Sensorless FOC for PMSM using Reduced Order Luenberger Observer

AN2590

Introduction

Current industry trends suggest that the Permanent Magnet Synchronous Motor (PMSM) is the first preference for motor control application designers. It has strengths, such as high power density, fast dynamic response, and high efficiency in comparison with other motors in the same category. Couple this with decreased manufacturing costs, and improved magnetic properties, the PMSM is a good recommendation for large-scale product implementation.

Microchip Technology Inc. produces a wide range of microcontrollers for enabling efficient, robust and versatile control of all types of motors, along with reference designs of the necessary tool sets. This results in a fast learning curve, and a shortened development cycle for new products.

Table of Contents

1. Field Oriented Control (FOC)

In case of the PMSM, the rotor field speed must be equal to the stator (armature) field speed (i.e., synchronous). The loss of synchronization between the rotor and stator fields causes the motor to stall.

FOC represents the method by which one of the fluxes (rotor, stator or air gap) is considered as a basis for creating a reference frame for one of the other fluxes, with the purpose of decoupling the torque and flux-producing components of the stator current. This decoupling assures the ease of control for complex three-phase motors in the same manner as DC motors with separate excitation. This means the armature current is responsible for the torque generation, and the excitation current is responsible for the flux generation. In this application note, the rotor flux is considered as a reference frame for the stator and air gap flux.

The particularity of the FOC in the case of a Surface Mounted Permanent Magnet type PMSM (SPM) is that the d-axis current reference of the stator i_{dref} (corresponding to the armature reaction flux on d-axis) is set to zero. The magnets in the rotor produce the rotor flux linkage, Λ_m, unlike AC Induction Motor (ACIM), which needs a constant reference value, i_{dref}, for the magnetizing current, thereby producing the rotor flux linkage. The d-axis current reference for Interior Permanent Magnet type PMSM (IPM) motors is explained later in this section.

The air gap flux is equal to the sum of the flux linkage of the rotor. This is generated by the permanent magnets and the armature reaction flux linkage generated by the stator current. For the constant torque mode in FOC, the d-axis air gap flux is solely equal to Λ_m, and the d-axis armature reaction flux is zero.

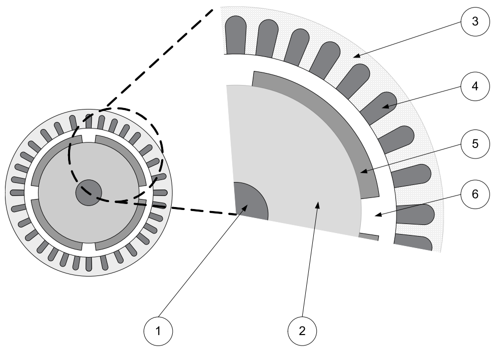
Conversely, in constant power operation, the flux generating component of the stator current, negative i_d, is used for air gap field weakening to achieve higher speed.

In sensorless control, where no position or speed sensors are needed, the challenge is to implement a robust speed estimator that is able to reject perturbations such as temperature, switching noise, electromagnetic noise, and so on. Sensorless control is usually required when applications are cost sensitive, where moving parts are not allowed. For example, when position sensors are used, or when the motor is operated in an electrically hostile environment. However, requests for precision control, especially at low speeds, should not be considered a critical matter for the given application.

The position and speed estimation is based on the mathematical model of the motor. Therefore, the closer the model is to the real hardware, the better the estimator will perform. The PMSM mathematical modeling depends on its topology, differentiating mainly two types: surface-mounted motor and interior permanent magnet (IPM) motor. Each motor type has its own advantages and disadvantages with respect to the application needs.

The proposed control scheme has been developed for both surface-mounted and interior permanent magnet synchronous motors. The surface mounted motor is shown in the following figure, which has the advantage of low torque ripple and is lower in price when compared with an interior PMSM. The air gap flux for the motor type considered is smooth therefore, the inductance value of the stator, $L_d = L_q$ (non salient PMSM), and the Back Electromagnetic Force (BEMF) is sinusoidal.

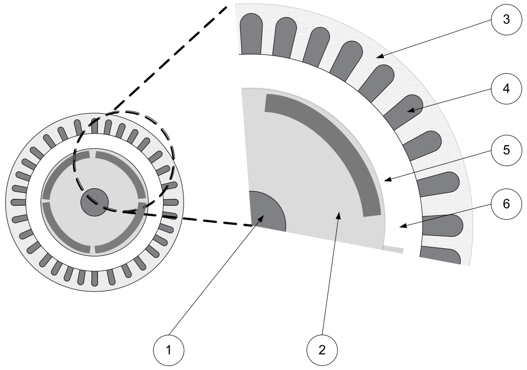
Figure 1-1. Surface Mounted PM PMSM Transversal Section



Motor's Transversal Section
1. Rotor shaft.
2. Rotor core.
3. Armature (stator).
4. Armature slots with armature windings
5. Rotor's permanent magnets.
6. Air gap.

The IPM motor shown in the following figure, exhibits additional reluctance torque in addition to the permanent magnet torque. It provides higher torque at a given operating current compared to the SPM type. In an Interior PM motor, the reluctance of the magnetic flux path varies according to the rotor position. This magnetic saliency results in the variation of the inductance at the motor terminal according to the rotor position. Therefore, the effective flux length of L_d and L_q are different, $L_d \neq L_q$ (salient), because of PM in the flux path. Therefore an IPM motor has inductance saliency, and it utilizes both reluctance torque and permanent magnet torque.

Figure 1-2. IPM Transversal Section



Motor's Transversal Section
1. Rotor shaft.
2. Rotor core.
3. Armature (stator).
4. Armature slots with armature windings
5. Rotor's permanent magnets.
6. Air gap.

The torque generation of IPM motors can be expressed, as shown in the following equation.

Equation 1-1. Torque Generation of IPM Motors

$$T = \frac{3}{2}p\left(\Lambda_m + \left(L_d - L_q\right) i_d\right)i_q$$

Where,

p is the number of pole pairs

L_d, L_q are the d-axis and q-axis inductances respectively

i_d, i_q are the d-axis and q-axis currents respectively

Λ_m is the magnetic flux linkage

The generated torque consists of both permanent magnet torque and the reluctance torque components. The PM torque is produced by the interaction between PM and torque current of stator windings. The reluctance torque is produced by the force acting on the magnetic material that tends to align with the main flux to minimize reluctance. Reluctance torque is independent of permanent magnet excitation.

In case of SPM motors, $L_d = L_q$ and the above equation simplifies to

Equation 1-2. Torque Generation of SPM Motors

$$T = \frac{3}{2} p \Lambda_m i_q$$

Therefore, the generated torque consists of only the permanent magnet torque component.

1.1 Direct Axis Current Reference

In IPM motors, the quadrature (q-axis) current components in phase with corresponding BEMF voltages, produce only magnetic torque but do not contribute in producing reluctance torque. The direct axis (d-axis) current components 90 degrees out of phase with BEMF voltages, produce reluctance torque in conjunction with quadrature axis currents. Therefore, there exists an infinite number of current vectors providing the same amount of torque. For the most efficient operation, the current vector with the lowest possible magnitude should be chosen to reduce winding losses.

