International
IOR Rectifier

DEVELOPER'S GUIDE

International Rectifier • 233 Kansas Street, El Segundo, CA 90245 • USA

IRMCK203
Application Developer's Guide

February 19, 2004
Version 1.0

IR WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245 Tel: (310) 252-7105

Table of Contents

List of Figures

List of Tables

1 Introduction

This document is provided as a supplement to the datasheet for the IRMCK203. It provides detailed information about the internal design and external interfaces of the product and describes how to configure the operation to conform to the requirements of a custom application. This document is intended for engineers who are developing an application using the IRMCK203 digital control IC.

The document is divided into three main sections. In the Concepts section, system design concepts are presented and theory of operation is described in detail. This section provides the background needed to begin IRMCK203 application development. The Techniques section provides practical "how-to" information, tips and examples to assist with the development process. The Reference section provides a complete definition of the host register map with a short description of each register and field. The registers are listed in sequential order for easy reference.

1.1 Constraints

The following are constraints for use of the IRMCK203 with a custom hardware system.

Analog Interface

The IRMCK203 has a built-in interface to the ADS7818 (BurrBrown) serial A/D converter (12 bit). Dc bus voltage feedback, external speed reference and Leg Shunt current can be obtained via ADS7818 in conjunction with MUX circuitry. An analog feedback application example is given in Section 2.4.2.

Current Feedback Interface

The IRMCK203 has a built-in interface circuit for two IR2175 motor current sensing high voltage ICs. With two IR2175 and two shunt resistors, the motor phase currents can be obtained for motor control purposes. A 10-bit resolution of current feedback data can be obtained. The practical power level limit for using shunt resistors is up to 3.7 kW. For a higher horsepower application, a resistor shunt becomes impractical due to power dissipation of shunt resistors (insert in series between motor and drive).

The IRMCK203 provides other means of current feedback through the use of an ADS7818 (BurrBrown) serial A/D converter and Inverter Leg Shunt resistor. Inverter Leg Shunt currents can be used (reconstruction of phase current inside IRMCK203) instead of IR2175 current feedback. However, the Leg Shunt option is recommended only for an Inverter switching frequency less than 10KHz.

1.2 Application Connections

Figure 1 shows a typical application connection block diagram. In order to complete a Sensorless drive control, all necessary components are shown in connection to IRMCK203. A fully self-contained drive evaluation board (IRMCS2031) based on the IRMCK203 Digital Control IC is available for drive performance evaluation.

The figure shows a typical hardware configuration. Users can customize the design without the effort of modifying code.

