

# XC800 Family

## AP08091

Sensorless Control of Brushless DC Motor using Infineon XC864  
Microcontroller

Application Note V1.0

2009-08

**Edition 2009-08**

**Published by  
Infineon Technologies AG  
81726 Munich, Germany**

**© 2009 Infineon Technologies AG  
All Rights Reserved.**

#### **LEGAL DISCLAIMER**

THE INFORMATION GIVEN IN THIS APPLICATION NOTE IS GIVEN AS A HINT FOR THE IMPLEMENTATION OF THE INFINEON TECHNOLOGIES COMPONENT ONLY AND SHALL NOT BE REGARDED AS ANY DESCRIPTION OR WARRANTY OF A CERTAIN FUNCTIONALITY, CONDITION OR QUALITY OF THE INFINEON TECHNOLOGIES COMPONENT. THE RECIPIENT OF THIS APPLICATION NOTE MUST VERIFY ANY FUNCTION DESCRIBED HEREIN IN THE REAL APPLICATION. INFINEON TECHNOLOGIES HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND LIABILITIES OF ANY KIND (INCLUDING WITHOUT LIMITATION WARRANTIES OF NON-INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF ANY THIRD PARTY) WITH RESPECT TO ANY AND ALL INFORMATION GIVEN IN THIS APPLICATION NOTE.

#### **Information**

For further information on technology, delivery terms and conditions and prices, please contact the nearest Infineon Technologies Office ([www.infineon.com](http://www.infineon.com)).

#### **Warnings**

Due to technical requirements, components may contain dangerous substances. For information on the types in question, please contact the nearest Infineon Technologies Office.

Infineon Technologies components may be used in life-support devices or systems only with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

CONFIDENTIAL

**Document Change History**

Date	Version	Changed By	Change Description
2009-08-17	1.0		

**We Listen to Your Comments**

Is there any information in this document that you feel is wrong, unclear or missing?  
Your feedback will help us to continuously improve the quality of this document.  
Please send your proposal (including a reference to this document) to:

[mcdocu.comments@infineon.com](mailto:mcdocu.comments@infineon.com)



	Table of Contents	Page
<b>1</b>	<b>Introduction .....</b>	<b>7</b>
1.1	Overview .....	7
1.2	Hardware and Software Components .....	7
<b>2</b>	<b>Principle of Sensorless Control.....</b>	<b>8</b>
2.1	Motor Theory .....	8
2.2	Principle of Operation.....	8
2.3	Hall Sensor Mode.....	8
2.4	Three Phase Inverter .....	9
2.5	Sensorless Mode of Operation.....	10
2.6	Back EMF Measurement.....	11
2.7	Speed Control of BLDC Motor .....	12
<b>3</b>	<b>Software Implementation .....</b>	<b>13</b>
3.1	Control System Overview.....	13
3.2	Peripheral Initialization .....	14
3.3	CCU6 T13 Period Match ISR .....	15
3.3.1	Commutation function .....	16
3.3.1.1	Bemf Detection Logic .....	16
3.3.1.2	Rampup Function .....	22
3.3.2	Channel Selection .....	23
3.3.3	PI Controller .....	23
3.3.4	Speed Rampup Function .....	25
3.4	CCU6 CCU62 Compare Match ISR.....	26
3.5	T12 PM and CTrap ISR.....	26
3.6	Timer T2 ISR.....	27

<b>List of Figures</b>	<b>Page</b>
Figure 1 Single Pole Pair BLDC Motor with Hall Sensor.....	8
Figure 2 Three Phase Voltage Source Inverter.....	9
Figure 3 Phase Voltage and Induced EMF .....	10
Figure 4 Phase Voltage and ADC Sampling Time .....	11
Figure 5 Open-loop Speed Control .....	12
Figure 6 Closed loop Speed Control .....	12
Figure 7 Block Diagram for Sensorless Control of BLDC Motor .....	13
Figure 8 T13 Period Match ISR.....	15
Figure 9 Behaviour of Motor during Open and Close Loop.....	16
Figure 10 Phase Voltage at 100% Duty Cycle .....	17
Figure 11 BEMF Detection Timing Diagram .....	18
Figure 12 Flow Chart of Commutation Function .....	21
Figure 13 Open loop Rampup Function .....	22
Figure 14 Channel Selection Function .....	23
Figure 15 Block Diagram for PI Controller.....	23
Figure 16 Speed Rampup Function .....	25
Figure 17 CCU62 Compare Match ISR.....	26

<b>List of Tables</b>	<b>Page</b>
Table 1    Example Commutation Sequence with reference to Hall Sensor Output .....	9
Table 2    Motor Position and Commutation Sequence .....	10
Table 3    Motor Operation Modes .....	16
Table 4    UART Request Message .....	27
Table 5    Variable Location .....	27

## **1 Introduction**

### **1.1 Overview**

The BLDC motor is used for consumer, industrial and automotive applications, due to its compact size, controllability and high efficiency. The BLDC is usually operated with rotor position sensors, since the electrical excitation must be synchronous with the rotor position. For the reasons of cost, reliability and mechanical packaging, it is desirable to eliminate position sensor. That makes it all the more important to control the BLDC motor without the position sensor (Sensorless Operation).

This application note describes the implementation of a Sensorless control algorithm for BLDC motors. In the following chapters, the principle of Sensorless control and software implementation of the same for the XC864 microcontroller is discussed in detail. Also the advantages of the microcontroller peripherals: CAPCOM6E (Capture and Compare Unit for modulation and PWM generation) and the fast 10-bit ADC (Analog-to-Digital Converter), which are specifically designed for the motor control applications are discussed.

### **1.2 Hardware and Software Components**

For a workable system, the following hardware and software components are required:

- PC with Microsoft Windows 2000 or Windows XP or Windows Vista operating system
- Infineon XC864 Drive card
- Infineon Low Voltage Inverter Board
- Infineon Drive Monitor Stick
- BLDC Motor – MAXON EC32 15W
- 24 V Power supply for Drive Board
- KEIL (μV3) Tool chain for Infineon XC864

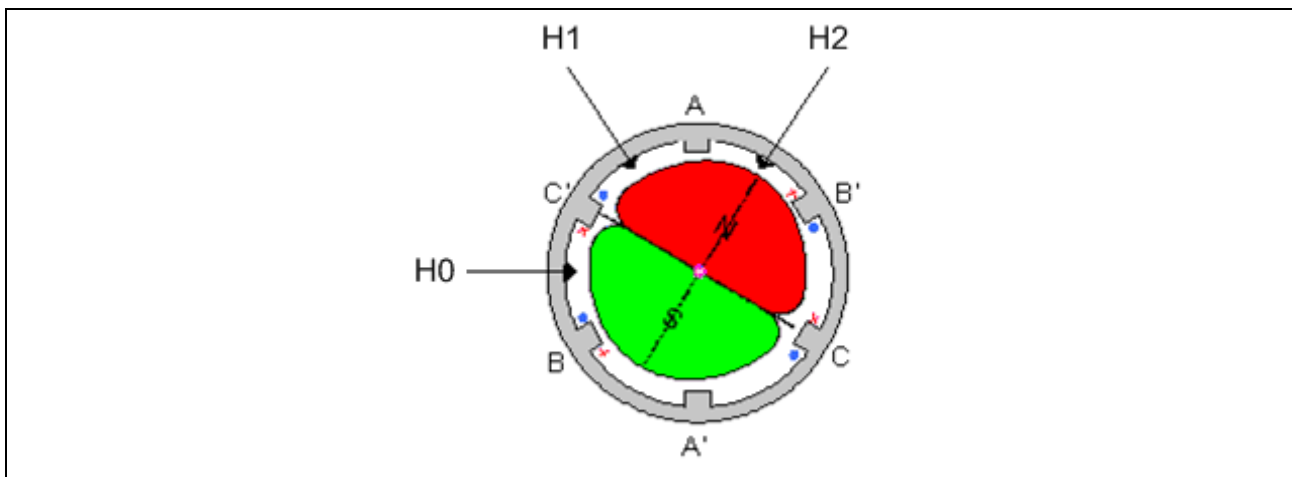
## 2 Principle of Sensorless Control

### 2.1 Motor Theory

The Brushless DC motors are a variant of Permanent magnet DC motors. PM DC Motors are synchronous motors in which the rotor field is driven with a constant current. By driving the rotor winding with a constant current, a fixed magnetic flux is established within the motor. The same also can be achieved by replacing the rotor winding with permanent magnets. By changing the stator magnets with three phase windings, the commutation can be achieved electronically compared to the mechanical commutation in common DC motors. Such motors are called Brushless DC motors. As this type of construction eliminates the need of brushes, the maintenance is reduced and the reliability is increased.

### 2.2 Principle of Operation

To rotate the motor, the stator windings should be energized in a sequence. In case of Brush DC motors, the brushes automatically will come into contact with the commutator of a different coil causing the motor to continue its rotation. But in the case of BLDC motors, the commutation has to be done through electronic switches which need the position of the rotor.



**Figure 1 Single Pole Pair BLDC Motor with Hall Sensor**

### 2.3 Hall Sensor Mode

In sensor mode operation, rotor position is sensed using Hall Effect sensors embedded into the stator. Most BLDC motors have three Hall sensors embedded into the stator on the non driving end of the motor. Each sensor gives a high or low signal, indicating the North or South Pole of the rotor is near. Based on the combination of these Hall sensor signals, the exact sequence of commutation can be determined.

Each commutation sequence has one of the windings energized to positive power (current enters into the winding), the second winding is negative (current exits the winding) and the third is in a non-energized condition. Torque is produced because of the interaction between the magnetic field generated by the stator coils and the permanent magnets. Ideally, the peak torque occurs when these two fields are at 90 degrees to each other and falls off as the fields move together. In order to keep the motor running, the magnetic field produced by the windings should shift position as the rotor moves to catch up with the rotor field. What is known as "Six-Step commutation or Block Commutation" defines the sequence of energizing windings.