By imposing a negative i_d, the produced torque can be increased which will be the maximum for the same total current drawn. This is called the Maximum Torque per Ampere (MTPA) control and the i_{dref} is calculated by:

Equation 1-3. Maximum Torque per Ampere (MTPA)

$$i_d = \frac{\Lambda_m}{2\left(L_q - L_d\right)} - \sqrt{\frac{\Lambda_m^2}{4\left(L_q - L_d\right)} + i_q^2}$$

In case of SPM motors, the quadrature axis current components, in phase with the corresponding BEMF voltages, produce the magnetic torque and the d-axis current reference (i_{dref}) is set to zero, (i.e., i_{dref}= 0.)

The motor characteristics enable some simplification of the mathematical model used in the speed and position estimator, while at the same time enabling the efficient use of the FOC. The mathematical model used in the estimator is explained in the later section.

1.2 A Matter of Perspective

One way to understand how FOC (sometimes referred to as vector control) works is to form an image of the coordinate reference transformation process. Visualizing an AC motor operation from the perspective of the stator, a sinusoidal input current applied to the stator can be observed. This time variant signal generates a rotating magnetic flux. The speed of the rotor is a function of the rotating flux vector. From a stationary perspective, the stator currents and the rotating flux vector look like AC quantities.

The spinning rotor is moving at the same speed as the rotating flux vector generated by the stator currents. Viewing the motor from the perspective of the rotor during steady state conditions, the stator currents look like constant values, and the rotating flux vector is stationary.

The objective is to control the stator currents to obtain the desired rotating flux vector components (which cannot be measured directly). With coordinate reference transformation, the stator currents can be controlled like DC values using standard control loops.

1.3 Vector Control Summary

The indirect vector control process can be summarized as follows:

1. The three-phase stator currents are measured. For currents in a motor with balanced three phase windings, these measurements provide values i_a and i_b. i_c is calculated by the following equation: $i_a + i_b + i_c = 0$.

2. The three-phase currents are converted to a stationary two-axis system. This conversion provides the variables i_α and i_β from the measured i_a and i_b and the calculated i_c values. The values i_α and i_β are time-varying quadrature current values as viewed from the perspective of the stator.

3. The stationary two-axis coordinate system is rotated to align with the rotor flux using a transformation angle calculated at the last iteration of the control loop. This conversion provides the i_d and i_q variables from i_α and i_β. The values i_d and i_q are the quadrature currents transformed to the rotating coordinate system. For steady state conditions, i_d and i_q are constant.

4. Error signals are formed using i_d, i_q and reference values for each are shown below.

1. The i_d reference, controls rotor magnetizing flux.
2. The i_q reference, controls the torque output of the motor.
3. The error signals are input to PI controllers.
4. The output of the controllers provide v_d and v_q, which are voltage vector that will be applied to the motor.

5. A new transformation angle is estimated from the position estimation observer using v_α, v_β, i_α and i_β. This new angle guides the FOC algorithm as to where to place the next voltage vector.

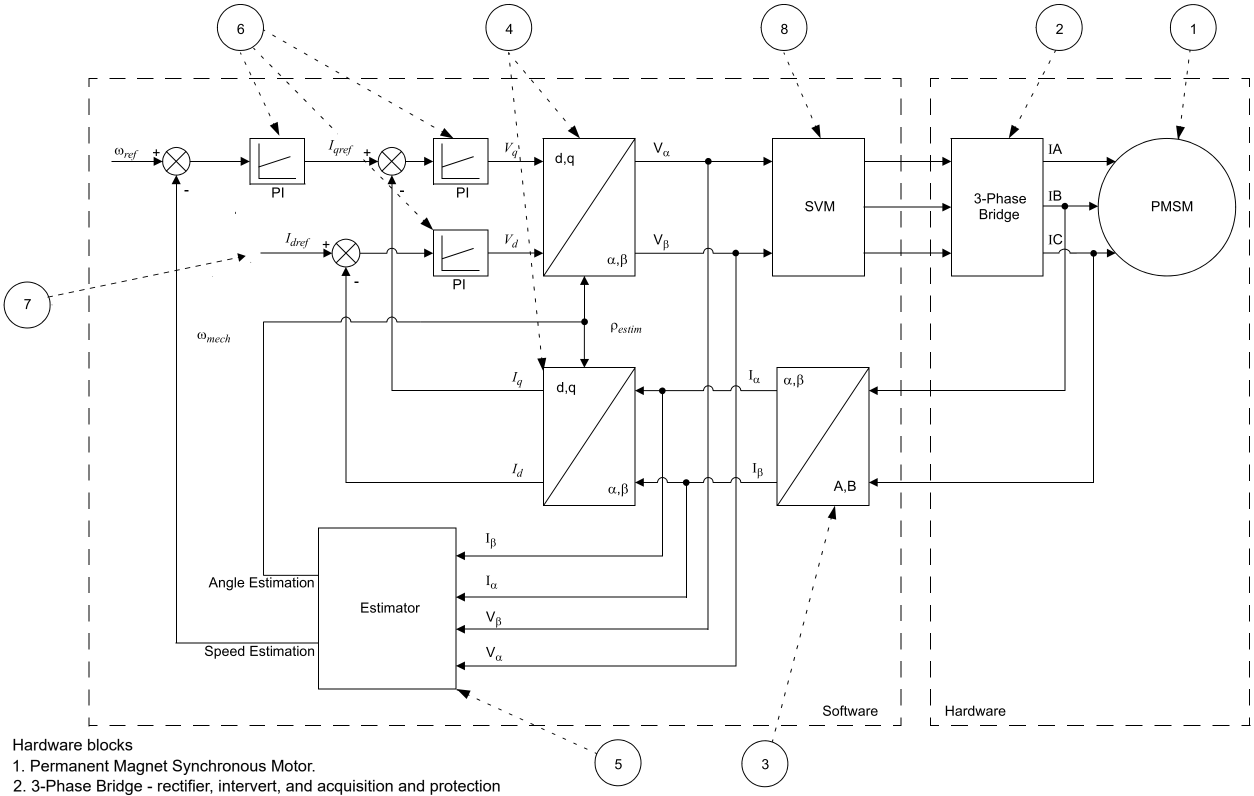
6. The v_d and v_q output values from the PI controllers are rotated back to the stationary reference frame using the new angle. This calculation provides the next quadrature voltage values v_α and v_β.

7. The v_α and $_{v\beta}$ values are transformed back to three-phase values v_a, v_b and v_c. The three-phase voltage values are used to calculate new PWM duty cycle values that generate the desired voltage vector.

The entire process of transforming to a rotating frame, PI iteration, transforming back to stationary frame and generating PWM is illustrated in the following figure.

The next sections of this application note describe these steps in greater detail.

Figure 1-3. FOC Block Diagram



Hardware blocks

1. Permanent Magnet Synchronous Motor.
2. 3-Phase Bridge - rectifier, intervert, and acquisition and protection circuitry software blocks (run by the PIC32 device).
3. Clarke forward transform block.
4. Park forward and inverse transform block.
5. Angle and speed estimator block.
6. Proportional integral controller block.
7. Id current reference.
8. Space vector modulation block.