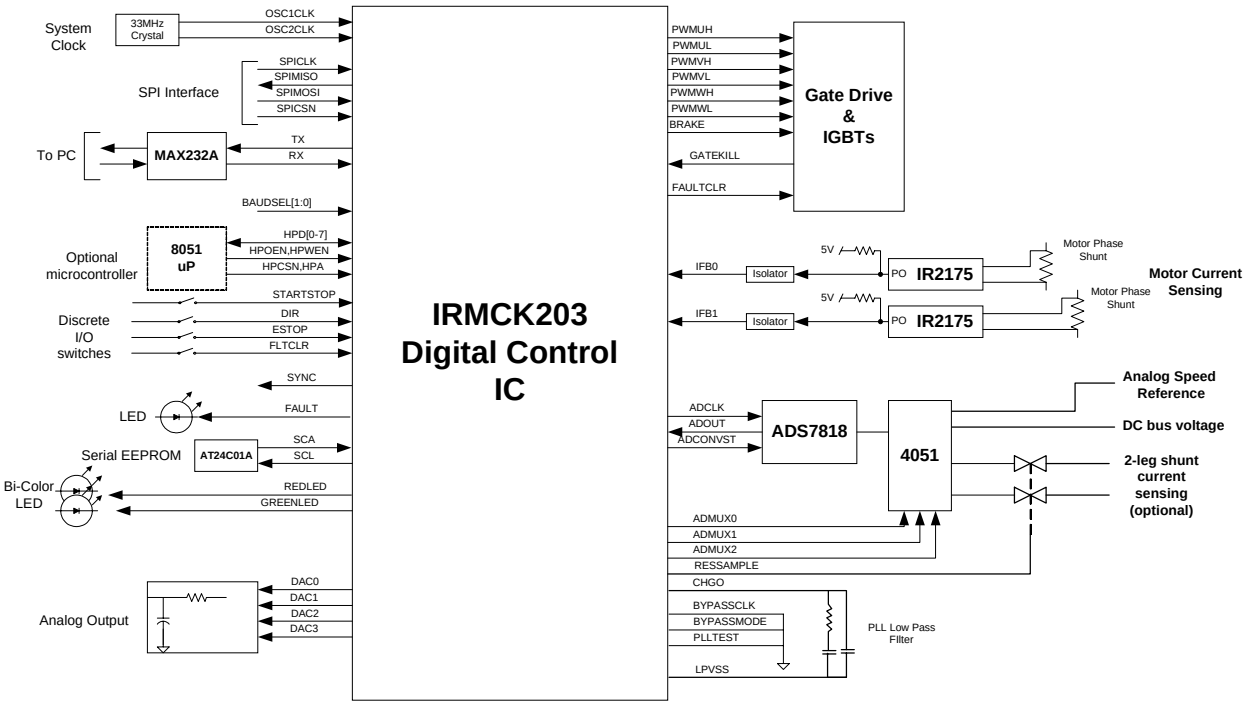


Figure 1. Typical Application Connections of IRMCK203

2 Concepts

Figure 2 shows the block control structure of the IRMCK203.

2.1 Regulators

2.1.1 Closed Loop Current Control

Two Proportional plus Integral (PI) type current regulators with output limits and Anti-windup control are provided for torque and flux regulation of motors. Torque current reference is supplied by Speed regulator output and Flux current reference is set to zero in order to achieve the maximum torque per ampere for a Surface-Mounted Permanent Magnet motor. The current regulator outputs are modulation depths. The modulation depths are fed to a Space Vector PWM modulator via a vector rotator (converts dc to ac waveform). Refer to Appendix A for a description of the Space Vector PWM module.

2.1.2 Closed Loop Velocity Control

A PI speed regulator with output limits (torque current limit) and Anti-windup control is provided for speed regulation. The speed reference is supplied by the Ramp block (as shown in Figure 2). Both speed acceleration rate and deceleration rate can be adjusted. In addition, the Ramp block provides minimum speed protection in order to ensure optimal speed control performance for Sensorless operation. The Ramp block input is the user-desired speed reference which can be obtained internally from the host register interface or externally via the A/D interface as shown in Figure 2. Details on how to set up external speed control mode (Standalone mode) are provided in Section 3.2.