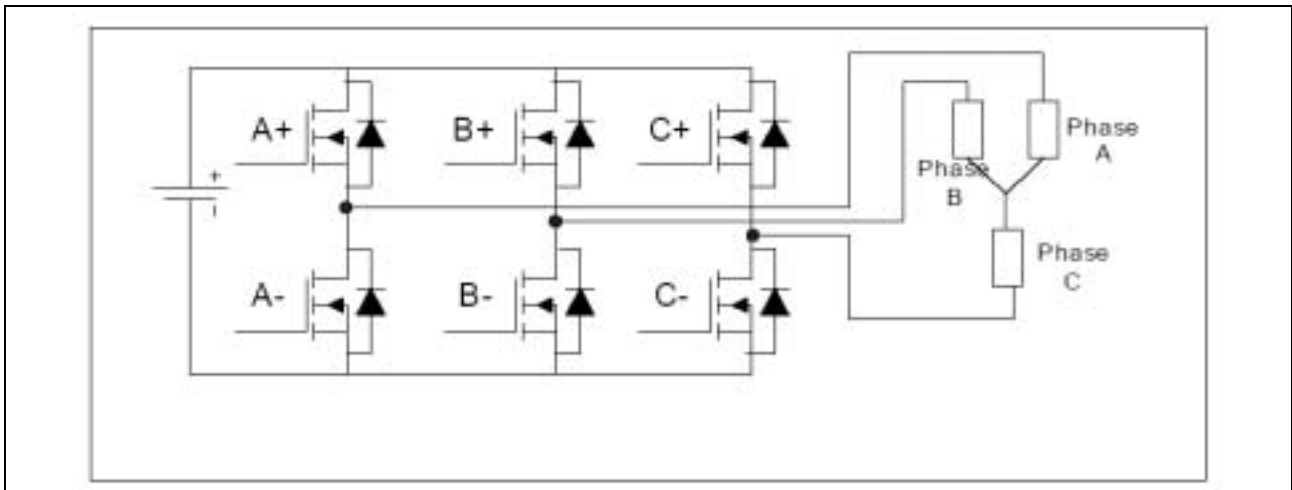


For every 60 electrical degrees of rotation, one of the Hall sensors changes its state as shown in Table 1. The motor takes six steps to complete one electrical cycle for a three phase machine. In general, the relationship between mechanical and electrical degrees is as stated below.

$$\text{Electrical Revolution} = \frac{\text{Mechanical Revolution}}{\text{Pole Pairs}} \quad \dots (1.1)$$

## 2.4 Three Phase Inverter

An inverter is an electronic circuit for converting direct current to alternating current. The structure of a typical three phase voltage source power inverter is shown in Figure 2. The six MOSFETs are controlled by the input PWM signals (A+, A-, B+, B-, C+ and C-), that shape the input voltages supplied to the motor terminals



**Figure 2 Three Phase Voltage Source Inverter**

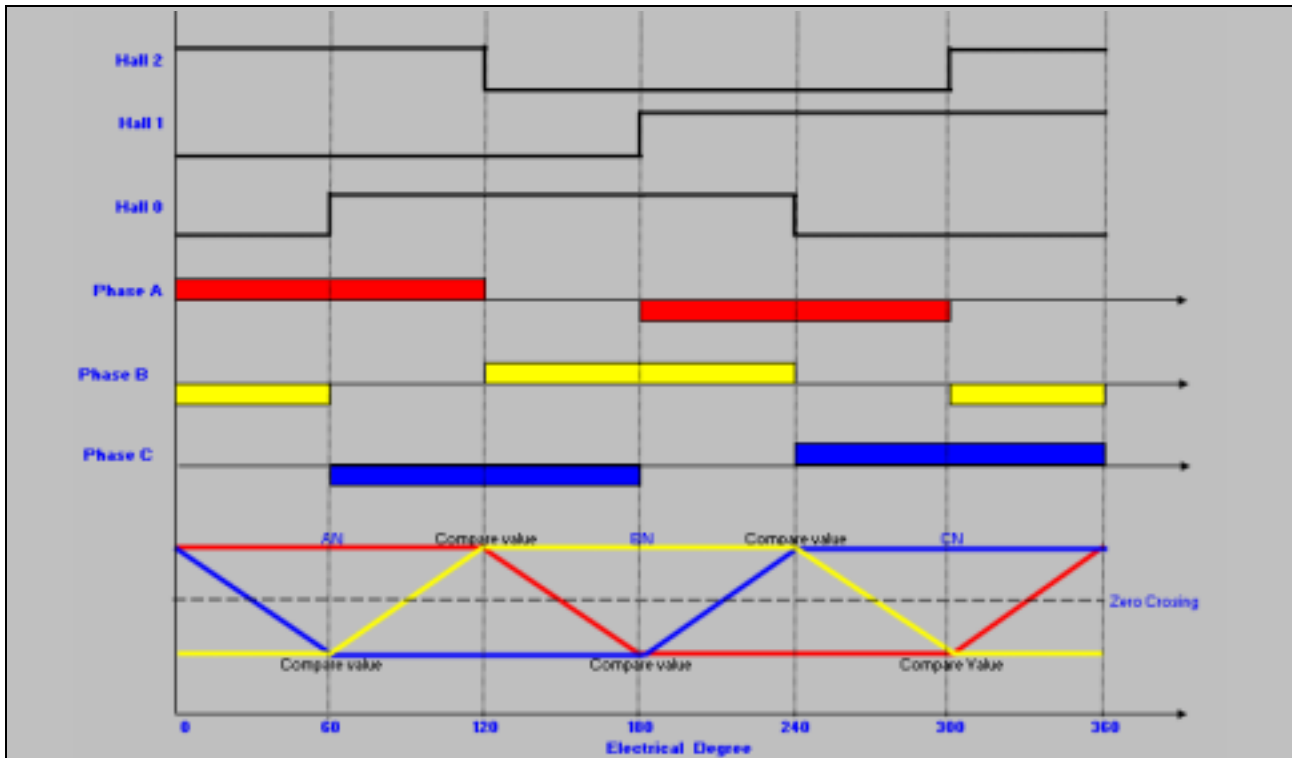
Note that whenever the MOSFET A+ is switched on, MOSFET A- must be switched off and visa versa, to prevent damaging shoot-through current.

**Table 1 Example Commutation Sequence with reference to Hall Sensor Output**

Hall Pattern [H2 H1 H0]	100	101	001	011	010	110
Phase C	+	0	-	-	0	+
Phase B	-	-	0	+	+	0
Phase A	0	+	+	0	-	-
A+	Off	On	On	Off	Off	Off
B+	Off	Off	Off	On	On	Off
C+	On	Off	Off	Off	Off	On
A-	Off	Off	Off	Off	On	On
B-	On	On	Off	Off	Off	Off
C-	Off	Off	On	On	Off	Off

## 2.5 Sensorless Mode of Operation

One of the most commonly used methods for acquiring position information is to monitor the induced EMF of the machine phases when they are not being energized. In BLDC motor drive systems, one phase is inactive 33.33% of the time and at any given time two phases conduct. During the inactive time of a Phase winding, an induced EMF appears across that winding, which can be sensed. The induced EMF of the phase will indicate when that phase has to be energized.



**Figure 3 Phase Voltage and Induced EMF**

As shown in Figure 3, the back-emf is trapezoidal in shape and it can be seen that at any given time only two of the 3 phases conduct. The inverter switching pattern can be derived easily from the back-emf. This switching pattern is organized into 6 commutation states

**Table 2 Motor Position and Commutation Sequence**

Position	Energized Phase	Non Energized Phase
0°	A+, B-	C
60°	A+, C-	B
120°	B+, C-	A
180°	B+, A-	C
240°	C+, A-	B
300°	C+, B-	A

## 2.6 Back EMF Measurement

In the Block Commutation while two phases are conducting the neutral voltage is approximately one half of the DC link voltage.

The relation between phase voltage and back EMF is as stated below

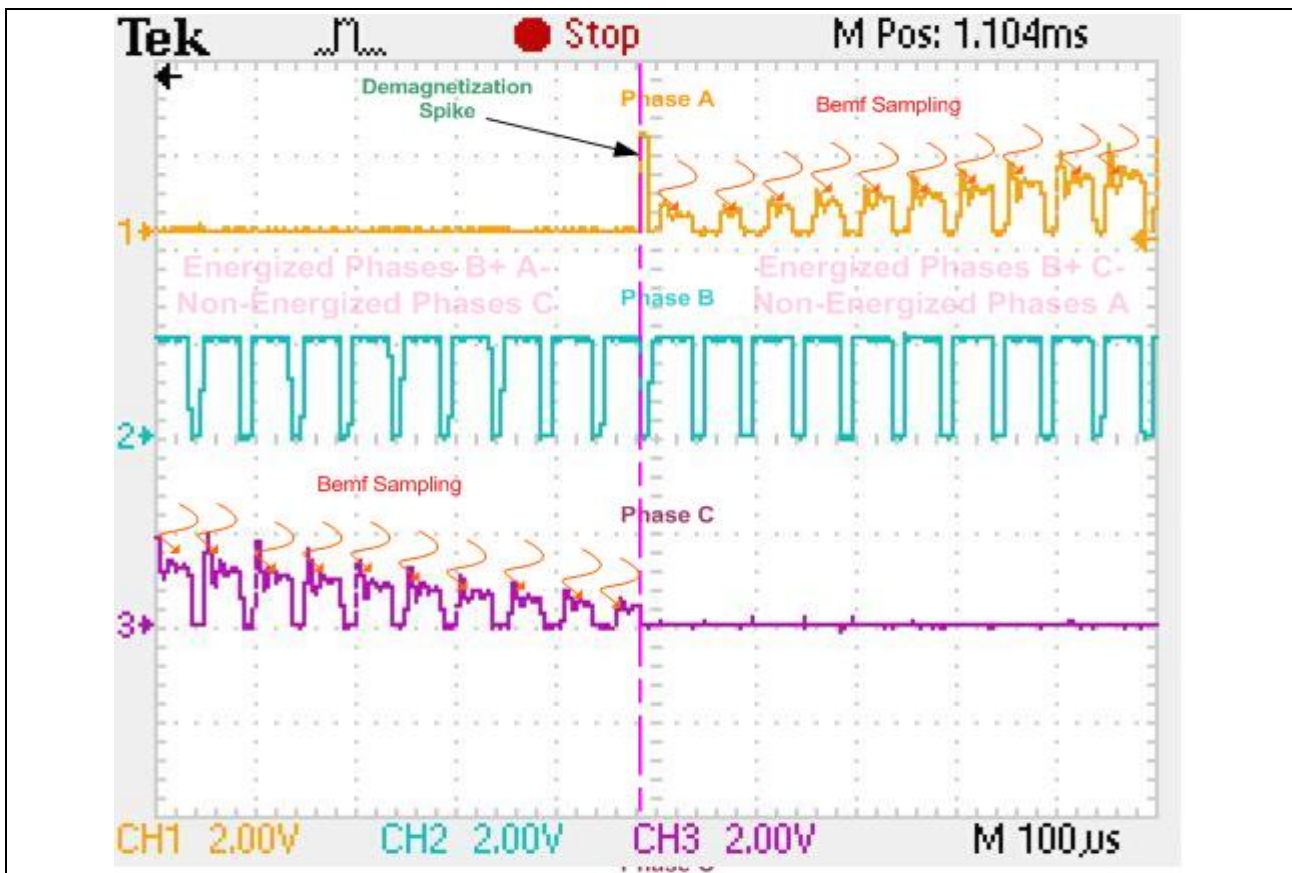
$$V_p = R \cdot I + L \cdot \frac{di}{dt} + E_{emf} \quad \dots (1.2)$$