1.4 Coordinate Transforms

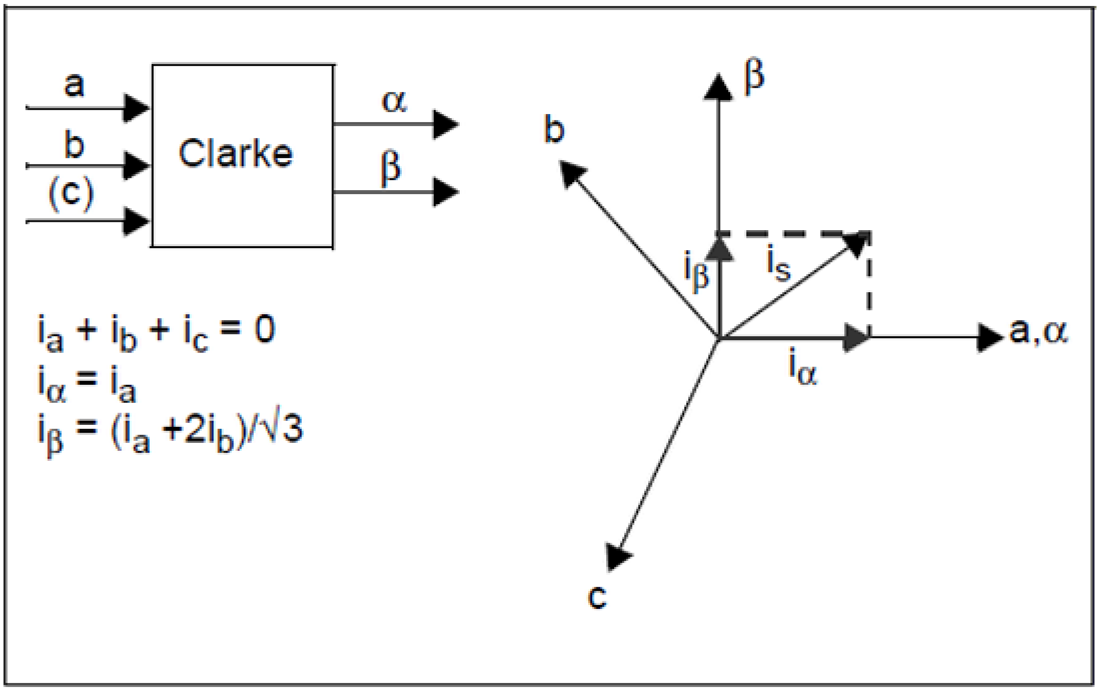
Through a series of coordinate transforms, the user can indirectly determine and control the time invariant values of torque and flux with classic PI control loops.

The process begins by measuring the three-phase motor currents. In practice, the instantaneous sum of the three current values is zero. Therefore, by measuring only two of the three currents, one can determine the third. Due to this fact, hardware cost can be reduced by the expense of the third current sensor.

1.4.1 Clarke Transform

The first coordinate transform, called the Clarke Transform, moves a three-axis, two-dimensional coordinate system, referenced to the stator, onto a two-axis system, keeping the same reference (see the following figure, where i_a, i_b and i_c are the individual phase currents).

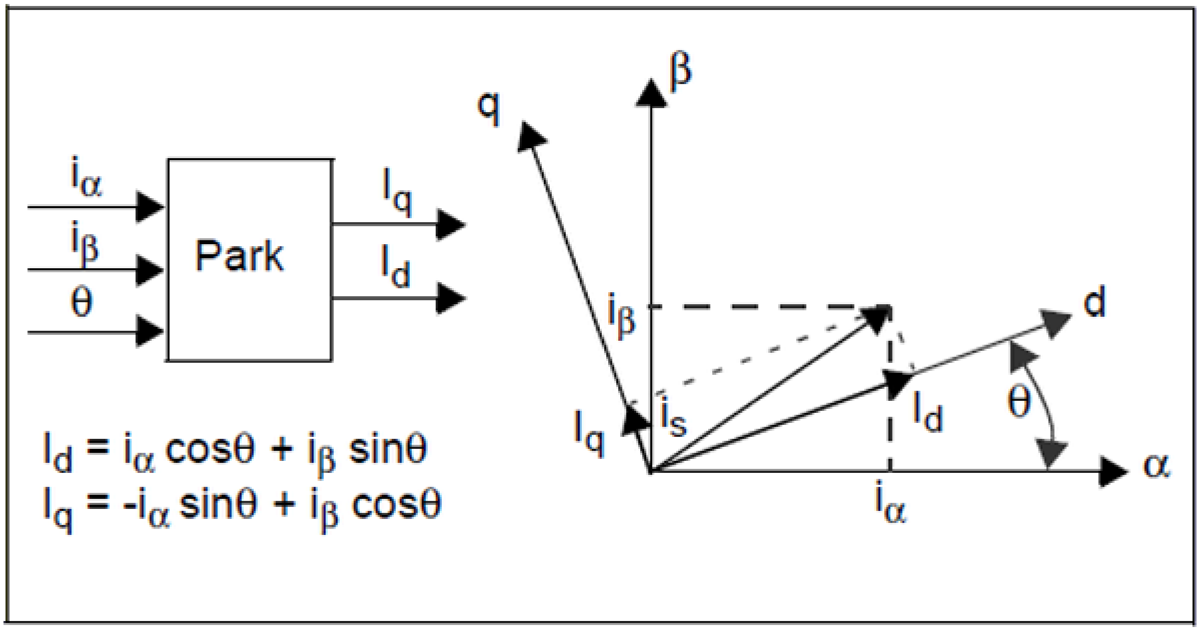
Figure 1-4. Clarke Transform



1.4.2 Park Transform

The stator current is represented on a two-axis orthogonal system with the axis called α-β. The next step is to transform this into another two-axis system that is rotating with the rotor flux. This transformation uses the Park Transform, as illustrated in the following figure. This two-axis rotating coordinate system is called the d-q axis. θ represents the rotor angle.

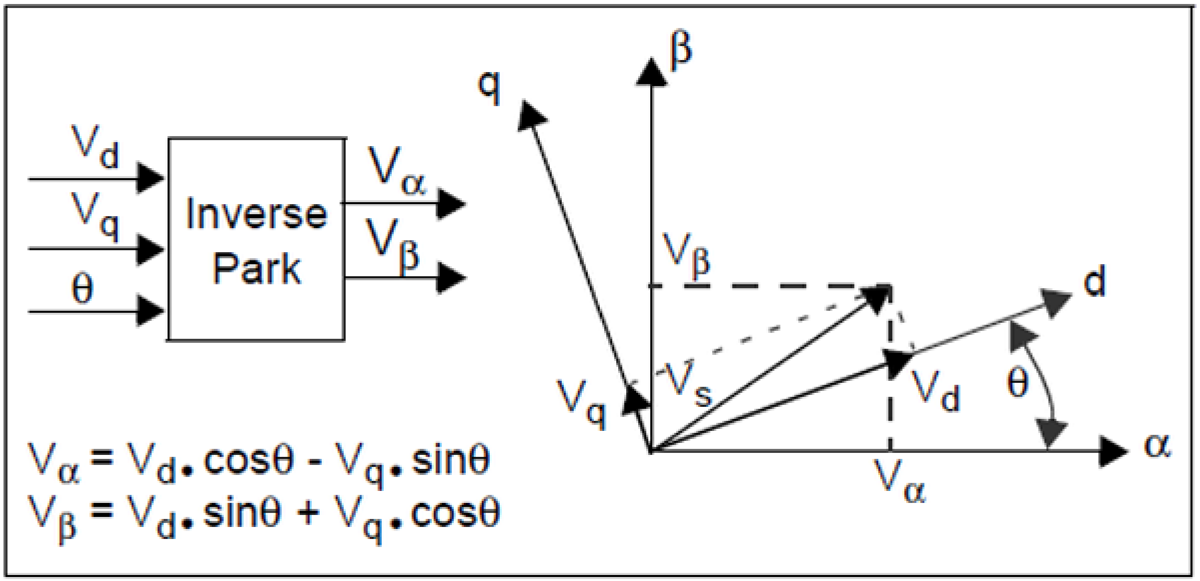
Figure 1-5. Park Transform


i_α
i_β
θ
Park
I_q
I_d
$I_d = i_\alpha \cos\theta + i_\beta \sin\theta$
$I_q = -i_\alpha \sin\theta + i_\beta \cos\theta$
q
β
i_β
d
I_q
i_s
I_d
θ
α
i_α

