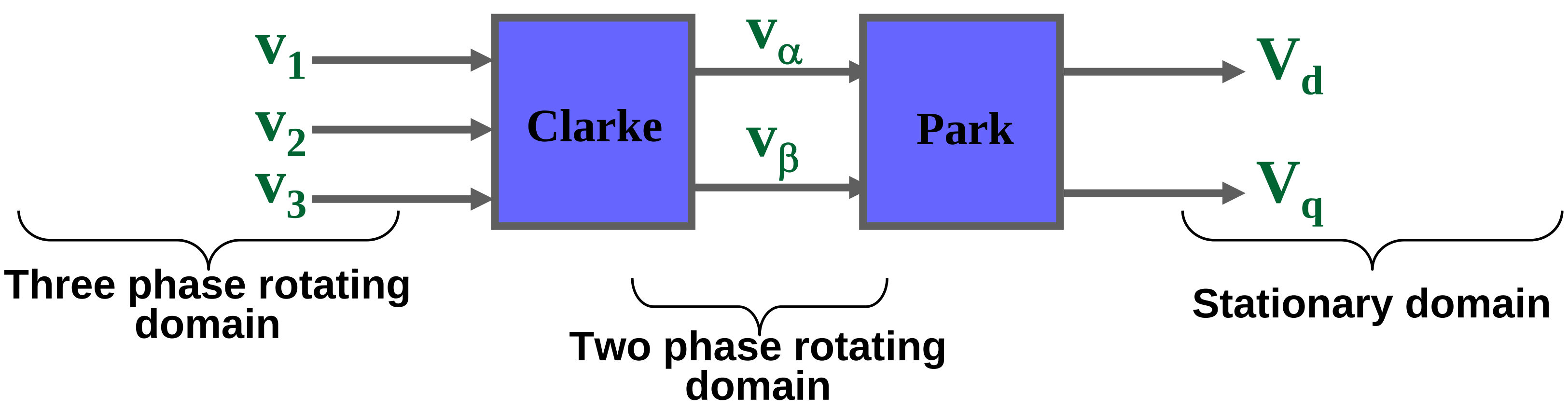
Park’s Transformation



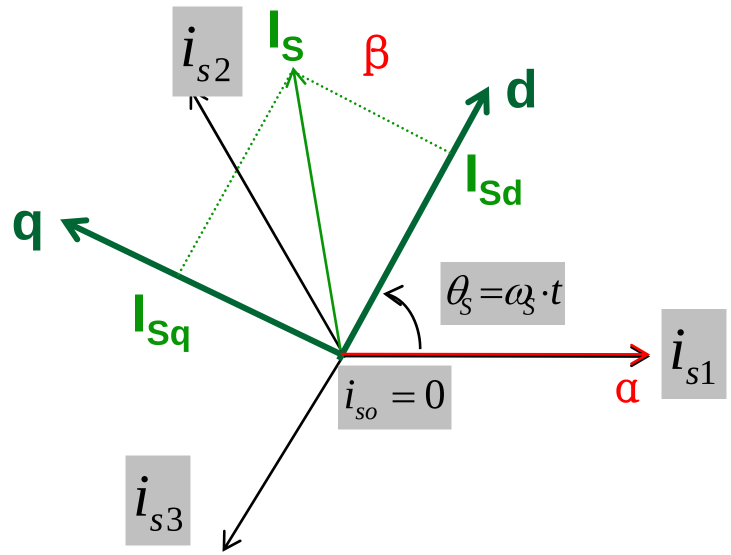
$$i_{sd} = i_\alpha \cdot \cos(\theta) + i_\beta \cdot \sin(\theta)$$
$$i_{sq} = -i_\alpha \cdot \sin(\theta) + i_\beta \cdot \cos(\theta)$$

$$i_\alpha = i_{sd} \cdot \cos(\theta) - i_{sq} \cdot \sin(\theta)$$
$$i_\beta = i_{sd} \cdot \sin(\theta) + i_q \cdot \cos(\theta)$$