Where	$V_p$	- Phase Voltage
	$R$	- Winding Resistance
	$L$	- Winding Inductance
	$I$	- Phase Current
	$di/dt$	- Rate of change of current over time
	$E_{emf}$	- Back emf

There is no current in the non-energized phase, so the equation (1.2) becomes

$$V_p = E_{emf} \quad \dots (1.3)$$

This means that by measuring the terminal voltage in the non-energized phase, the back emf can be easily determined. However the above conclusion is valid only when the two conducting phases are active. If one or both of the phases are being chopped, then the neutral voltage will vary and the relation between terminal voltage and back emf will not be valid. For this reason, the terminal voltage measurement should be synchronized with the PWM signal used for chopping. This is shown in Figure 4.



**Figure 4 Phase Voltage and ADC Sampling Time**

The disadvantage of using the ADC is that it is difficult to achieve a high speed range. This is because the ADC sampling is performed only once per PWM cycle. Therefore when the motor speed increases, the number of PWM cycle per commutation is decreased. However to obtain an accurate zero crossing measurement probably a minimum of 12 PWM periods per commutation are needed. This limits Sensorless operation at high speed especially for motors with a large number of poles. This problem can be worse for applications that want to minimize switching losses by using a low PWM frequency.

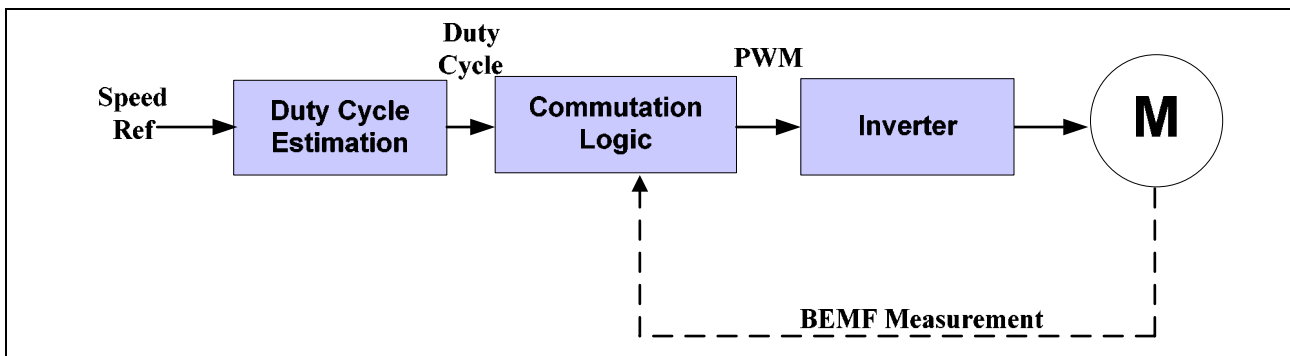
## 2.7 Speed Control of BLDC Motor

The speed of the motor is directly proportional to the applied voltage. The average voltage applied to the motor can be varied using Pulse Width Modulation (PWM) by switching the MOSFET on or off. At 100% PWM duty cycle the motor will run at rated speed provided the rated dc voltage is supplied. To operate the motor at a desired speed below the rated speed, either the high side or low side transistor should be pulse width modulated.

Two control schemes are possible:

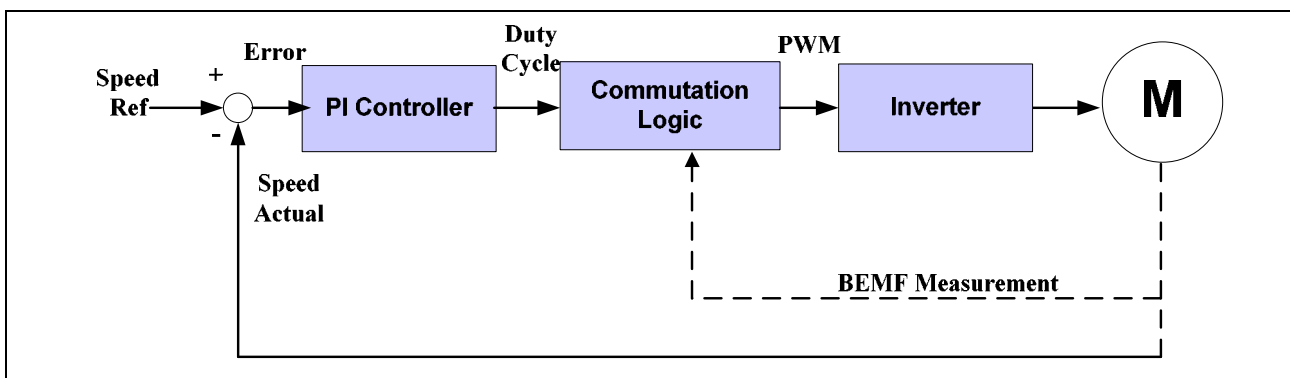
1. Open-loop speed Control (Voltage Control)
2. Closed-loop speed Control

In Open loop speed control, the duty cycle is calculated based on the set reference speed. In case of closed loop speed control the actual speed is measured and compared with the reference speed to find the error difference. This error difference will be supplied to the PI controller. The output from the PI controller gives the desired duty cycle.



**Figure 5 Open-loop Speed Control**

Figure 5 shows the open loop speed control of BLDC motor. The duty cycle for a set reference speed is estimated based on the nominal base speed of the motor.



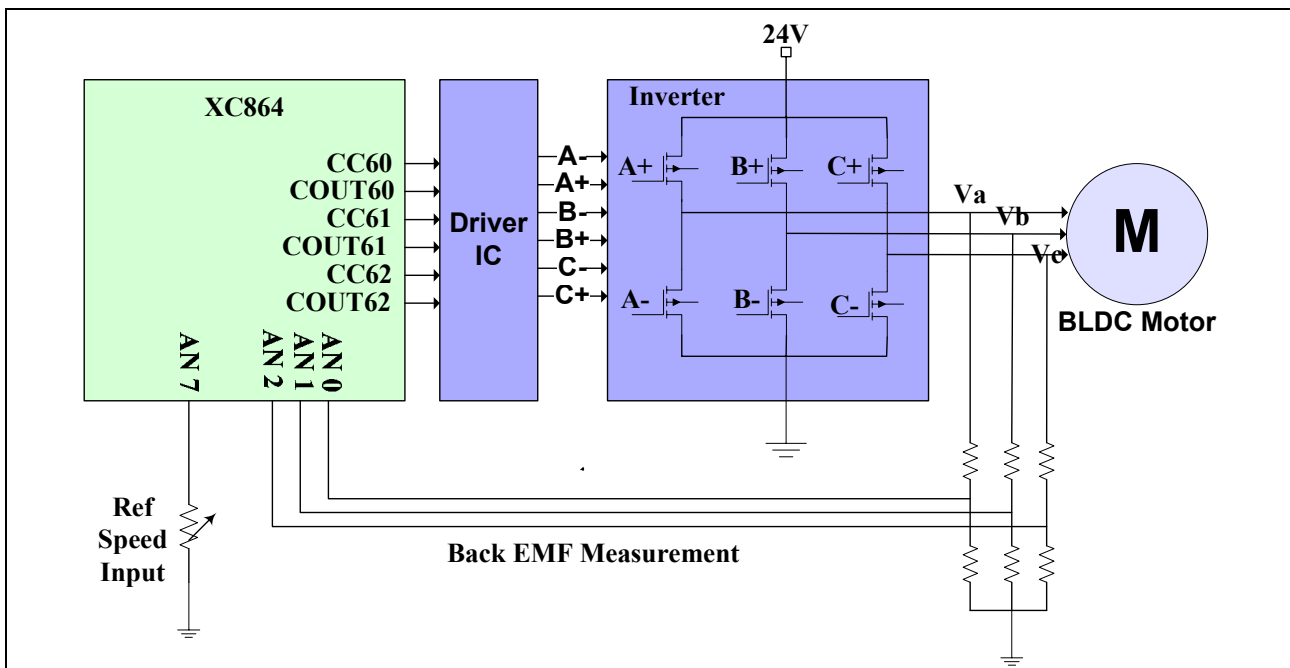
**Figure 6 Closed loop Speed Control**

## 3 Software Implementation

In this chapter, the implementation of a Sensorless Speed control of BLDC motor in the XC864 microcontrollers is discussed in detail.

### 3.1 Control System Overview

An implementation of a Sensorless Speed control of BLDC for inverter fed induction motors in closed loop is shown in Figure 7.



**Figure 7 Block Diagram for Sensorless Control of BLDC Motor**

Three on-chip peripheral modules are used to implement this application in the XC864 microcontroller and they are CCU6E (CAPCOM6E), ADC (Analog-to-Digital Converter) and Timer T2.