1.4.3 Inverse Park

After the PI iteration, there are two voltage component vectors in the rotating d-q axis. The user will need to go through complementary inverse transforms to get back to the three-phase motor voltage. First, transform from the two-axis rotating d-q frame to the two-axis stationary frame α-β. This transformation uses the Inverse Park Transform, as illustrated in the following figure.

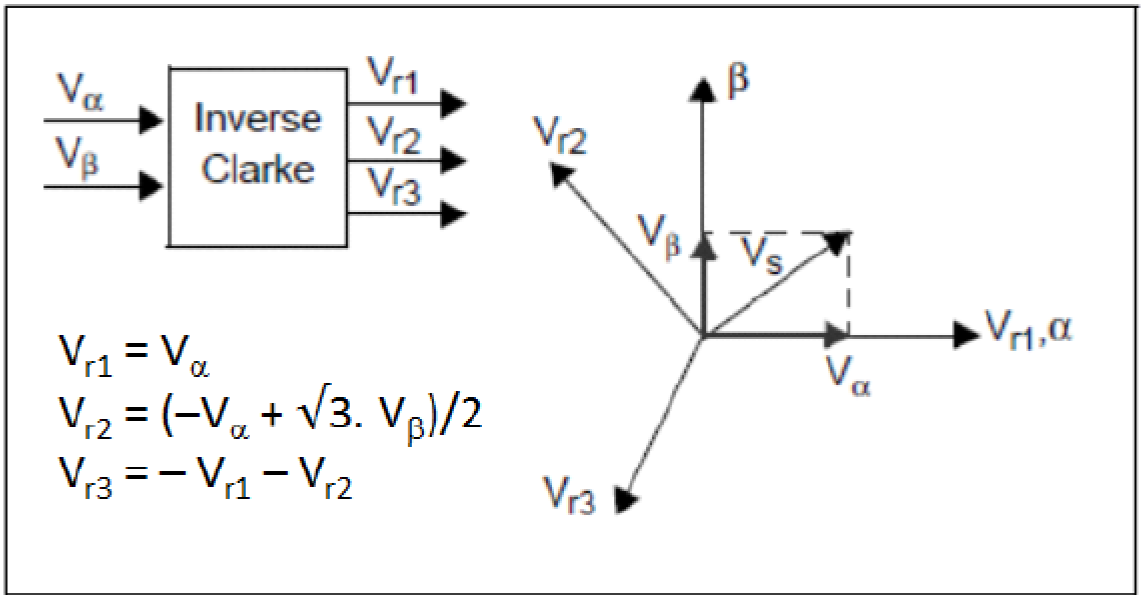
Figure 1-6. Inverse Park Transform



1.4.4 Inverse Clarke

The next step is to transform from the stationary two-axis α-β frame to the stationary three-axis, Three-phase reference frame of the stator. Mathematically, this transformation is accomplished with the Inverse Clark Transform, as illustrated in the following figure.

Figure 1-7. Inverse Clark Transform



1.5 Sensorless Position Estimation

1.5.1 Luenberger Observer

The *Luenberger Observer* is a general method used to find out the internal state of a linear system, when the inputs and the outputs are known.

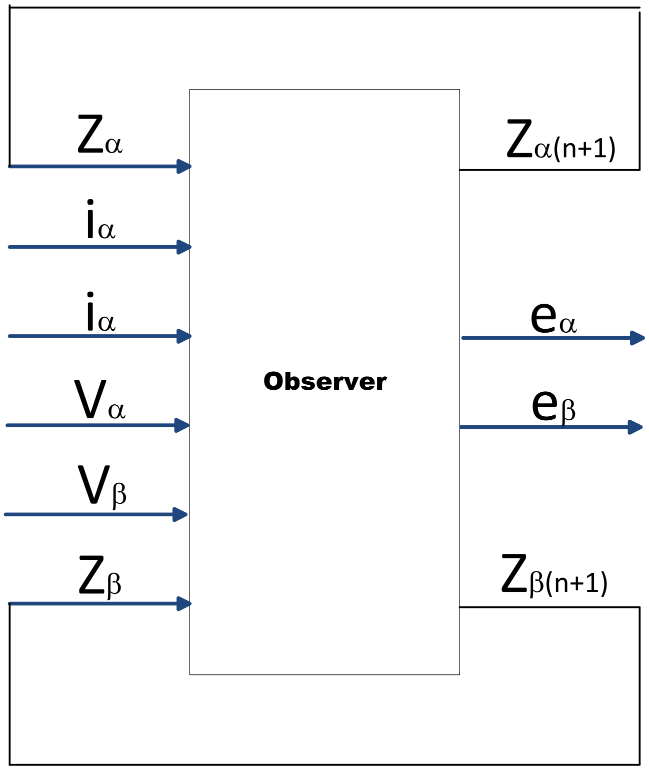
Here the implementation of Reduced Order Luenberger Observer is described for the BEMF. The BEMF vector position is found with an `arctan()` operation from the BEMF components, which are derived from the Luenberger observer internal state variables. Now the flux vector position is found since its position lags BEMF by 90°.

The speed is obtained from the position derivative. This needs to be heavily filtered to give good results. A fourth order filter can be used, composed by a first order FIR filter (moving average) followed by three equal first order IIR filters.

The Reduced Order Luenberger Observer provides good results for both steady state and dynamic operating conditions.

1.5.2 Reduced Order Luenberger Observer

Figure 1-8. Reduced Order Luenberger Observer



The discrete implementation of the Reduced Order Luenberger Observer is shown in the previous figure, and is represented by the following equations.

Equation 1-4. Discrete Implementation of the Reduced Order Luenberger Observer

$$\begin{cases} \hat{z}_\alpha\left(n+1\right) = \left(1-h\right)z_\alpha\left(n\right) + \left(\frac{hL_s}{T_c} - R_s\right)\left[hi_\alpha\left(n\right) - \omega T_c i_\beta\left(n\right)\right] + \left[hv_\alpha\left(n\right) - \omega T_c v_\beta\left(n\right)\right] \\ \hat{z}_\beta\left(n+1\right) = \left(1-h\right)z_\beta\left(n\right) + \left(\frac{hL_s}{T_c} - R_s\right)\left[hi_\beta\left(n\right) + \omega T_c i_\alpha\left(n\right)\right] + \left[hv_\beta\left(n\right) + \omega T_c v_\alpha\left(n\right)\right] \end{cases}$$

The estimated BEMF is calculated using the following equation:

Equation 1-5. Estimated BEMF

$$\begin{cases} \hat{e}_\alpha(n) = z_\alpha(n) - \frac{hL_s}{T_c} i_\alpha(n) + \omega L_s i_\beta(n) \\ \hat{e}_\beta(n) = z_\beta(n) - \frac{hL_s}{T_c} i_\beta(n) - \omega L_s i_\alpha(n) \end{cases}$$

Where

$\hat{z}_{\alpha,} \hat{z}_\beta$ - Imaginary Internal State variables represent no physical parameter in α-β reference frame.