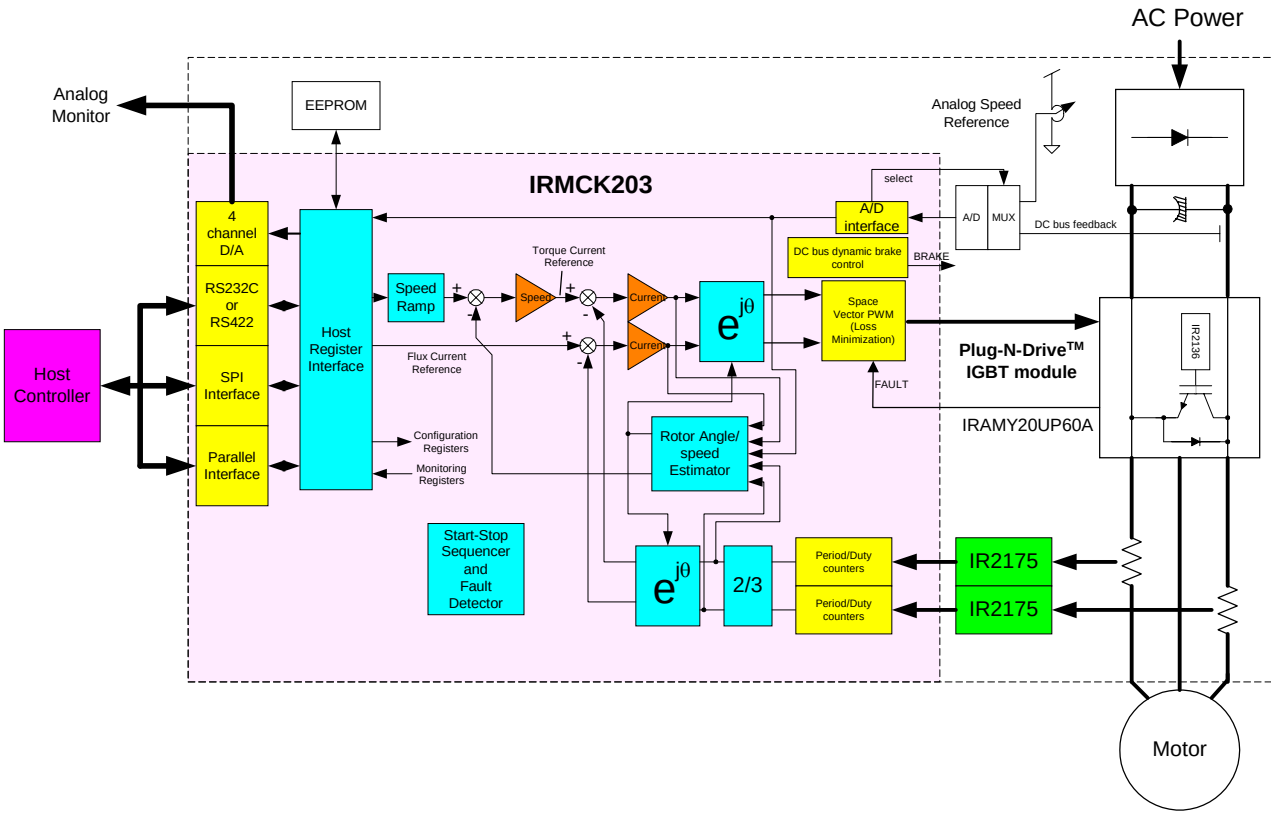


Figure 2. Detailed Control Structure

2.1.3 Rotor Angle Estimation

Motor shaft angle information is required for high performance control of Permanent Magnet motors. The IRMCK203 Sensorless control IC contains a motor shaft angle estimator, which provides shaft angle and motor speed information. There is no need for encoder or hall sensor element feedback.

2.1.4 Start-Stop Sequencer and Fault Detection

A Start-Stop sequencer provides total drive sequencing for handling start-stop and start-up failure retry functions. There are various fault triggers (detailed in Section 2.6) to ensure that the drive is protected under various fault conditions.

2.2 Current Feedbacks

2.2.1 Using IR2175

Two channels of current feedback interface logic are provided in the IRMCK203. Each module measures the incoming varying duty period of the 130 kHz carrier frequency signal at the IR2175 output. Measurement is performed for both carrier frequency period and on duty period at the same time using fast counters. Counting frequency is 133 MHz with a 33.3 MHz system clock.

The IR2175 is the unique high voltage IC capable of measuring the motor phase current through an associated shunt resistor, which can generate ±260mV voltage range. The output of the IR2175 is an open drain with a 130 kHz fixed carrier frequency where the duty variance is linearly proportional to ±260 mV input voltage. The counting frequency is 133.3 MHz when the system clock crystal frequency is 33.3 MHz, which yields 10-bit resolution of the current measurement data from the IR2175. A more detailed description of the IR2175 can be found in Appendix B.

2.2.2 Using Inverter Leg Shunt

The IRMCK203 provides another means of current feedback through the use of an ADS7818 (BurrBrown) serial A/D converter. Inverter Leg Shunt (install in Low side) currents can be used instead of IR2175 current feedback for Sensorless motor control. The minimum requirement is two Leg Shunt feedbacks (V and W phases). An application example is given in Section 2.4.2 for interfacing Leg Shunt currents to the IRMCK203 digital control IC.

In the IRMCK203, the selection of current feedback is done via a user configuration parameter (provided in the Motor commissioning tools). The Leg Shunt option is recommended for Inverter switching frequencies less than 10KHz.

2.3 Communication

The IRMCK203 contains a rich set of externally addressable "Host" registers documented in Section 4 of this guide. There are three physical interfaces that can access the Host Registers: RS-232, SPI and Host Parallel.