The CCU6E module is used to generate the PWM control signals for the inverter. For this purpose, timer T12, timer T13, CC60SR, CC61SR, CC62SR compare registers and MCMOUT registers are used. Timer T12 and Timer T13 operation are configured for edge aligned Mode. Dead-time control is enabled for the six PWM signals to avoid shoot-through current. T12 Timer is used to update the commutation pattern and calculating the motor speed. Timer T13 is used for pulse width modulation to control the motor speed. The main control algorithm is executed in Timer T13 period match ISR

The ADC module is used to measure the induced emf in the non energized phase and the speed reference. The measured induced EMF value of Phase A (in channel 0), Phase B (in channel 1) and Phase C (in channel 2) are stored in result register 0. Channel 7 is configured to read the speed reference value via POT and the result is stored in result register 2.

In the Timer T2 overflow interrupt service routine, a scaled reference speed is calculated based on the ADC input.

The software is divided into several routines:

- Main loop:
  - Initialization (CPU, I/O ports, CAPCOM6, ADC, UART and Timer T2)
- Interrupt routines:
  - CAPCOM6
    - T13 Period Match
    - T12 Compare match of Channel 2
    - T12 Period Match and CTRAP
  - T2 Over Flow
  - UART

### **3.2 Peripheral Initialization**

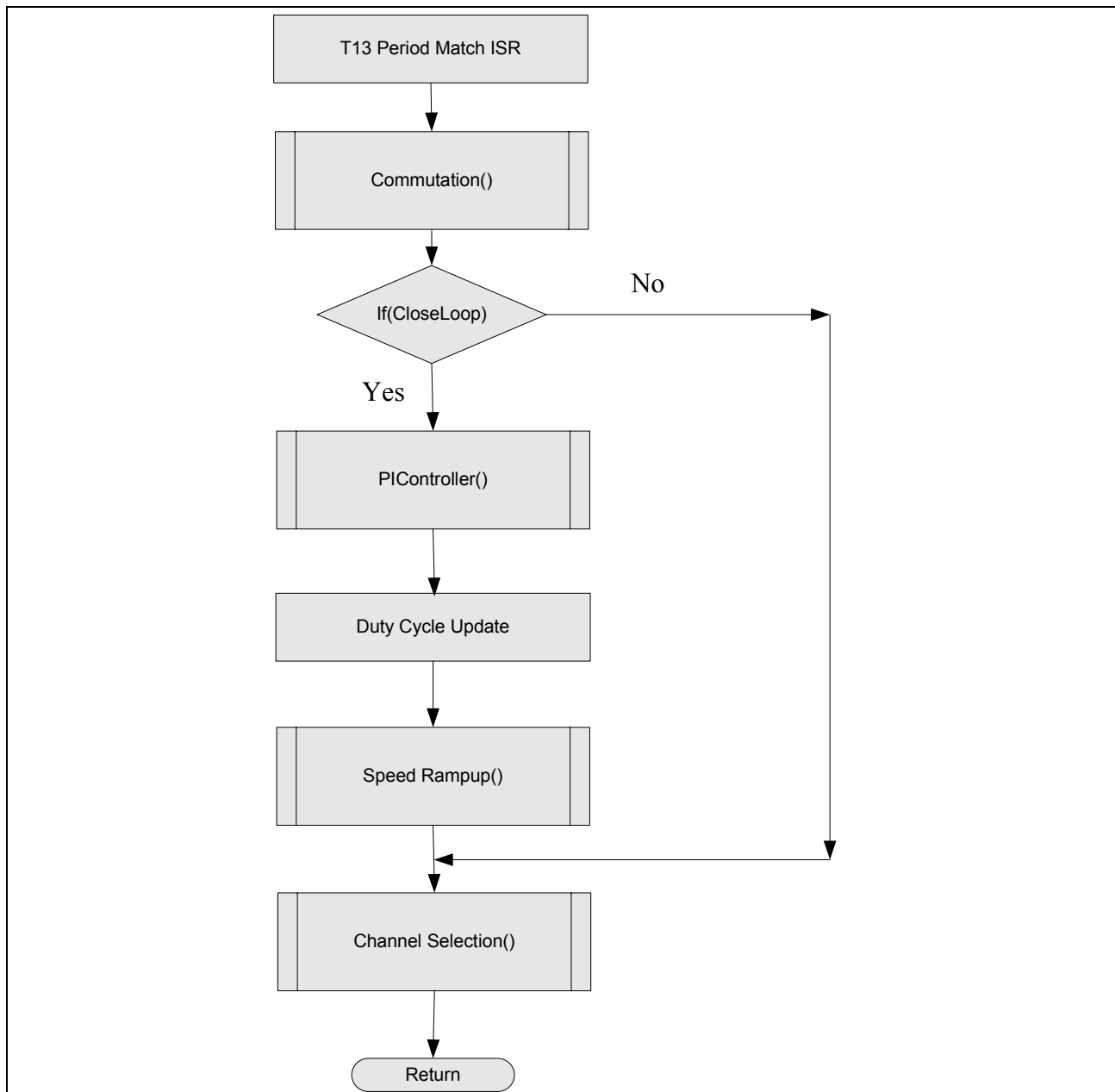
All the necessary initialization routines have been performed, before the motor is started:

- Port initialization:
  - P3.0, P3.1, P0.0, P0.1, P0.4 and P0.5 are used as alternate output for the CAPCOM6 output (CC6x, COUT6x).
  - P0.3 is used as output for Enable /Disable Drive Board
- CAPCOM6 initialization:
  - To enable Multi-channel and Hall Sensor modes
  - To set the passive output level as High
  - Timer T13 Period value set to 50  $\mu$ s (20 kHz)
  - Timer T12 configured for Edge aligned mode
  - MCMOUT register shadow transfer enabled during CCU61 compare match with optional synchronization on T13 zero match.
  - Trap function is enabled for emergency stop.
- ADC initialization:
  - P2.0-P2.2 (Channel 0-2 )as AD channels, 10bit, Sampling at T13 Compare match, conversion time 5.1  $\mu$ s, Arbitration Slot 0 (Sequential Source) for measuring the phase voltages
  - Select P2.7 (Channel 7) as AD channel, 10 bit, sampling at T13 Period match, conversion time 5.1  $\mu$ s, Arbitration Slot 1 (Parallel Source) for measuring the reference speed
- Timer T2 Initialization:
  - Timer overflow value 9.216 ms, Automatic reload
- UART Initialization:
  - Mode 1- 8 bit shift UART, Baud rate : 256 kbaud

After peripheral initialization, the motor initialization function is called. During the function call motor control specific variables are initialized and the motor start function is called to start the motor.

### 3.3 CCU6 T13 Period Match ISR

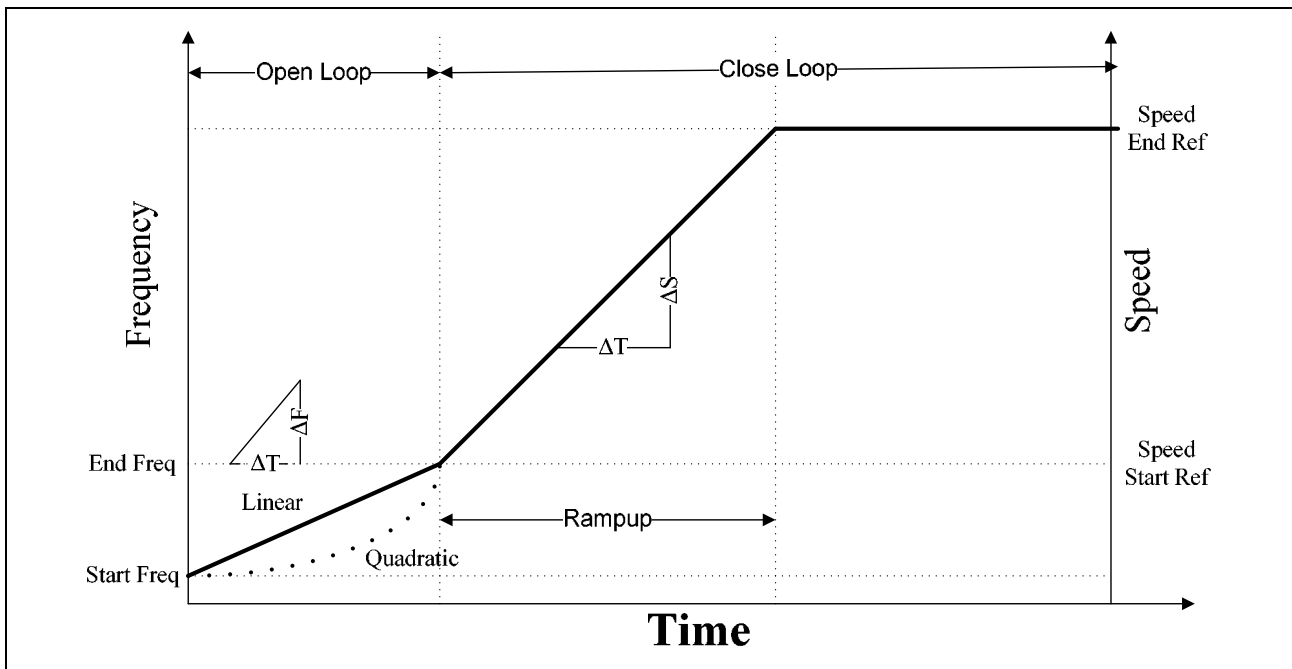
This interrupt routine is executed for every 50  $\mu$ s (PWM frequency is 20 K). During this interrupt routine, the Commutation function and Channel selection function are called and if motor is running in closed loop (Sensorless Mode) the PI controller function and Speed Rampup functions are also called.



**Figure 8 T13 Period Match ISR**

### 3.3.1 Commutation function

Once the Motor Start function is called, the motor will start to run in Open Loop mode. During this phase, the commutation speed and the phase voltage are increased continuously until the back-EMF voltage is interpretable. Then the application switches to the closed loop mode and the motor is accelerated until it reaches the reference speed level.