$\hat{e}_{\alpha,} \hat{e}_\beta$ - BEMF State variables in α-β reference frame.

T_c – Computational Step time of the observer. Typically it is the control loop period.

R_s – Per Phase Stator resistance of the motor.

L_s - Per Phase Synchronous inductance of the motor.

ω - Electrical speed of the motor in rad/sec.

The constants to be considered are as follows:

Equation 1-6. Constant 1

$0 < h < 1$(arbitrary, its value determines the system dynamics)

Equation 1-7. Constant 2

$$k = h\frac{L_s}{T_c}$$

Equation 1-8. Constant 3

$$c_0 \triangleq 1 - h$$

Equation 1-9. Constant 4

$$c_1 \triangleq k - R_s$$

Equation 1-10. Constant 5

$$c_2 \triangleq \omega T_c$$

Equation 1-11. Constant 6

$$c_3 \triangleq \omega L_s$$

The variables c_2 and c_3, which are speed dependent, should be recomputed at every iteration.

1.5.3 Phase Angle Correction

The phase angle (rotor position) is computed from the phase current measurements taken in previous cycles. The phase currents measured at the beginning of the control loop cycle are produced by the corresponding voltages applied in the previous cycle. These currents and these voltages are used to estimate the phase angle, which is used to determine the phase voltages for the next cycle.

The total delay is comprised between one and two sampling periods therefore, the error in the phase estimation is shown in the following equation.

Equation 1-12. Phase Estimation Error

$$\Delta\Phi = k\omega T_c, 1 \le k \le 2$$

When the speed is low, the error is negligible, but when the speed is high it is necessary to keep it into account (i.e., 18000 rpm in a 4 poles motor means 3770 rad/s, with PWM Frequency of 10 kHz, with k = 1.5, 0.565 rad that is 32.4°).

Figure 1-9. When the PWM Frequency and the Control Frequency are the Same

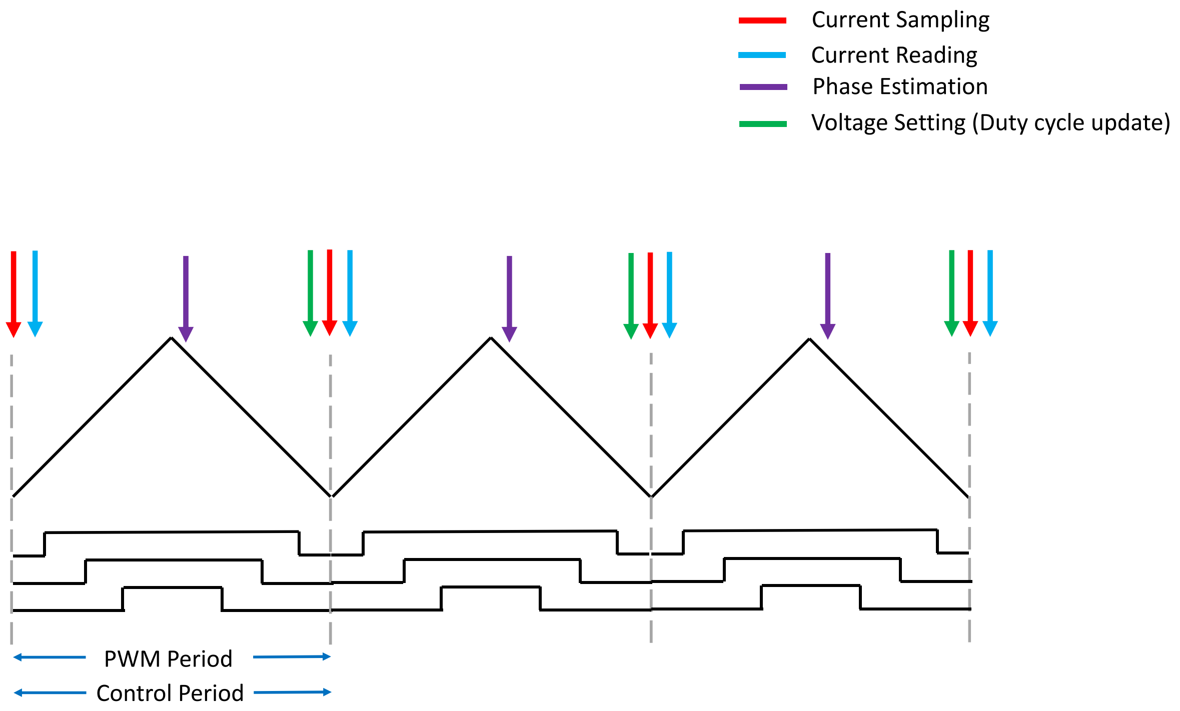
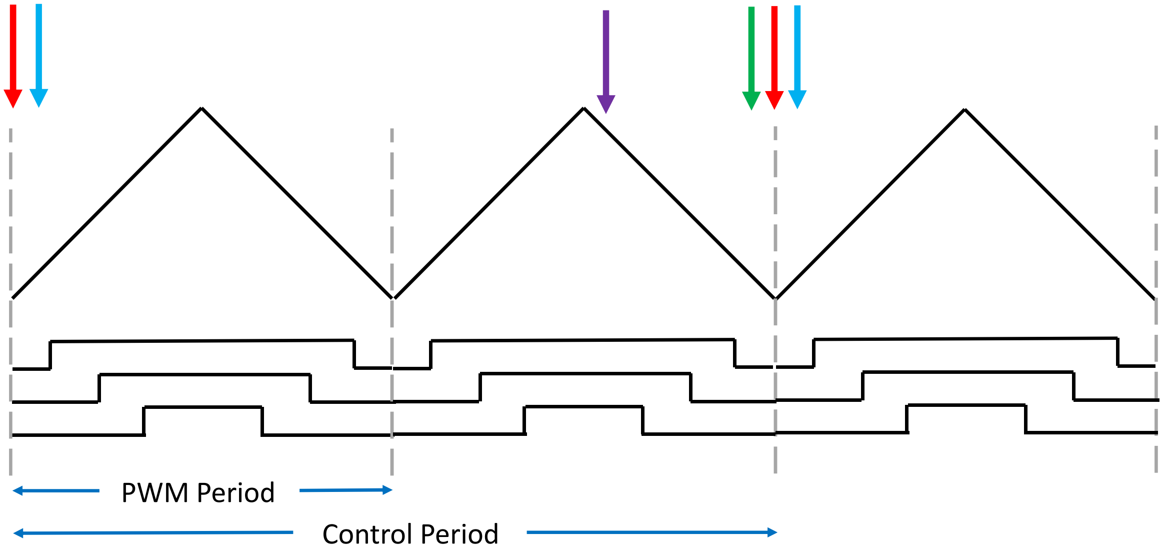


Figure 1-10. When the PWM Frequency is an Integer Multiple of the Control Frequency