Transform summary

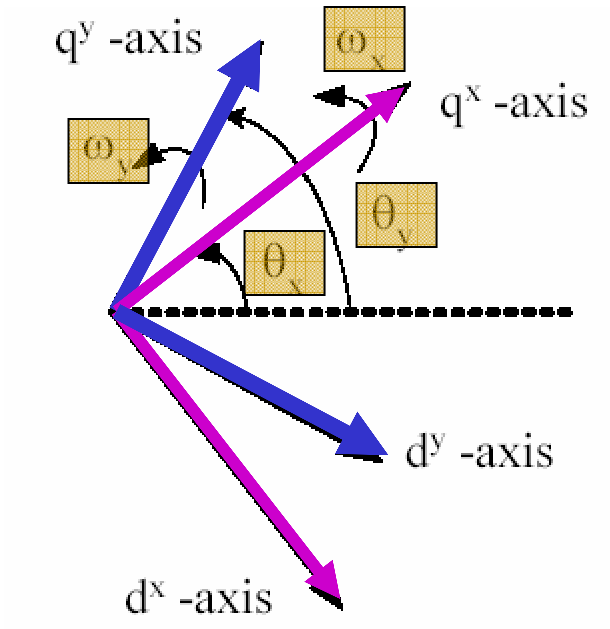


- Stator phase current example:
- I_s is moving at θ_s and its PARK coordinates are constant in (d,q) rotating frame.
- Can be applied on any three-phase balanced variables (flux…)



Transformations Between Reference Frames

- In order to reduce the complexity of some derivations, it is necessary to transform the variables from one reference frame (d^x, q^x) to another one (d^y, q^y) .



Transformations Between Reference Frames

- In this regard, we can rewrite the transformation equation as

$$\left[f_{qd0s}^{y}\right] = T_{qd0s}^{x \to y} \cdot \left[f_{qd0s}\right]$$

- But we have

$$\left[f_{qd0s}^{x}\right] = T_{qd0s}^{x} \cdot \left[f_{abcs}\right]$$

- we get

$$\left[f_{qd0s}^{y}\right] = T_{qd0s}^{x \to y} \cdot T_{qd0s}^{x} \cdot \left[f_{abcs}\right]$$

Transformations Between Reference Frames

- In another way, we can find out that

$$\left[f_{qd0s}^{y}\right]=T_{qd0s}^{y}\cdot\left[f_{abcs}\right]$$

- we obtain

$$T_{qd0s}^{x\to y}=T_{qd0s}^{y}\cdot T_{qd0s}^{x}{}^{-1}$$

- Then, the desired transformation can be expressed by the following matrix:

$$T_{qd0s}^{x\to y}=\begin{bmatrix}\cos(\theta_y-\theta_x) & -\sin(\theta_y-\theta_x) & 0\\ \sin(\theta_y-\theta_x) & \cos(\theta_y-\theta_x) & 0\\ 1 & 1 & 1\end{bmatrix}$$

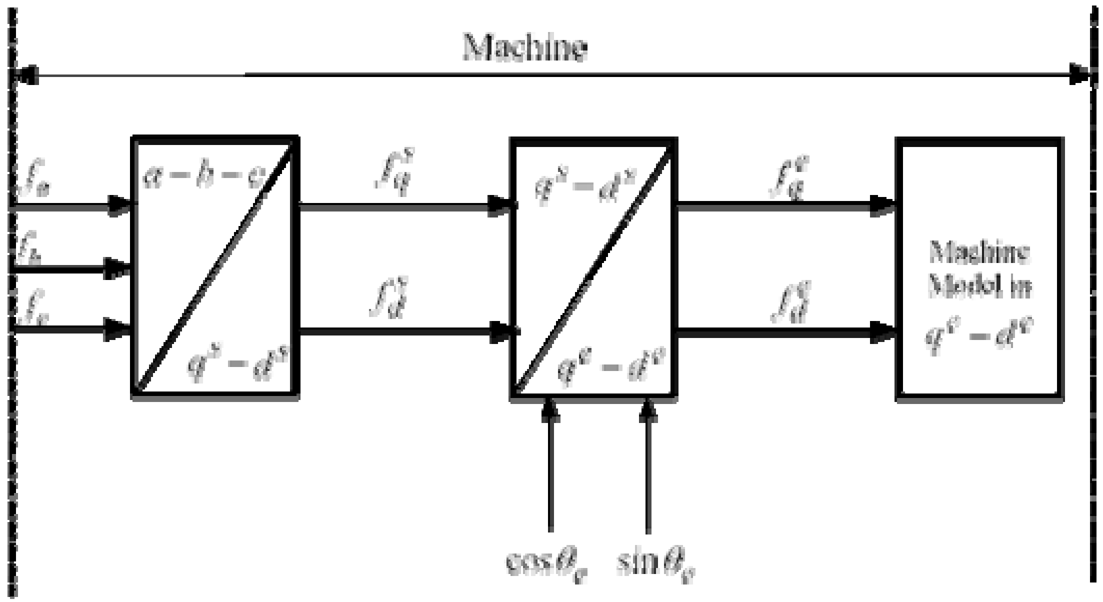
Field Oriented Control (FOC) Transformations