2.3.1 RS-232 Serial Interface

The slowest of the three, the Serial Interface, is used for inter-board communications typically using cables as the connection medium. The IRMCK203 implements an error detecting protocol layer that facilitates maintaining the integrity of the Host Registers. Prior to updating any Host Register, the incoming data must match a checksum string to detect single bit errors. Please refer to the RS-232 protocol documentation in Section 4.1.3 for the specific protocol definition. The RS-232 Serial Interface supports four baud rates based on the signal levels on pins 30 and 42 of the IRMCK203, as shown in Table 1.

BAUDSEL1 PIN 42	BAUDSEL0 PIN 30	Resulting BAUD Selection
0 or low	0 or low	19.2 K BAUD
0 or low	1 or high	38.4 K BAUD
1 or high	0 or low	57.6 K BAUD
1 or high	1 or high	1 MEG BAUD

Table 1. BAUD Selection Table

The RS-232 interface implements a byte serial physical layer in addition to an error checking protocol layer. The coding of the bit-serial data is US ASCII, 8 data bits, 1 stop bit and no parity.

Table 2 describes the physical layer signals of the RS-232 interface.

Signal Name	Direction	Description
TX	Output	A bit-serial signal originated by the IRMCK203 in response to a microprocessor-generated request.
RX	Input	Bit-serial data sent to the IRMCK203 by the microprocessor to interrogate one of the Host Registers.

Table 2. External RS-232 Signal Description

2.3.2 SPI Interface

The SPI Interface is also a byte serial interface, but can operate at much greater transfer rates than the RS-232 interface. Bit rates of up to 8 MHz can be achieved. The SPI Interface performs a serial byte read and write in a "full duplex" mode. Refer to the SPI Access documentation in Section 4.1.2 for the protocols required to access the Host Registers, and the SPI timing section of the IRMCK203 datasheet for the physical layer specifications.

Table 3 describes the physical layer signals of the SPI interface.

Signal Name	Direction	Description
SPICLK	Input	Serial clock generated by the SPI master logic.
SPIMOSI	Output	Serial data: Master Input and Slave Output.
SPIMISO	Input	Serial data: Master Output and Slave Input.
SPICSN	Input	Chip Select signal. Used to qualify the SPICLK, SPIMISO and SPIMOSI signals.

Table 3. External SPI I/F Signal Description

2.3.3 Host Parallel Interface

Designed to transfer bytes in a bit parallel fashion, this is the fastest interface of the three. The Host Parallel interface is compatible with all popular microprocessors, including Motorola and Intel based bus protocols. Refer to the Parallel Access documentation in Section 4.1.1 for the protocols required to access the Host Registers, and the Host Parallel timing section of the IRMCK203 datasheet for the physical layer specifications.

Table 4 describes the physical layer signals of the Host Parallel interface.

Signal Name	Direction	Description
HP_nOE	Input	When logic low, or 0, indicates the beginning of a parallel data transfer cycle.
HP_nWE	Input	When logic low, or 0, indicates that the data/address transfer cycle is a write cycle, with data being sourced by the microprocessor. When high, the data cycle is a read cycle, with data being sourced by the IRMCK203
HP_D [7:0]	Input/Output	An 8-bit wide data bus.
HP_A	Input	Address attribute signal. When high, or a logic 1, indicates that the data on the HP_D[7:0] bus is a address to be loaded into the IRMCK203 address register.

Table 4. External Host Parallel I/F Signal Description

2.3.4 Synchronization of PWM Cycle to an External Microprocessor

A dedicated SYNC signal is provided on the IRMCK203 (pin 52) that allows synchronization of the internal IRMCK203 logic to an external microprocessor. This synchronization is useful when external microprocessor control loops are implemented. Also, an external trace buffer could be implemented to interrogate various nodes in the IRMCK203 while the IRMCK203 is actively controlling the motor.

The SYNC signal has a long pulse width suitable to connect to an edge or level sensitive microprocessor interrupt input pin. The low going edge of this pulse is an indication to the microprocessor that the IRMCK203 is starting a new PWM cycle. Refer to the ADC System Level Timing section of the IRMCK203 datasheet for specific timing information. Both the SPI and Host Parallel Interfaces are suitable for PWM Cycle and trace buffer synchronization.

The SYNC signal offers the microprocessor a timing window to access the entire Host Register set. The number of SYNC pulses per PWM load can be configured using the support tools described in Section 3.

The SYNC pulse width is suitable for connecting opto-isolation circuitry between the IRMCK203 and the microprocessor.