**Figure 9 Behaviour of Motor during Open and Close Loop**

This function can handle 4 different operation modes

**Table 3 Motor Operation Modes**

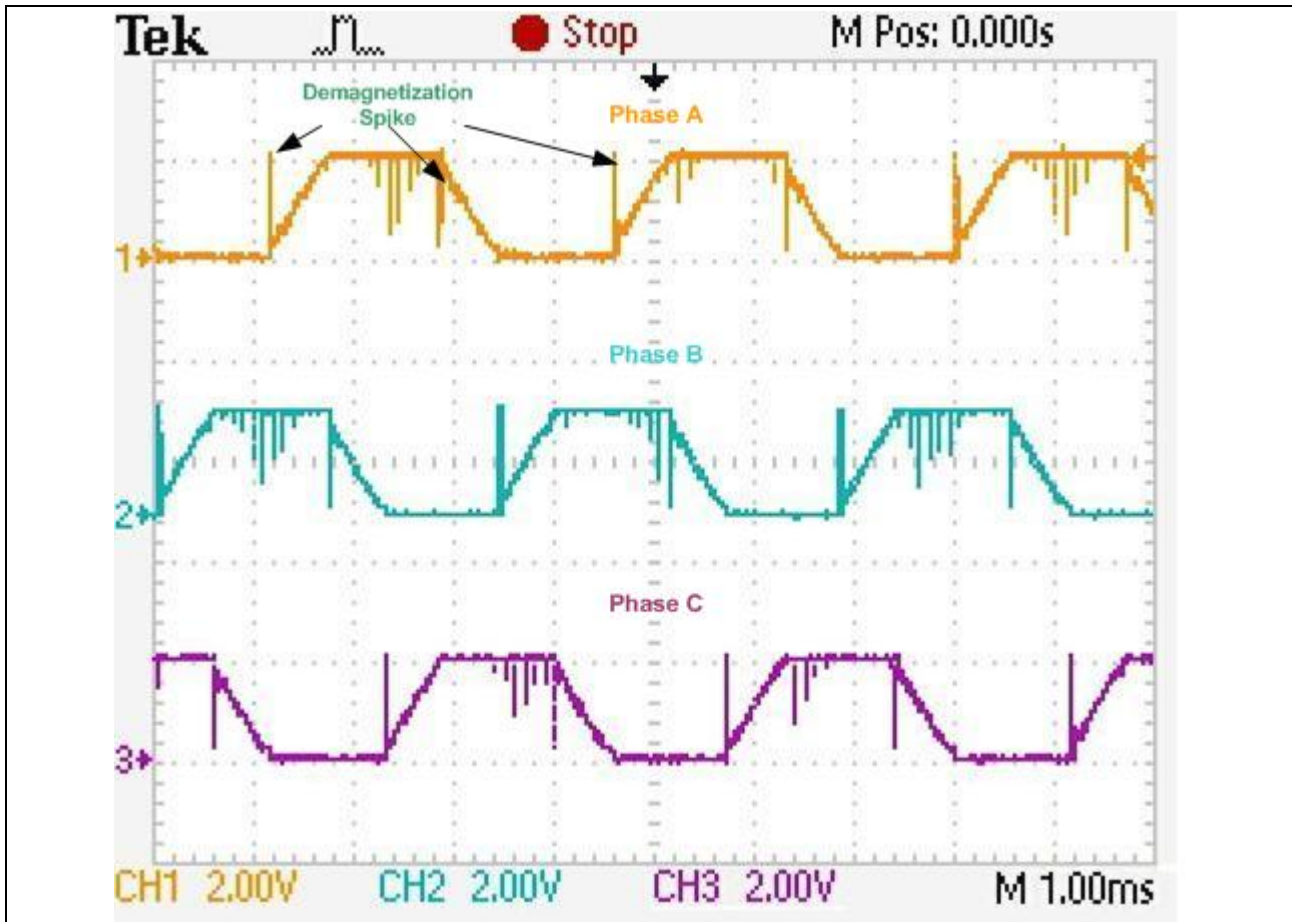
State	Action
1	Open Loop - Rampup Phase for Bemf Detection
2	Start of time Between two Zero Crossing
3	Normal Running Mode
4	Turn off Motor

#### 3.3.1.1 Bemf Detection Logic

As discussed in 2.6, the back emf measurement should be synchronized with the PWM signal used for chopping. In the implementation Timer T13 is used for chopping, so unexcited phase voltage is measured during every CCU63 compare match event.

When a new commutation pattern has been loaded into MCMOUT register, the unexcited phase voltage is measured for every CCU63 Compare ISR via ADC. Demagnetization spikes will occur whenever a new commutation pattern is applied. This spike will affect the back emf and may be interpreted as a zero crossing event. In order to ignore this spike, zero crossing detection is ignored for a predefined delay time after applying every new commutation pattern.