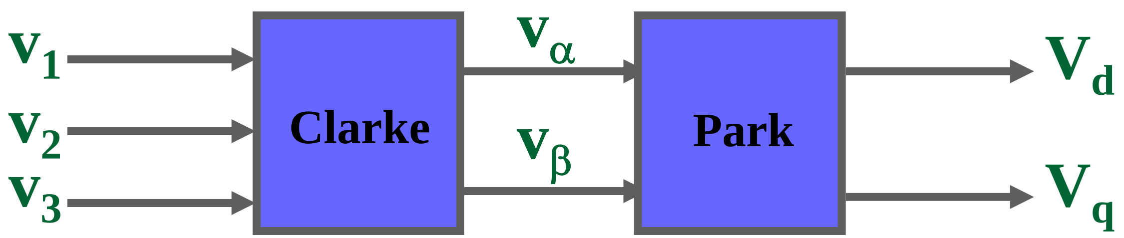
- In the case of FOC of electric machines, control methods are performed in a two-phase reference frame fixed to the rotor (q^r-d^r) or fixed to the excitation reference frame (q^e-d^e).

- We want to transform all the variables from the three-phase a-b-c system to the two-phase stationary reference frame and then retransform these variables from the stationary reference frame to a rotary reference frame with arbitrary angular velocity of ω.

Field Oriented Control (FOC) Transformations

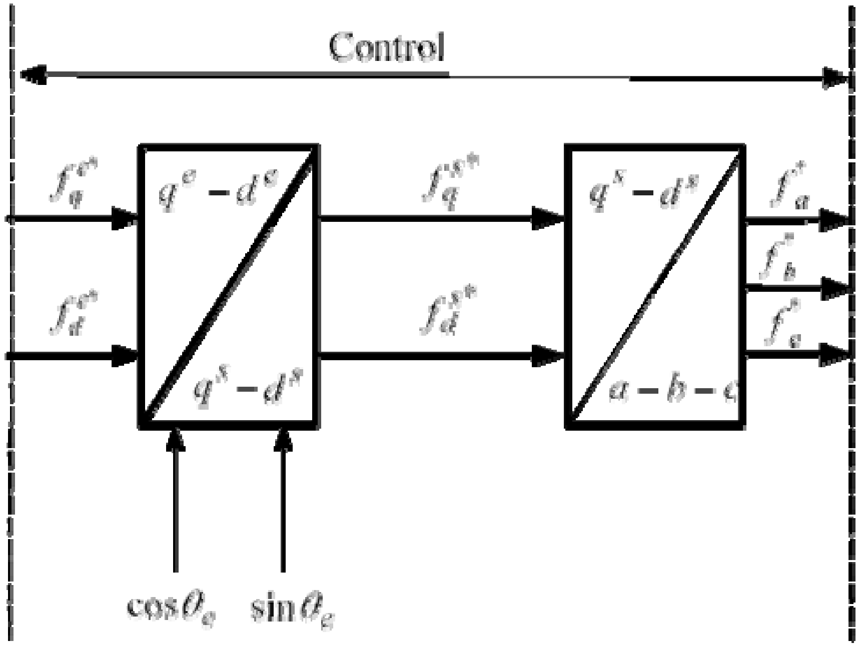
- These transformations are usually cascaded, the block diagram of this procedure is shown:





Field Oriented Control (FOC) Transformations

- In the vector control method, after applying field- oriented control it is necessary to transform variables to stationary a-b-c system.
- This can be achieved by taking the inverse transformation.



Implementing Clarke's Transformation

- Clarke's Transformation is to transfer the three-phase stationary parameters from a-b-c system to the two-phase stationary reference frame.
- It is assumed that the system is balanced

$$f_a + f_b + f_c = 0$$

we have

$$f_\alpha = \frac{2}{3} f_a - \frac{1}{3} f_b - \frac{1}{3} f_c$$

$$f_\beta = \frac{1}{\sqrt{3}} (f_b - f_c)$$

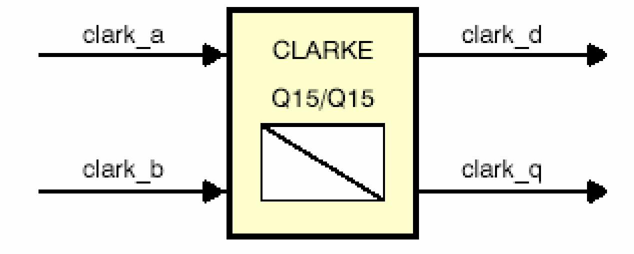
Then we get:

$$f_\alpha = f_a$$

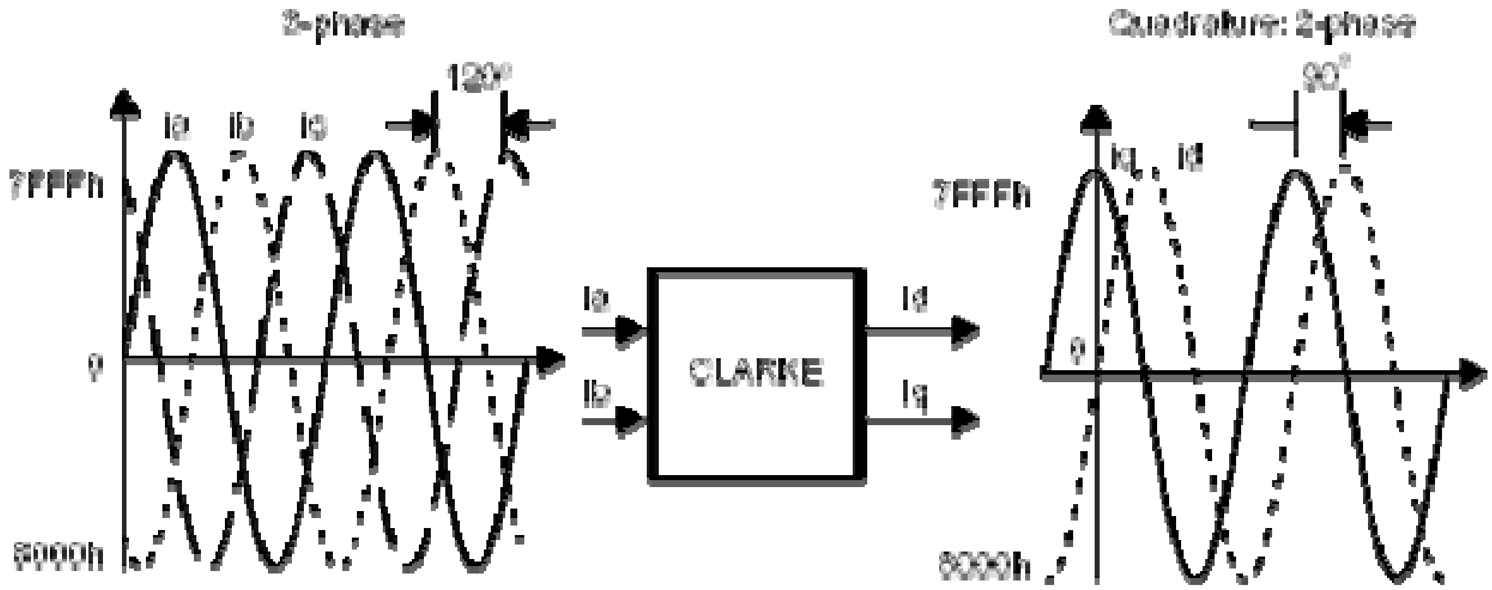
$$f_\beta = \frac{1}{\sqrt{3}} (f_a + 2 f_b)$$

Clarke's Transformation in DSP

- Clarke's Transformation Converts balanced three phase quantities into balanced two phase quadrature quantities.



Clarke's Transformation in DSP

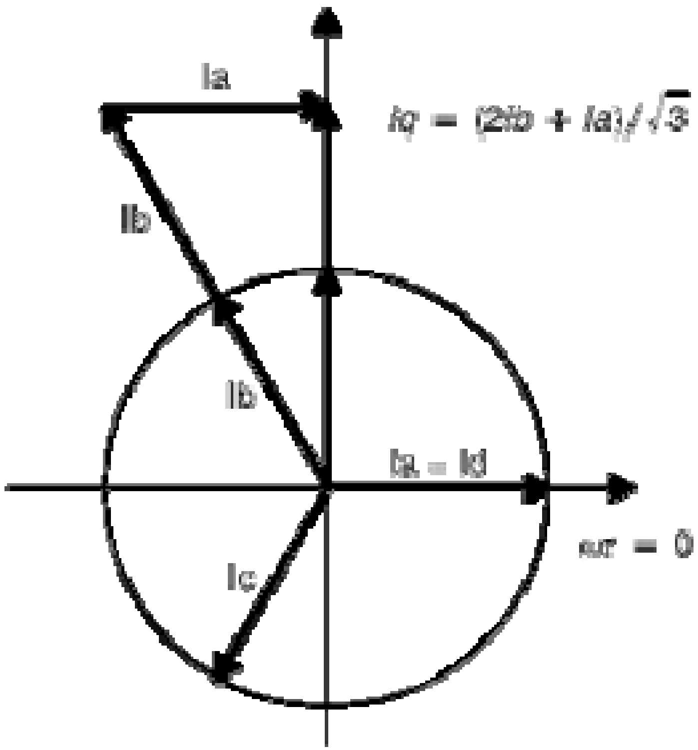


SUN Dan
College of Electrical Engineering, Zhejiang University

Clarke's Transformation in DSP

The instantaneous input and the output quantities are defined by the following equations:

$$ia = I \times \sin(\omega t)$$
$$ib = I \times \sin(\omega t + 2\pi/3)$$
$$ic = I \times \sin(\omega t - 2\pi/3)$$
$$\begin{cases} id = I \times \sin(\omega t) \\ iq = I \times \sin(\omega t + \pi/2) \end{cases}$$



Clarke's Transformation in DSP

- To enjoy better resolution of the variables in fixed point DSP, we transfer all variables to the Q15-based format.
- With this consideration, the maximum value of inputs and outputs can be ($2^{15}-1$) or 7FFFh in hexadecimal.
- In this base, the variables can vary in the range 8000h-7FFFh.