1.6 Proportional Integral Controller Background

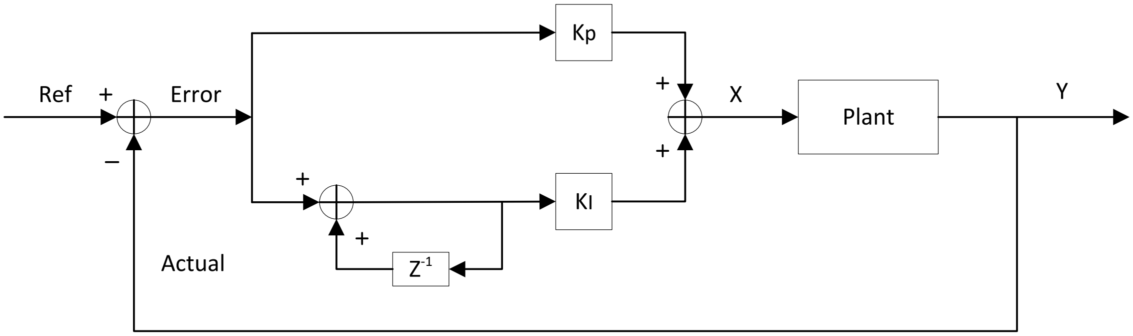
The Proportional Integral (PI) controller shown in the following figure uses the measurement of the controlled quantity (Y) to calculate an “error,” which corrects the quantity used for the control (X).

The quantity which operates on the plant (X) is computed as the sum of a term proportional to the error and a term proportional to the integral of the error.

The proportional term alone is inadequate in all the cases in which a non-zero control quantity is needed in steady state.

This “offset”, which allows the zeroing of the error in steady state, is given by the integral action.

Figure 1-11. Proportional Integral Controller



When the output (X) of the PI controller reaches its saturation value, and if the error is not zero, the integral memory continues to grow up.

In these conditions, to avoid numerical problems or delayed action when the error changes its sign, a correction of the integral action is needed. This “anti-windup” can be a simple clamp of the integral memory, or it can consist in a more or less sophisticated integral action modification strategy.

A good compromise between the complexity and the achievable results can be the following strategy:

- When the output is saturated, and the error has the same sign of the output, then the integration is stopped, even if the integral memory is still under its saturation level
- Another good strategy is to limit the integral memory under a clamp value obtained by subtracting the proportional action to the maximum allowable output value

1.6.1 Control Loop Dependencies

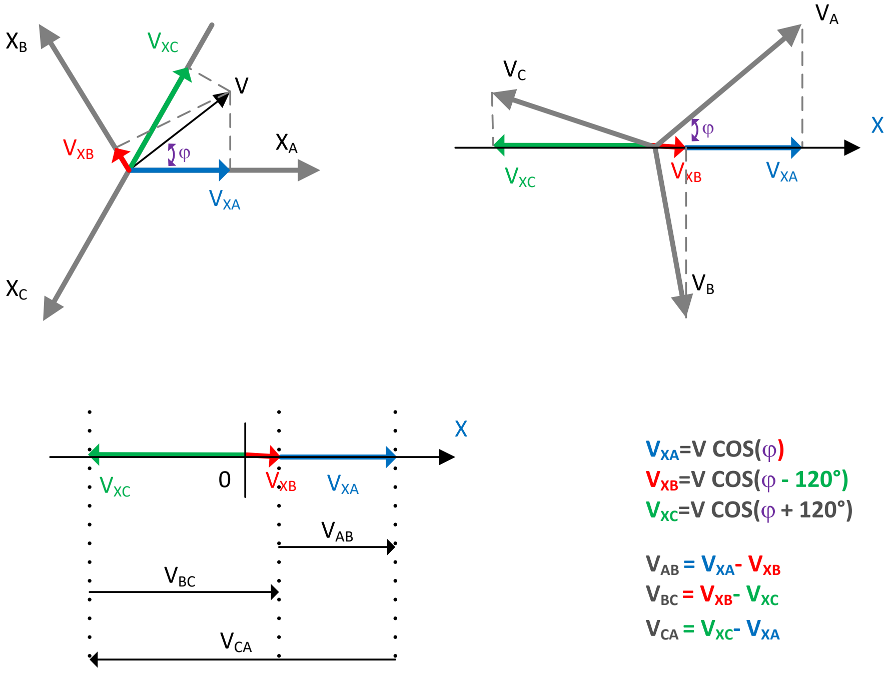
There are three interdependent PI control loops in this application. The outer loop controls the motor velocity. The two inner loops control the transformed motor currents, i_d and i_q. As mentioned previously, the i_d loop is responsible for controlling flux, and the i_q value is responsible for controlling the motor torque.

1.7 Space Vector Modulation

The final step in the vector control process is to generate pulse-width modulation signals for the three-phase motor voltage signals. In this application note the clamped (Flat top) modulation is explained. This type of modulation has an advantage of reducing switching losses compared to the conventional SVPWM.

Three phase voltages can be obtained from the projection of a single vector on three axes displaced each other 120 degrees as shown in the following figure. The same phase voltages can be obtained from the projections of three vectors displaced each other 120 degrees on a single axis as shown in the following figure.

Figure 1-12. Three Phase Voltage Vectors



$$V_{XA}=V\ COS(\varphi)$$
$$V_{XB}=V\ COS(\varphi - 120°)$$
$$V_{XC}=V\ COS(\varphi + 120°)$$

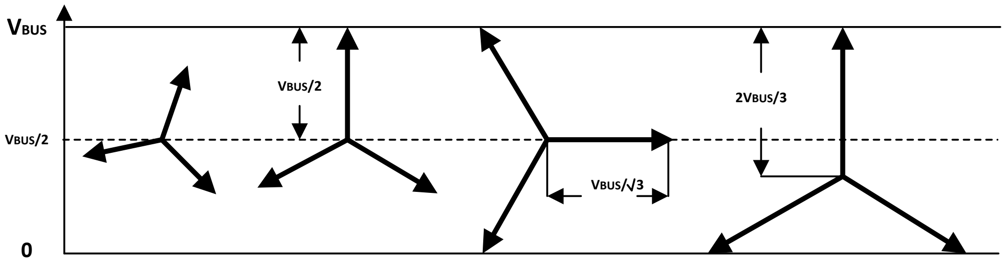
$$V_{AB} = V_{XA} - V_{XB}$$
$$V_{BC} = V_{XB} - V_{XC}$$
$$V_{CA} = V_{XC} - V_{XA}$$

The voltage differences do not depend on the position of the vectors in the X axis. This means that the phase-to-phase voltages are not influenced by the common offset of the single-phase voltages, called a *homopolar component* (this is the mean value of the three single-phase voltages).

Consider these points:

- Since no neutral connection is present in normal motors, the homopolar component of the applied phase voltages (*phase-neutral* voltage) has no influence on the motor.
- We are interested in the *phase-to-phase* voltages rather than the *phase-to-neutral* voltages.
- The maximum amplitude for the sinusoidal voltage of [(v_{BUS}/2)(1+sin(ωt))] is v_{BUS}/2 with the sinusoidal midpoint (offset) at v_{BUS}/2, referred as phase voltage (Refer to *Maximum Voltage Amplitude* below). For phase-to-phase voltages this limit becomes v_{BUS}*√(3)/2.
- This limit can be exceeded by moving the midpoint from the v_{BUS}/2 position, accordingly with the phase voltage angular position. This will affect the phase voltages (which will not be sinusoidal anymore), but not the phase-to-phase voltages. Refer to *Obtained Phase and Line Voltages* below.
- This is true till the limit (for phase voltage) of v_{BUS}/√(3) (which becomes v_{BUS} for the delta voltages). Above that, the phase voltages displaced each other by 120 degrees can be created only for some angular positions, for which the limit becomes 2v_{BUS}/3 (remember the fundamental vectors of space-vector modulation). Refer to the following figure.