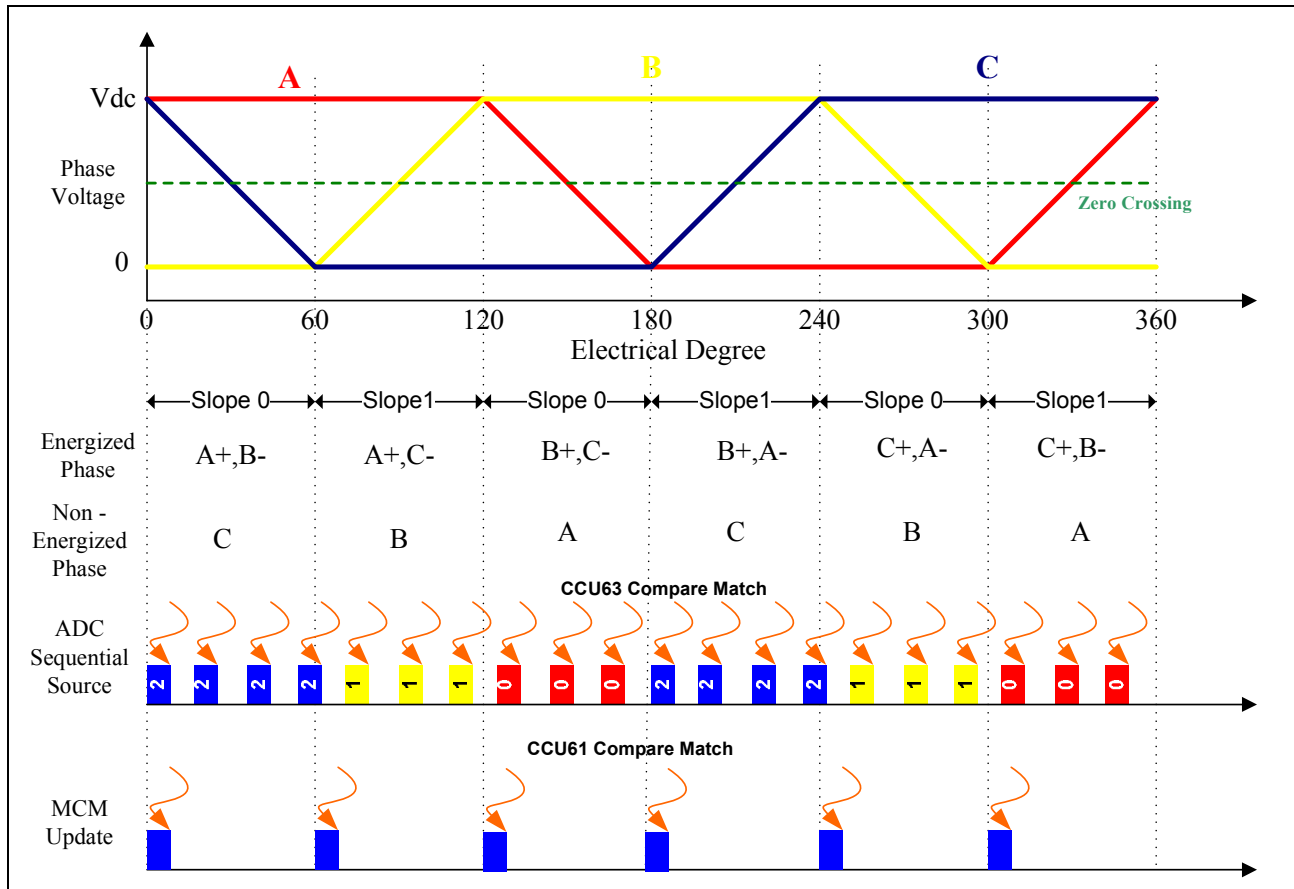


**Figure 10 Phase Voltage at 100% Duty Cycle**

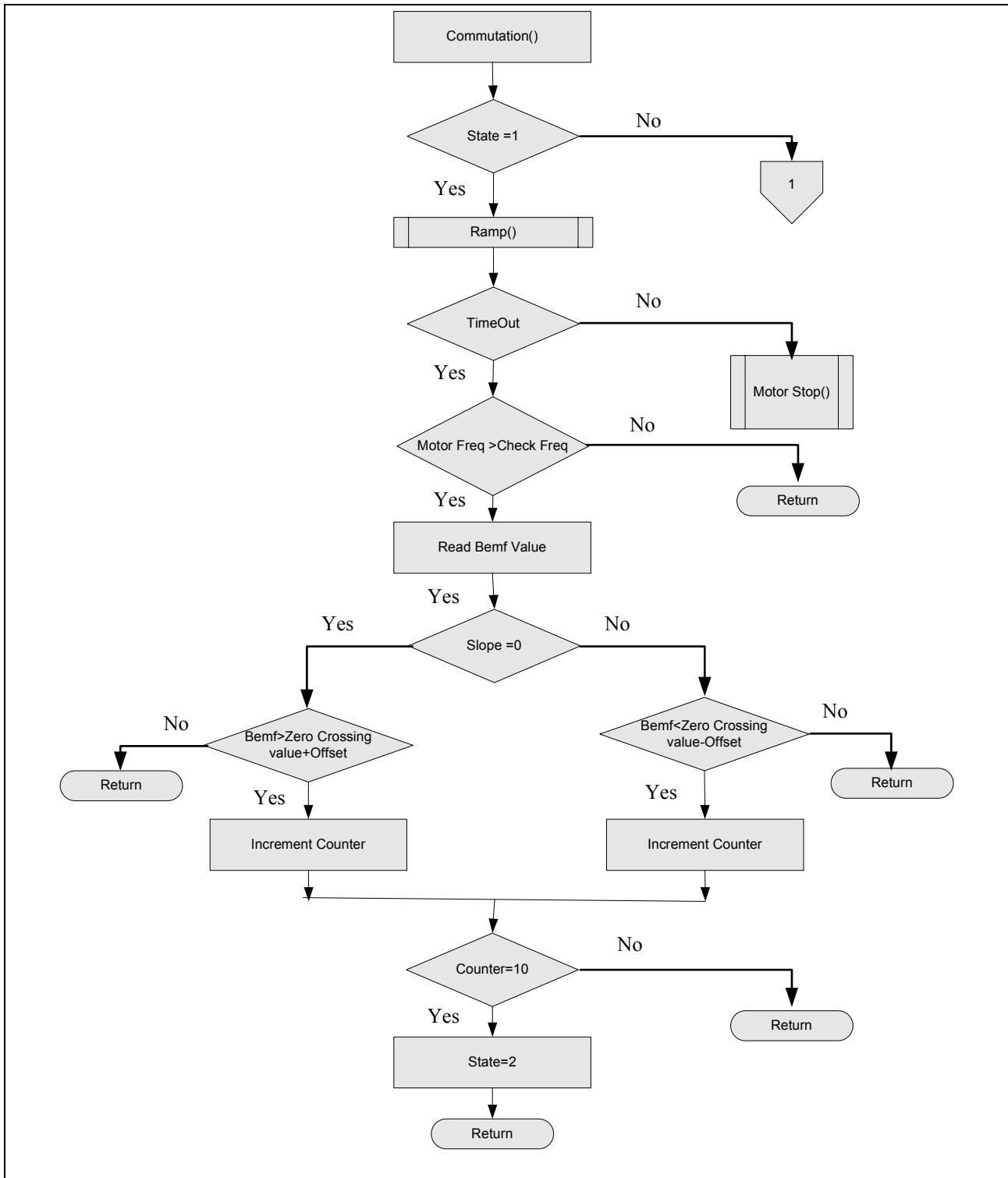
If the voltage values on two measurement are greater than zero crossing value for positive slope (slope =1) or less than zero crossing value for negative slope (slope =0), then timer T12 will be stopped and the timer value is captured. Now the commutation pattern should be updated into the MCMOUT register after half of the T12 timer value. To accomplish this, the following steps needs to be done.

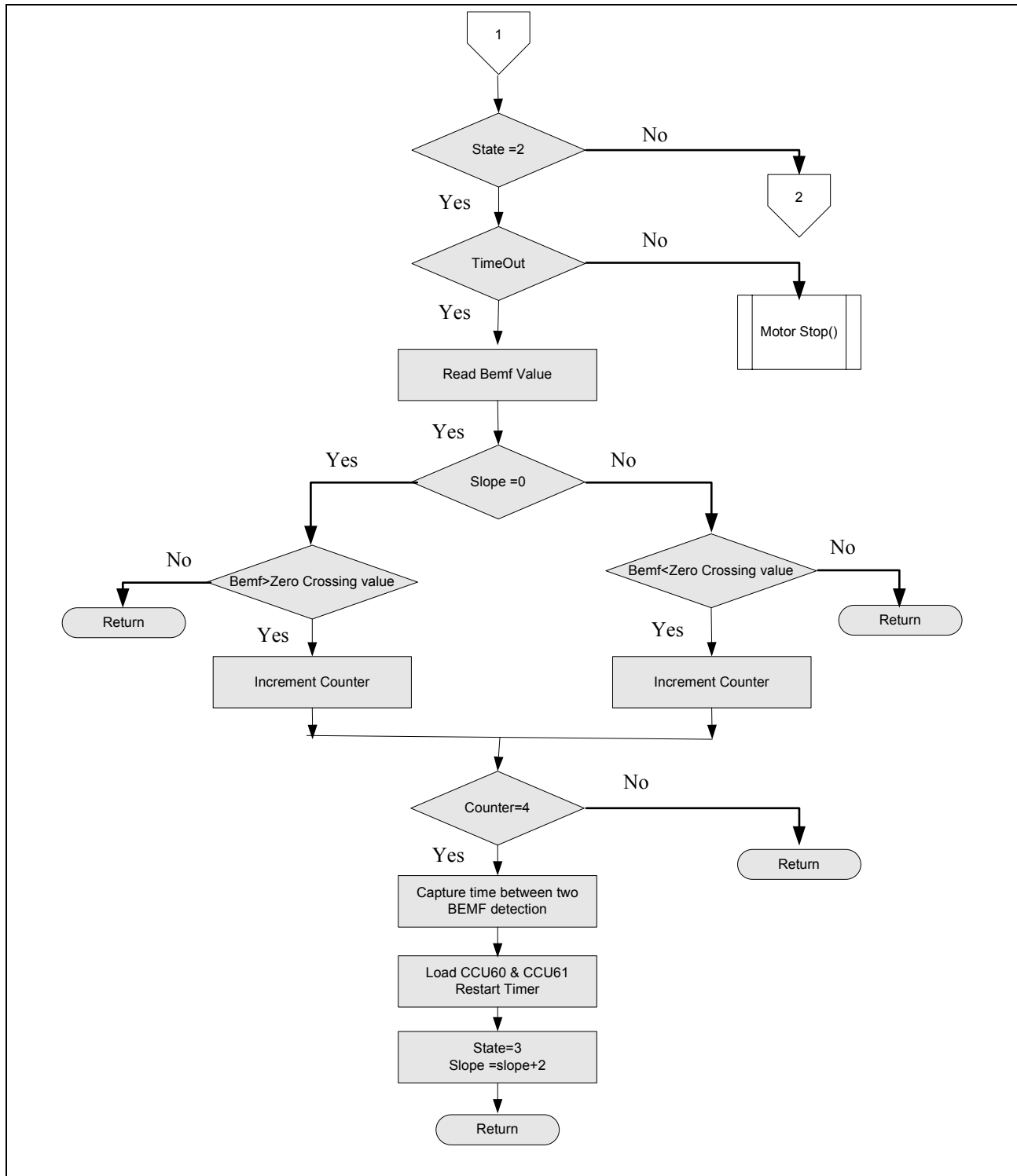
- Half of the T12 timer value should be loaded into the CCU61 compare register
- Timer T12 should be reset and started again

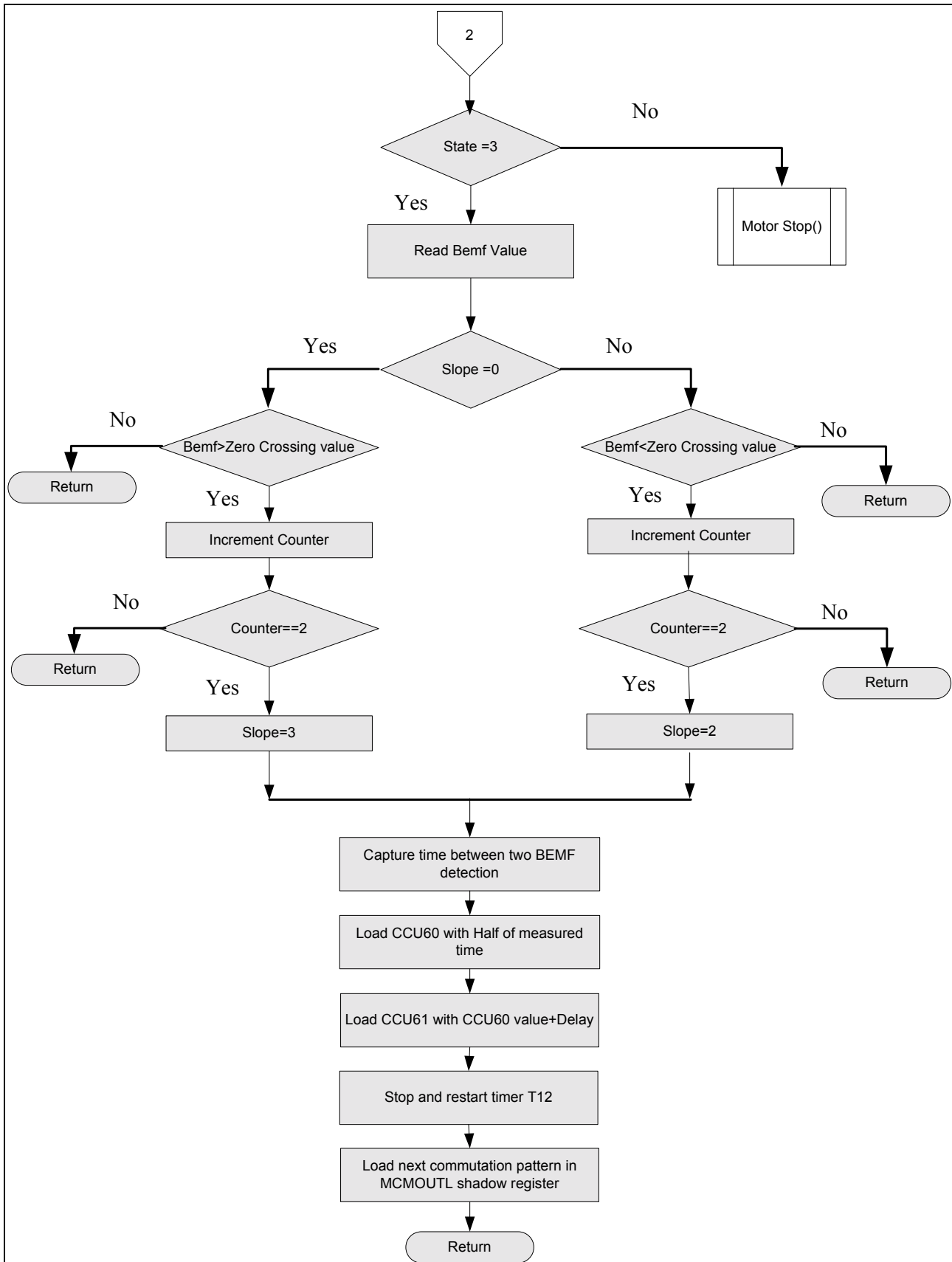
The MCMOUT shadow transfer will happen during CCU61 compare match event. Also the next commutation pattern is loaded into the MCMOUT shadow register after the MCMOUT shadow transfer happened.