- Then 1/ √ 3 is represented by

```
LDP     #sqrt3inv            ;sqrt3inv=(1/sqrt(3))
                             ;=0.577350269
SPLK    #018918,sqrt3inv     ;1/sqrt(3) (Q15)
                             ;=0.577350269*2^15
```

Clarke's Transformation in DSP

- Clarke's transformation is implemented as follows:

```
SETC    SXM              ;Sign extension mode on
LDP     #clark_a         ;clark_alfa = clark_a
LACC    clark_a          ;ACC = clark_a
SACL    clark_alfa       ;clark_d = clark_a
                         ;clark_beta=(2*clark_b+clark_a)/
                         ;sqrt(3)
SFR                      ;ACC = clark_a/2
ADD     clark_b          ;ACC = clark_a/2 + clark_b
SACL    clk_temp         ;clk_temp = clark_a/2 + clark_b
LT      clk_temp         ;TREG = clark_a/2 + clark_b
MPY     sqrt3inv         ;PREG=(clark_a/2+clark_b)*
                         ;(1/sqrt(3))
PAC                      ;ACC=(clark_a/2+clark_b)*
                         ;(1/sqrt(3))
SFL                      ;ACC=(clark_a+clark_b*2)*
                         ;(1/sqrt(3))
SACH    clark_beta       ;clark_beta=(clark_a+clark_b*2
                         ;(1/sqrt(3))
SPM     0                ;SPM reset
RET
```

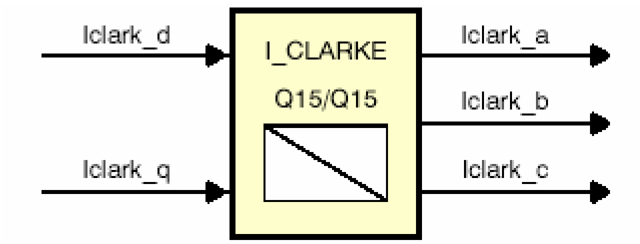
Implementing Inverse Clarke's Transformation

- The inverse Clarke functions Converts balanced two phase quadrature quantities into balanced three phase quantities.