Figure 1-13. Maximum Voltage Amplitude



This *clamped modulation* is realized by subtracting from each phase voltage, the lowest of the three ones on every step. The result of this operation is obviously always greater than or equal to zero. This offset addition will not influence the delta voltages, as mentioned earlier.

An intuitive representation is the *rotating star*, as shown in the following figure. There is always a request of zero voltage in one of the three inverter arms, each arm has no commutations during one third of the electrical period, so the commutation losses are reduced (by one third).

Figure 1-14. Rotating Star Representation

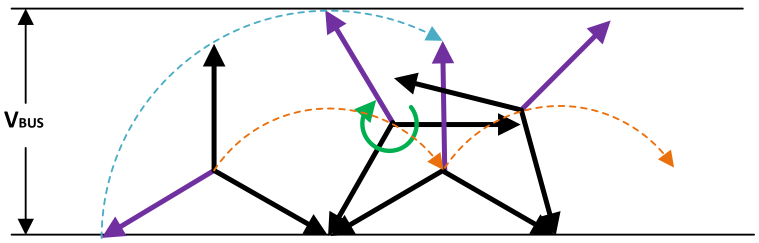
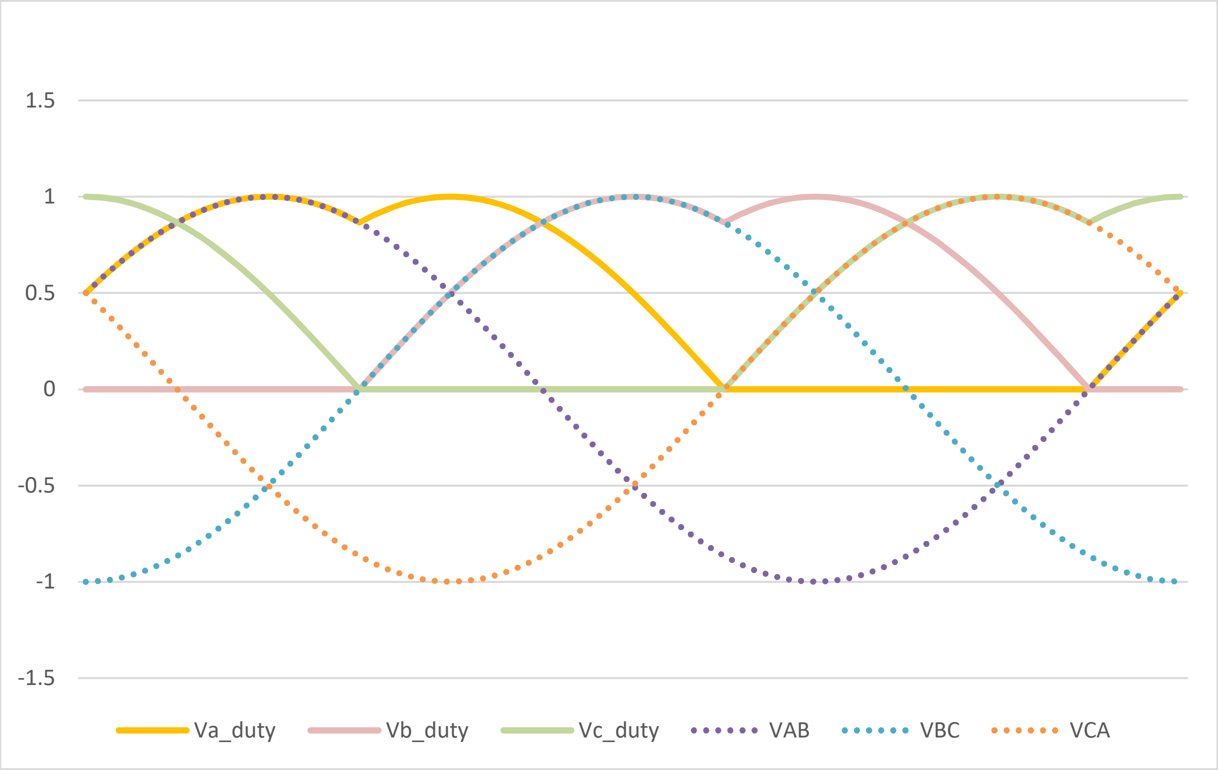


Figure 1-15. Obtained Phase and Line Voltages



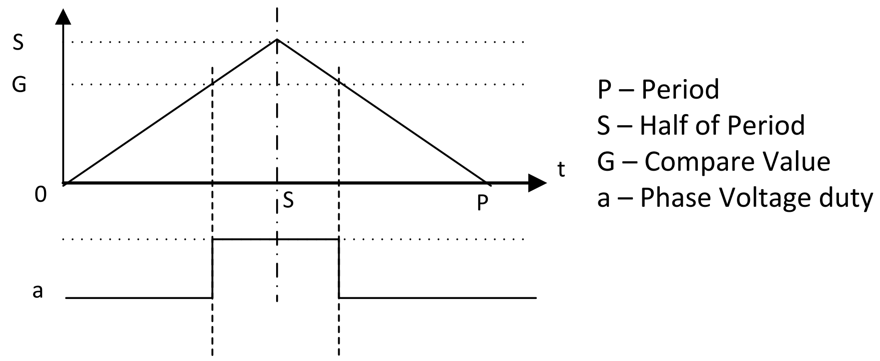
Note:
- Va_duty, Vb_duty and Vc_duty are the phase voltages.
- VAB, VBC and VCA are the phase-to-phase (Line) Voltages.

Implementation steps are as follows:

- The inputs of the modulation function are the three instantaneous voltages requests from the inverse Clarke transformation. These are v_{a_ref}, v_{b_ref} and v_{c_ref}.
- For the Center Aligned PWM (as shown in the following figure), find the three-phase voltage values with the simple modulation formula (CompareValue = (Period/2)×[(v_{BUS}-v_{REF})/v_{BUS}]). These three values cannot be directly applied to the PWM compare registers, since some of them can be negative, while the values to be written in the PWM peripheral compare registers must be positive.
  - Compare_v_a = (Period/2).[(v_{BUS}-v_{a_ref})/v_{BUS}])
  - Compare_v_b = (Period/2).[(v_{BUS}-v_{b_ref})/v_{BUS}])
  - Compare_v_c = (Period/2).[(v_{BUS}-v_{c_ref})/v_{BUS}])
- The lowest of the three values is found, then it is subtracted to all the three. One of the three results will be zero, the other ones positive (or zero). This also determines which phases are the candidates for the next current readings (the two ones with larger low switch ON period that is the two ones with higher compare register value).
  - v_{min} = min (Compare_v_a, Compare_v_b, Compare_v_c)
  - v_{a_duty} = Compare_v_a - v_{min}
  - v_{b_duty} = Compare_v_b - v_{min}
  - v_{c_duty} = Compare_v_c - v_{min}

- The calculated duty cycle values should be within the limits of 0 and half the PWM Period. The maximum limit shall be lesser if the dead time, and the minimum ON time of the lower switches to assure safe bootstrap operations for the high side driving circuitry is considered.
- If a clamp is applied in the above case to limit the duty cycle, the resulting effect to the output voltages should be considered. Therefore, the V_α and V_β components should be calculated again as these parameters are used in the rotor angle estimator.
- To keep a sinusoidal undistorted waveform, it is necessary for a preventative clamp of the voltage vector amplitude below the value $v_{BUS}/\sqrt{(3)}$. The voltage value v_{BUS} used for the clamp in implementation should keep into account the voltage losses due to the dead-times and also of the fluctuations of the bus voltage. If the bus voltage comes from rectified mains it will be affected by a ripple. To avoid torque ripple in motor control, the *minimum* bus voltage value should be used in the clamp calculation.