**Figure 11 BEMF Detection Timing Diagram**



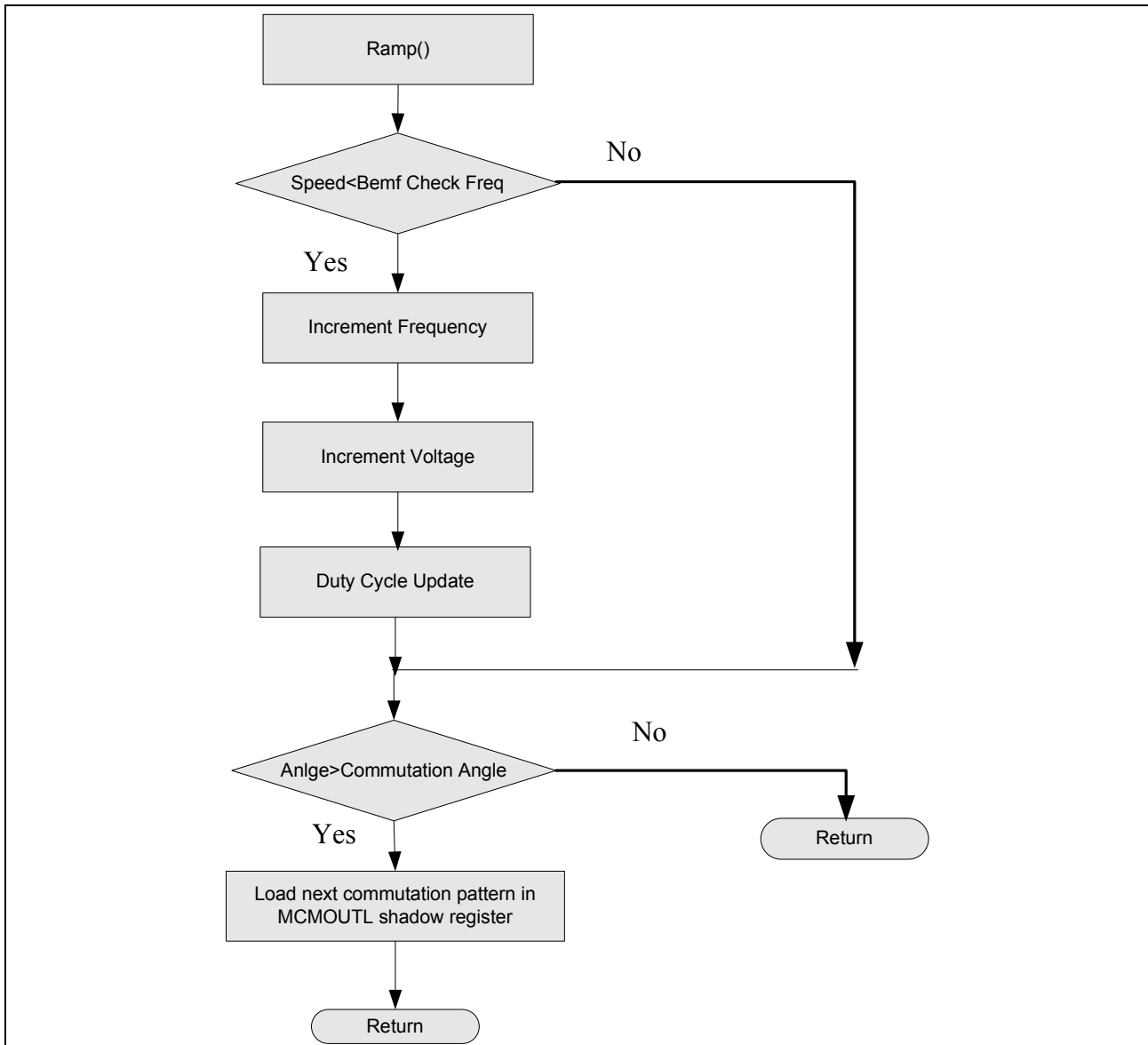




**Figure 12 Flow Chart of Commutation Function**

### 3.3.1.2 Rampup Function

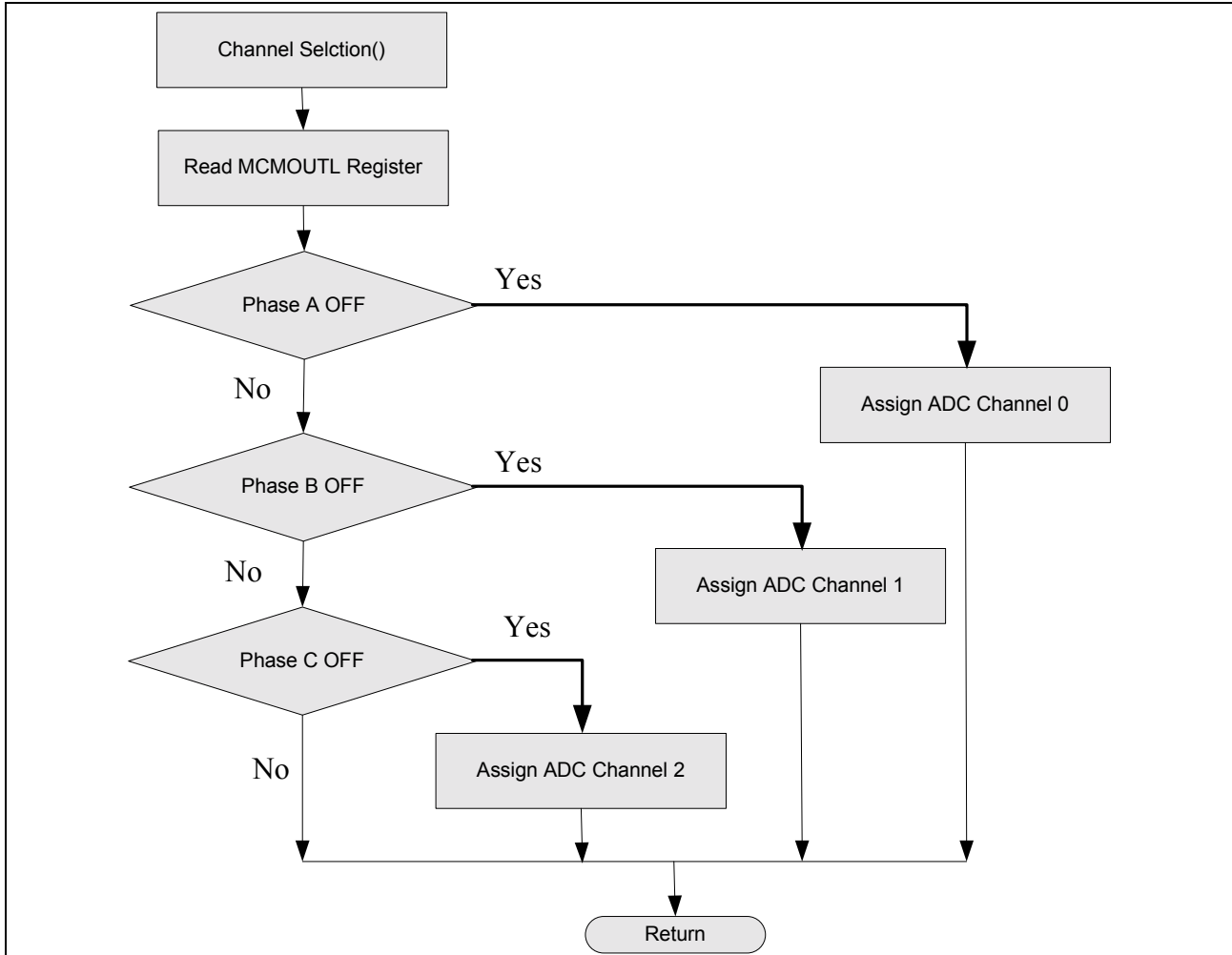
This function will be called from the commutation function when the motor is running in open loop mode. During this function call the applied voltage and speed are increased based on a voltage increment value and a frequency increment value.



**Figure 13 Open loop Rampup Function**

### 3.3.2 Channel Selection

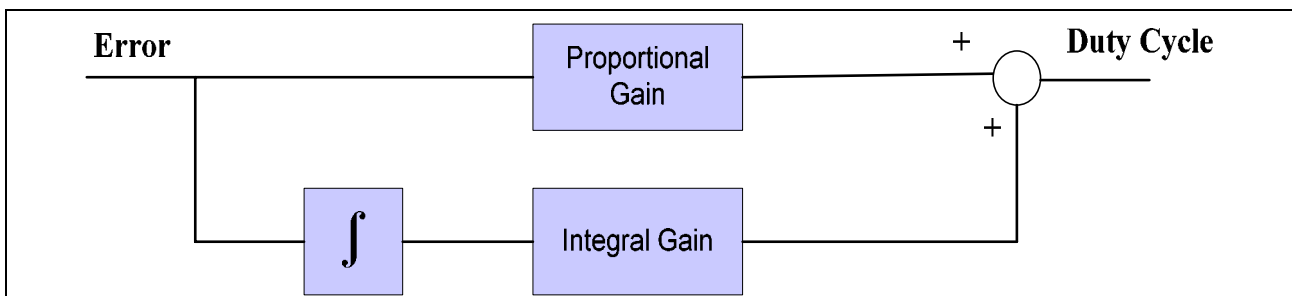
This function is used to find non-energized Phase winding and select appropriate ADC channel, to measure the induced voltage.



**Figure 14 Channel Selection Function**

### 3.3.3 PI Controller

A PI controller is used for regulating the speed. The error difference between the reference speed and the actual speed is fed to the controller. The PI controller functionality is shown in Figure 15.