2.4 External Interfaces

This section describes the external interfaces supported by the IRMCK203 in addition to the host register interface described in Section 2.3. These include the discrete I/O interface used for standalone operation and the analog I/O interface provided for diagnostic purposes.

2.4.1 Discrete I/O External Interface

The discrete I/O external interface signals provide a means of controlling basic motor operation without using the host register interface. In this mode of operation, the analog reference (described later in this section) is used to directly control the target speed.

Figure 3 shows a schematic diagram of the discrete I/O signals. The signals are described in Table 5.

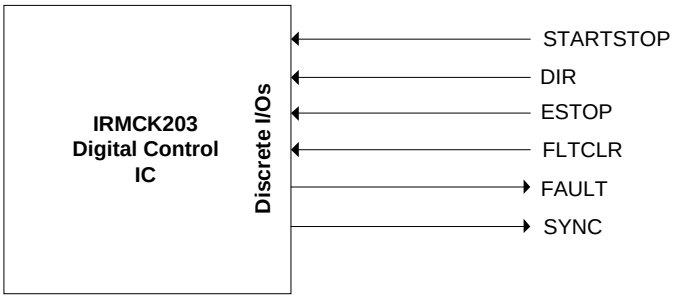


Figure 3. Discrete I/O Signals

Signal Name	Direction	Description
STARTSTOP	Input	Motor start/stop control. A positive edge transition of this signal starts the motor and a negative edge transition stops the motor.
DIR	Input	Motor direction control. 1 = forward, 0 = reverse. This signal is latched when the motor is started, so that changing it while the motor is running has no effect.
ESTOP	Input	Emergency stop. When this signal is set to "1", PWM is unconditionally disabled. This signal overrides the START/STOP control
FLTCLR	Input	A 1 µsec pulse on this signal clears a drive fault condition. Equivalent to setting the FltClr bit of the FaultControl register (see Section 4.2.6).
FAULT	Output	This signal indicates the presence of a drive fault condition. The level is high when any of the bits in the FaultStatus register are set (see Section 4.3.4).
SYNC	Output	This signal is held low for 2 µsec on each PWM period. (The falling edge indicates the start of the PWM period.)

Table 5. External Interface Signal Description

NOTE: When the ExtCtrl bit in the SystemConfig register is set to "0", the ESTOP and negative edge of the START/STOP signals are functional, but all other external interface signals are inactive.

To configure the discrete I/O interface, write a "1" to the ExtCtrl bit in the SystemConfig host write register to enable the external interface pins. (Refer to Section 4.2.7 for more information about the SystemConfig register.)

2.4.2 Analog I/O Interface

IRMCK203 provides analog input capability through the use of the ADS7818 A/D converter and MUX circuitry. The intended inputs are speed reference, dc bus voltage and two Inverter Leg Shunt currents.

Analog Input

Figure 4 shows the typical hardware configuration for the analog input interface. The multiplexor input A0 (shown on the diagram) accepts voltages in the range 0 – 5V, with two possible mappings:

- 2.5V = zero speed (0 digital count), 0V = max speed (16, 383 digital count)
- 2.5V = zero speed (0 digital count), 5V = max speed (16, 383 digital count)

The example implements the first of the two mappings (0V max speed), supplying a +15 volt analog reference for an external variable resistor. (The DIR signal controls the motor direction, as described in Table 5.)

In this example circuit, the IRMCK203 automatically scans through A/D conversion of all four channels at the beginning of each PWM cycle (SYNC output). The v and w phase currents followed by dc bus voltage and speed reference are scanned in. In this example, the dc bus feedback gain is 100 times attenuation. The Leg Shunt amplifier gain for this example is 7.97 and the A/D converter scaling is 4095 digital counts per 5V. This information is required during drive commissioning for scaling of dc bus voltage and current feedback.

Leg Shunt feedback can be eliminated if IR2175 is intended for current feedback. The interface to the IR2175 is straightforward and can be found in the IR2175 data sheet.