Figure 1-16. Center-Aligned PWM

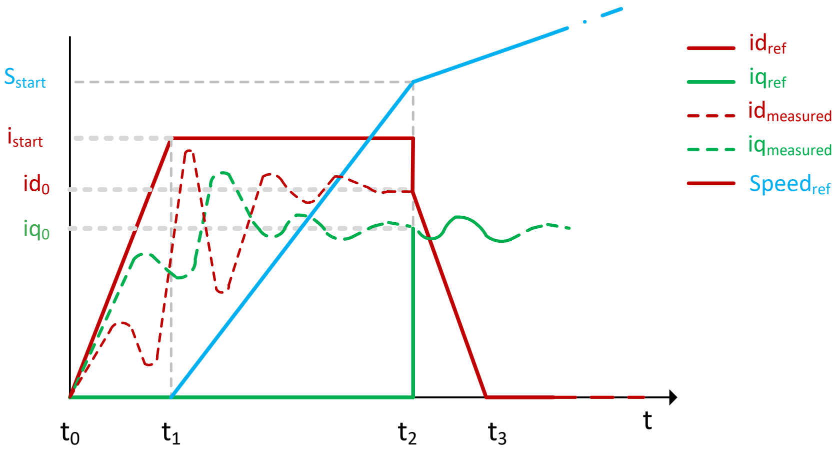


1.8 Startup Procedure

When the motor is in Stop condition, the position is usually unknown. The BEMF observer cannot work when the speed is lower than a minimum value which is determined by the motor and the application. There is a necessity of an open loop startup procedure, which allows the motor to reach the minimum speed at which the observer can work correctly. At this point the speed loop can be closed and the control assumes the form already described in the block scheme.

Since the effective position is not known, an *arbitrary* position (d, q) reference system will be used. Hence the quantities which are referred to as d-quantity or q-quantity in this process will not have any determined relationship with respect to the rotor position, until the speed loop is closed.

Figure 1-17. Startup Procedure



Notes:

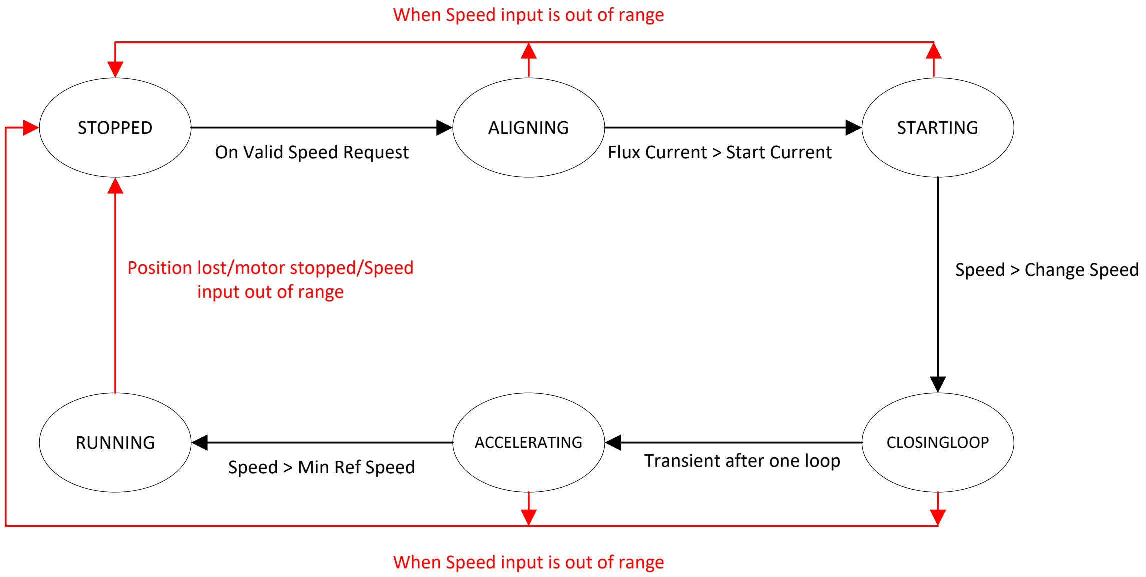
- t0 - t1 Alignment Phase
- t1 - t2 Starting Phase
- t2 - t3 Phase which combines Closing loop, Accelerating and Running.

The important parameters to be chosen for the startup procedure are as follows:

- *Start up speed*: The *speed reached during the startup* phase. This is a parameter which can be modified by the user. It should be high enough to allow a good behavior of the phase estimation process.
- *Start up time*: The *time required to reach the start up speed*; it should be long enough to allow the estimation algorithm to stabilize (to recover from the initial condition errors).
- *Start up current* (i_{start}): The *current level* imposed during the startup; the value of this current (for a particular load) should be set to the minimum value of the current which can spin the motor in an open loop without stalling.

1.9 Motor Control Management State Machine.

Figure 1-18. Motor Control Management State Machine


When Speed input is out of range
STOPPED
ALIGNING
STARTING
On Valid Speed Request
Flux Current > Start Current
Position lost/motor stopped/Speed input out of range
Speed > Change Speed
RUNNING
ACCELERATING
CLOSINGLOOP
Speed > Min Ref Speed
Transient after one loop
When Speed input is out of range

The previous diagram represents the motor control management state machine state transitions. In each state the following operations are performed:

- *Stopped*: Motor control variables reset.
- *Aligning*: The first part of the startup procedure is performed. The flux current (i_d) is increased till its nominal startup value, while the speed is kept zero.
- *Starting*: The second part of the startup procedure is performed. The speed is increased till change frequency (speed), while the flux current amplitude is kept constant.
- *Closing loop*: The reference system change is performed, with current PI memories updating.
- *Accelerating*: Verification for whether the actual speed reached the required minimum speed value specified, and the estimated position from the observer is well close to the open loop rotor angle.
- *Running*: The normal motor operations are performed. In case of a SPM, the flux current component (i_d) reference is reduced to zero. In the case of the IPM, the negative (i_d) reference is calculated from the MTPA algorithm. The torque current component (i_q) reference is obtained with a speed control PI.

The BEMF observer angle and speed estimation are performed during the Accelerating and Running States.

2. Revision History

2.1 Revision C - November 2024

This have the same content that revision B. New version to release the document in OnlineDocs.

2.2 Revision B - December 2017

Document migrated to Publication Manager.

2.3 Revision A

This is the initial released version of this document.

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-0183-5

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.