**Figure 15 Block Diagram for PI Controller**

In continuous time domain, the duty cycle output is given by,

$$\text{Duty Cycle} = K \text{ error} + K \int \text{error} dt \quad \text{.....(3.1)}$$

In discrete time domain, the PI controller is implemented as described by the following equations.

$$\begin{aligned} Y_n(k+1) &= Y_n(k) + K_i * e(k) \\ Y(k+1) &= Y_n(k+1) + K_p * e(k) \end{aligned} \quad \text{.....(3.2)}$$

Where,

$K_i$  - Integral Gain

$K_p$  - Proportional Gain

$e(k)$  - Error value

$y(k+1)$  - Next computed duty cycle

$y_n(k)$  - Integrated error value till last computation

$y_n(k+1)$  - Current Integrated error value

The actual  $K_p$  and  $K_i$  values are scaled and will be used in target as follows:

$$\begin{aligned} K_p &= k_p * 2^{15} / 64 \\ K_i &= k_i * 2^{15} \end{aligned} \quad \text{.....(3.3)}$$

Where,

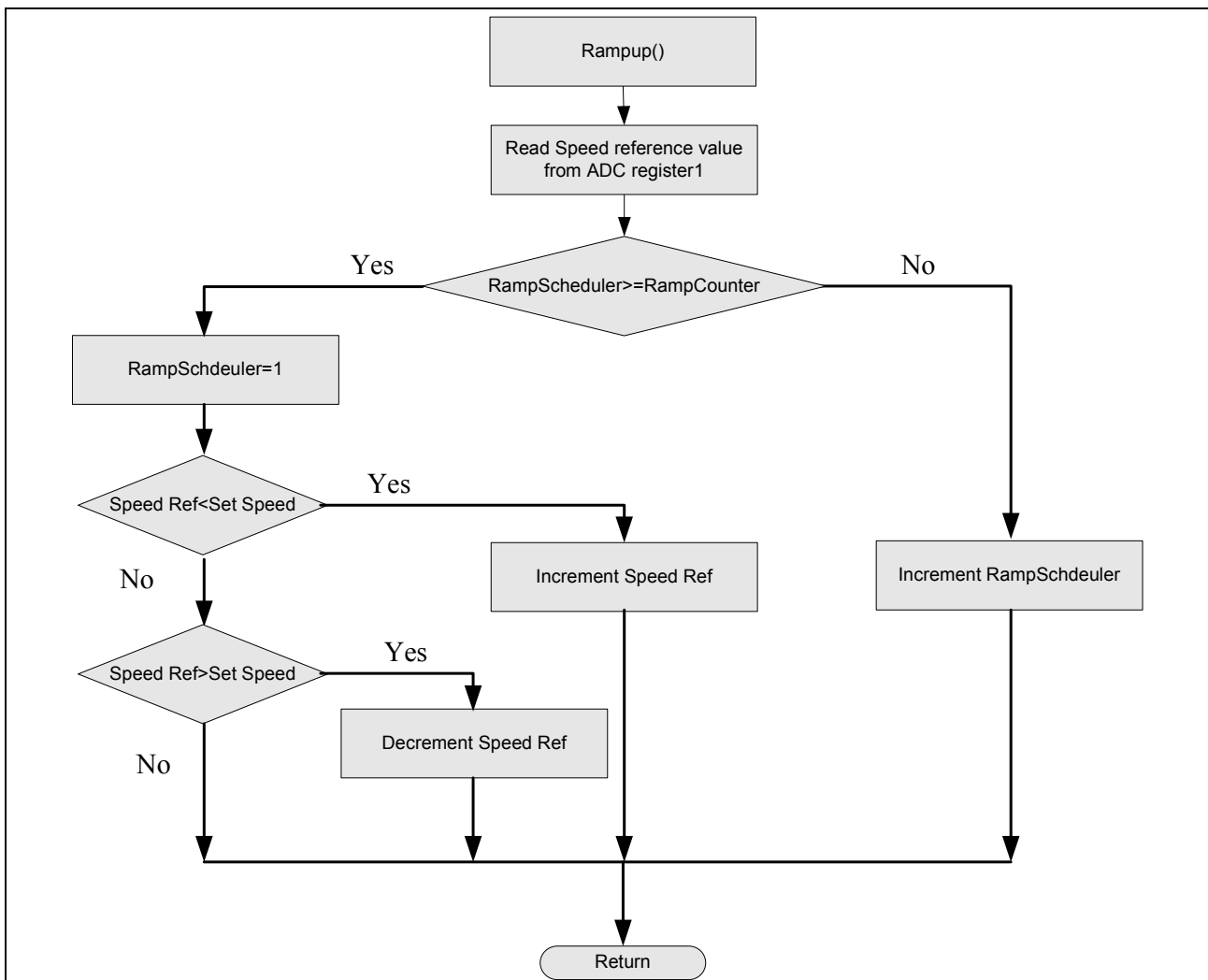
$k_p$  and  $k_i$  are the Scaled Proportional and Integral Gain values used in software.



### 3.3.4 Speed Rampup Function

In this function the motor speed reference value is determined based on user input via POT. The speed slew rate (RPM/Second) depends upon the function call rate and the ramp scheduler value. The function call rate is fixed for particular PWM frequency, for 20 kHz function call rate is 50  $\mu$ s. So the ramp scheduler value will be calculated based on required speed slew rate and PWM frequency.

$$\text{RampScheduler} = \frac{1}{\text{Required Slew Rate} * \text{Function Call Rate}} \quad \text{..... (3.4)}$$

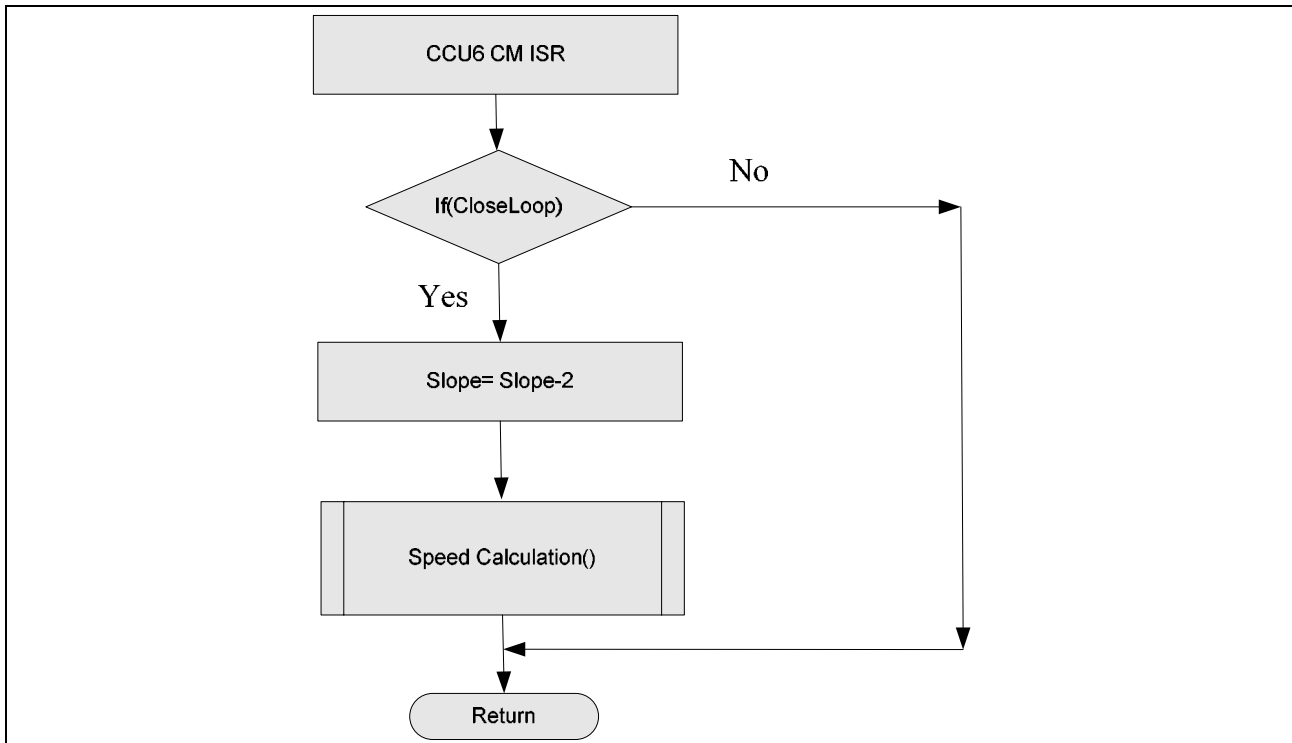


**Figure 16 Speed Rampup Function**

### 3.4 CCU6 CCU62 Compare Match ISR

During this interrupt routine the Speed calculation function will be executed. The speed calculation needs the time between zero crossing values. The time will be ascertained by Timer12 (CAPCOM6E). On every zero crossing, the Timer T12 will be stopped and the time between zero crossing values is captured.

To reduce the measurement errors, the time between two zero crossing events is averaged over (6 \* Pole pairs) measured values.



**Figure 17 CCU62 Compare Match ISR**

### 3.5 T12 PM and CTrap ISR

If the CCU6 trap input becomes active or the T12 Period match (Timeout) occurred the motor will be stopped. So during this ISR the Motor Stop function will be called.

### 3.6 Timer T2 ISR

During the interrupt routine, the speed reference value is calculated based on POT input. Also if there is any request via UART (START, STOP, GET and SET), the system will respond to that request.

UART messages are 4 bytes in length with a Message ID at the beginning

**Table 4 UART Request Message**

Action	UART Message [B0 B1 B2 B3]			
START	0x05	00	00	00
STOP	0x06	00	00	00
Read Value	0x8C	Address of Variable	00	00
Set Value	0x80	Address of Variable	Value	

**Table 5 Variable Location**

Variable Name	Location
Speed Start Ref	0x28
Speed End Ref	0x8C
Rampup Counter	0x8E
Proportion Gain (Kp)	0x92
Integral Gain (Ki)	0x94
Motor Speed	0x90

[www.infineon.com](http://www.infineon.com)