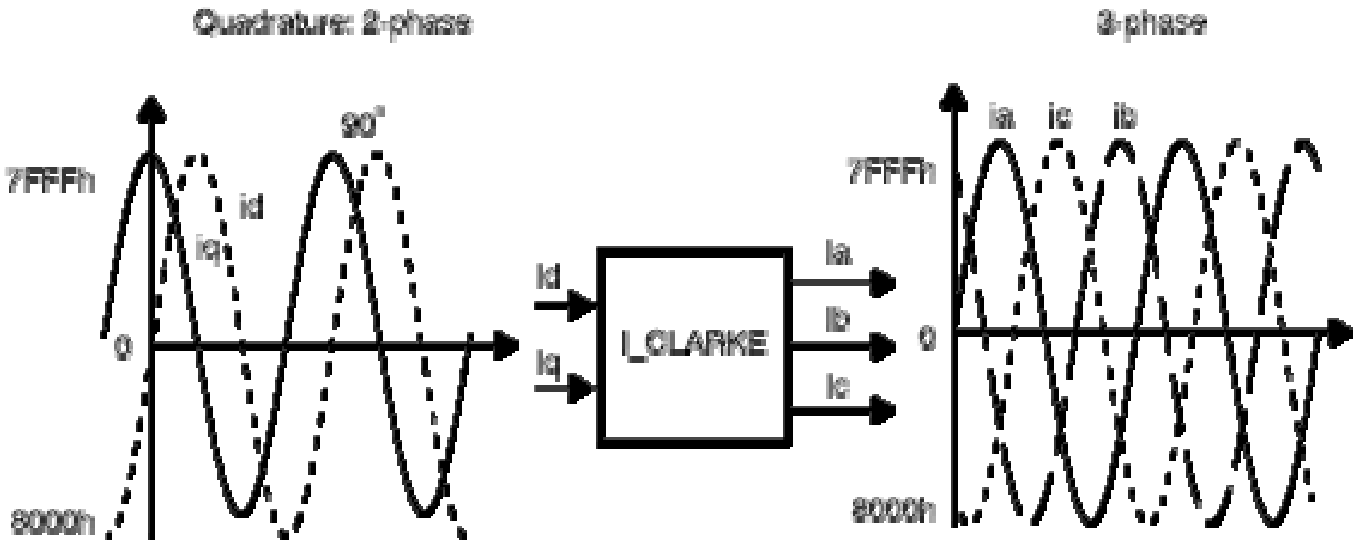
$$f_a = f_\alpha$$

$$f_b = \frac{-f_\alpha + \sqrt{3} * f_\beta}{2}$$

$$f_c = \frac{-f_\alpha - \sqrt{3} * f_\beta}{2}$$

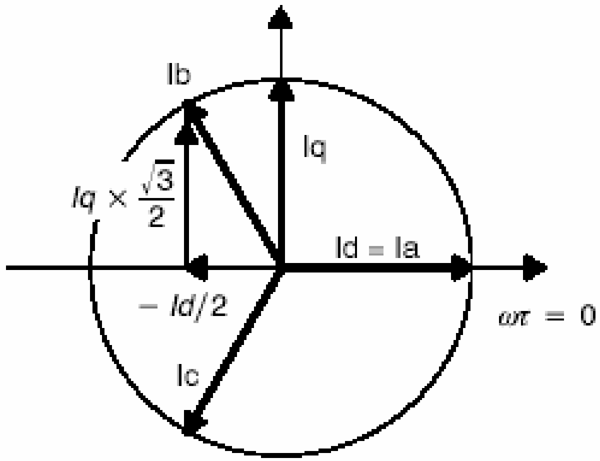


Implementing Inverse Clarke’s Transformation



$$\begin{cases} id = I \times \sin(\omega t) \\ iq = I \times \sin(\omega t + \pi/2) \end{cases}$$

$$\begin{cases} ia = I \times \sin(\omega t) \\ ib = I \times \sin(\omega t + 2\pi/3) \\ ic = I \times \sin(\omega t - 2\pi/3) \end{cases}$$



Implementing Inverse Clarke's Transformation

```
I_CLARKE_INIT:
        LDP     #half_sqrt3             ;Variables data page
        SPLK    #28377,half_sqrt3       ;Set constant sqrt(3)*0.5 in Q15
                                        ;format
RET
I_CLARKE:
        LDP     #f_clark_alpha          ;Variables data page
        SPM     1                       ;SPM set for Q15 multiplication
        SETC    SXM                     ;Sign extension mode on
        LACC    f_clark_alpha           ;ACC = f_alpha
        SACL    f_clark_a               ;f_a = f_alpha
        LT      f_clark_beta            ;TREG = f_clark_beta
        MPY     half_sqrt3              ;PREG=f_clark_beta * half_sqrt3
        PAC                             ;ACC= f_clark_beta * half_sqrt3
SUB     f_clark_alpha,15                ;ACC=f_beta*half_sqrt3-f_alpha/2
        SACH    f_clark_b
        PAC                             ;ACC high = f_beta*half_sqrt3
        NEG                             ;ACC high = - f_beta*half_sqrt3
        SUB     f_clark_alpha,15        ;ACC high=-f_beta*half_sqrt3-
                                        ;f_alpha/2
        SACH    f_clark_c               ;f_c = - f_beta * half_sqrt3 -
                                        ;f_alpha/2
        SPM     0                       ;SPM reset
        CLRC    SXM                     ;Sign extension mode off
RET
```

Calculation of Sine/Cosine

- To implement the Park and the inverse Park transforms, the sine and cosine functions need to be implemented.
- This method realizes the sine/cosine functions with a look-up table of 256 values for 360° of sine and cosine functions.
- The method includes linear interpolation with a fixed step table to provide a minimum harmonic distortion.
- This table is loaded in program memory.
- The sine value is presented in Q15 format with the range of $-1<value<1$.

Calculation of Sine/Cosine

- The first few rows of the look-up sine table are presented as follows:

```
;SINVALUE          ;      Index    Angle    Sin(Angle)
----------------          --------  --------  ------------
SINTAB_360
.word      0       ;      0        0        0.0000
.word      804     ;      1        1.41     0.0245
.word      1608    ;      2        2.81     0.0491
.word      2410    ;      3        4.22     0.0736
.word      3212    ;      4        5.63     0.0980
```

Calculation of Sine/Cosine

- The following assembly code is written to read values of sine from the sine Table in Q15 format:

```
LACC    theta_p, 9       ;Input angle in Q15 format and
                         ;left shifted by 15
SACH    t_ptr            ;Save high ACC to t_ptr (table
                         ;pointer)
LACC    #SINTAB_360
ADD     t_ptr
TBLR    sin_theta        ;sin_theta = Sin(theta_p) in Q15
```

Calculation of Sine/Cosine

- Note that 0 < theta_p < 7FFFh (i.e., equivalent to 0 < theta_p < 360 deg).
- The TBLR instruction transfers a word from a location in program memory to a data-memory location specified by the instruction.
- The program-memory address is defined by the low-order 16 bits of the accumulator.
- For this operation, a read from program memory is performed, followed by a write to data memory.

Calculation of Sine/Cosine

- To calculate the cosine values from the sine Table in Q15 format, we write the following code:

```
LACC    theta_p
ADD     #8192           ;add 90 deg, cos(A)=sin(A+900)
AND     #07FFFh         ;Force positive wrap-around
SACL    GPR0_park       ;here 90 deg = 7FFFh/4
LACC    GPR0_park,9
SACH    t_ptr
LACC    #SINTAB_360
```

Implementation of Park's Transformation

- With field-oriented control of motors, it is necessary to transform variables,
  - from a-b-c system to two-phase stationary reference frame, *qs-ds*,
  - and from two-phase stationary reference frame *qs-ds* to arbitrary rotating reference frame with angular velocity of ω (*q-d* reference frame).

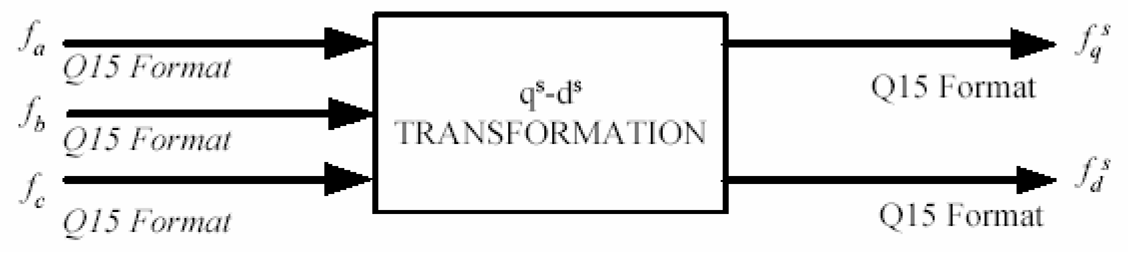
- The first transformation is dual to Clarke's transformation
  - but the q^s axis is in the direction of α -axis, and d^s axis is in negative direction of β -axis.

Implementation of Park's Transformation a-b-c to q^s-d^s

- This transformation transfers the three-phase stationary parameters from an a-b-c system to a two-phase orthogonal stationary reference frame.

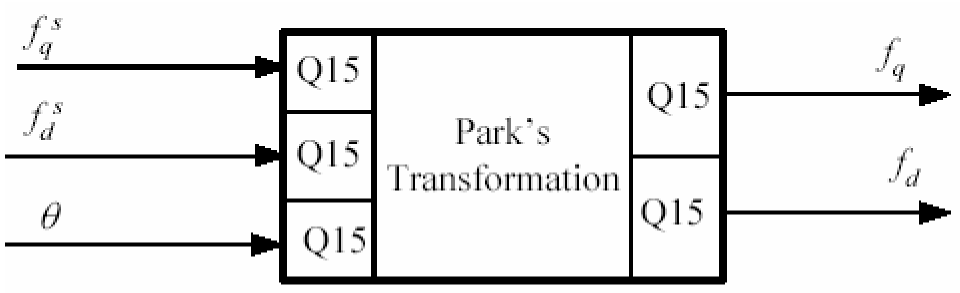
$$f_q^s = f_a$$

$$f_d^s = -\frac{1}{\sqrt{3}}(2f_b + f_a)$$



Implementation of Park's Transformation q^s-d^s to q-d

- This transformation converts vectors in a balanced two-phase orthogonal stationary system into an orthogonal rotary reference frame.



$$f_q = \cos\theta . f_q^s - \sin\theta \cdot f_d^s$$

$$f_d = \sin\theta . f_q^s + \cos\theta \cdot f_d^s$$

Implementation of Park's Transformation q^s-d^s to q-d

- In this transformation, it is necessary to calculate sin θ and cos θ, where the method to calculate them was presented in a previous section.
- All the input and outputs are in the Q15 format and in the range of 8000h-7FFFh

Implementation of Park’s Transformation q^s-d^s to q-d

- The following code is written to implement Park’s transformation:

```
SPM     1             ;SPM set for Q15 multiplication
ZAC                   ;Reset accumulator
LT      f_q_s         ;TREG = f_q_s
MPY     sin_theta     ;PREG = f_q_s * sin(theta)
LTA     f_d           ;ACC = f_q_s * sin(theta) and
                      ;TREG =f_q_s
MPY     cos_theta     ;PREG = f_d_s* cos_teta
MPYA    sin_theta     ;ACC=f_q_s*sin_teta+f_d_s*
                      ;cos_teta andPREG=f_q_s*sin_teta
SACH    park_D        ;f_d =f_q_s * cos_teta + f_d_s*
                      ;sin(theta)
```

Implementation of Park's Transformation q^s-d^s to q-d

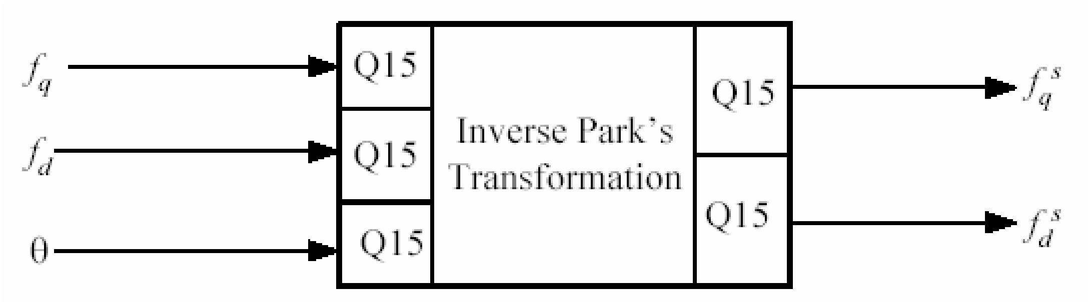
```
LACC    #0            ;Clear ACC
LT      f_d_s         ;TREG = f_d_s
MPYS    cos_theta     ;ACC=- f_d_s* *sin(theta) and
                      ;PREG = f_q_s * cos(theta)
APAC                  ;ACC=- f_d_s*sin(theta) +f_q_s*
                      ;cos(theta)
SACH    f_q           ;fq = -f_d_s*sin(theta) +f_q_s*
                      ;cos(theta)
SPM     0             ;SPM reset
RET
```

Implementation of Park's Transformation q-d to q^s-d^s

- This transformation projects vectors in an orthogonal rotating reference frame into a two-phase orthogonal stationary frame.

$$f_q^s = \cos\theta \cdot f_q + \sin\theta \cdot f_d$$

$$f_d^s = -\sin\theta \cdot f_q + \cos\theta \cdot d_d$$



Implementation of Park's Transformation q-d to q^s-d^s

- The following code is written to implement this transformation:

```
SPM     1           ;SPM set for Q15 multiplication
ZAC                 ;Reset accumulator
LT      f_q         ;TREG = fq
MPY     cos_theta           ;PREG = fq * cos(theta)
LTA     f_d                 ;ACC=fq*cos(theta) and TREG =fd
MPY     sin_theta           ;PREG = fd * sin(theta)
MPYA    sin_theta           ;ACC=fq*cos(theta)+fd*sin(theta) and
                            ;PREG=fd*sin(theta)
SACH    f_q_s               ;fd=fq*cos(theta)+fd*sin(theta)
LACC    #0                  ;Clear ACC
LT      f_d                 ;TREG = fd
MPYS    cos_theta ;ACC = -fd*sin_theta and PREG = fd*cos_theta
APAC
SACH    f_d_s
SPM     0           ;SPM reset
RET
```

Implementation of Park's Transformation q^s-d^s to a-b-c

- This transformation transforms the variables from the stationary two-phase frame to the stationary a-b-c system.

- This system is also dual to the inverse Clarke transformation where the q^s-axis is in the direction of the α-axis and the d^s-axis is in the negative direction of β-axis.

$$f_a = f_q^s$$

$$f_b = \frac{-f_q^s - \sqrt{3} f_d^s}{2}$$

$$f_c = \frac{-f_q^s - \sqrt{3} f_d^s}{2}$$

SUN Dan
College of Electrical Engineering, Zhejiang University

Conclusion

- With FOC of synchronous and induction machines, it is desirable to reduce the complexity of the electric machine voltage equations.
- The transformation of machine variables to an orthogonal reference frame is beneficial for this purpose.
- Park's and Clarke's transformations, two revolutions in the field of electrical machines, were studied in depth in this chapter.
- These transformations and their inverses were implemented on the fixed point LF2407 DSP.

Reading materials

- Bpra047 - Sine, Cosine on the TMS320C2xx
- Bpra048 - Clarke & Park Transforms on the TMS320C2xx
- Spru485a - *Digital Motor Control Software Library*
  - CLARKE,
  - PARK,
  - I_CLARKE,
  - I_PARK,