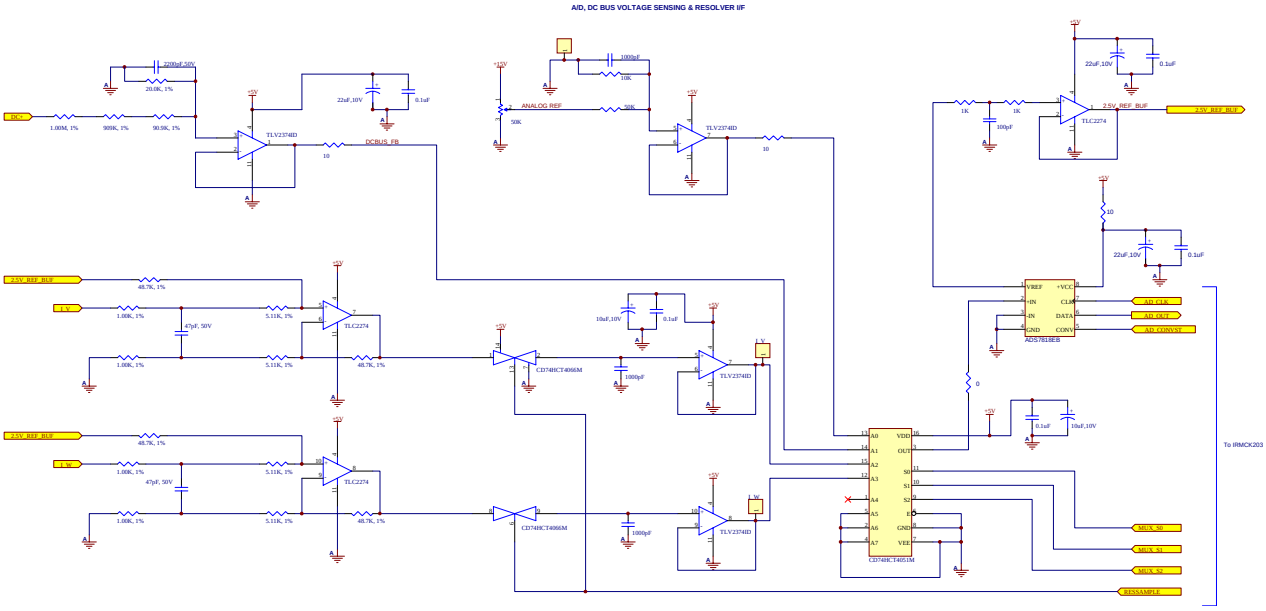


Figure 4. Analog Interface Example

Analog Output

The diagnostic D/A interface provides four sources of diagnostic data and is intended for use with external RC filters for oscilloscope display. The user can select one of four sets of data sources by setting the value of the DacSel register in the D/AConverter write register group (see Section 4.2.14) as shown in Table 6.

DacSel Value	Selected Data Sources	
0	0	Flux
	1	Rotor angle
	2	Torque current
	3	Closed loop status
1	0	DC bus voltage
	1	Alpha voltage
	2	Torque current reference
	3	Motor speed
2	0	Q-axis command voltage
	1	D-axis command voltage
	2	Alpha current
	3	Beta current
3	0	Flux magnitude
	1	Current error at parking
	2	Parking diagnostic flag
	3	W-phase current

Table 6. Analog Output Data Source Selection

Each signal is encoded as a pulse-width modulated 8-bit value output at a frequency of 128 KHz. Therefore, hardware filtering is required to require to extract the actual signal. The data values are updated on each sync pulse. The values for each data source are scaled so that the valid range is represented as an 8-bit unsigned value. For

example, the values of Q-axis and D-axis command voltage, which have an actual range of –16,384 to 16,383, are rescaled to the range 0 – 255 (so that 0 represents –16,384 and 255 represents 16, 383).

2.5 Sequencing Control

Sequencing control is provided in the IRMCK203 system to facilitate basic I/O sequencing. The signals shown in Table 7 can be directed either by local discrete I/O pins or the host register interface. STOP is always activated by either the host interface register or the local START/STOP input pin.

Signal	Description
START	Start motor signal from host or external user interface.
STOP	Start motor signal from host or external user interface.
ESTOP	This signal, which is not shown in figure 1, stops the motor unconditionally regardless of the state.
FAULT	Indicates a pending FAULT condition. It is cleared upon FLTCLR assertion.
FLTCLR	Clear pending FAULT.
Start OK	Signal from startup control module that indicates a successful startup occurred
Startup Fault	Indicates a failed startup attempt occurred
Retries	Running count of the number of retries attempted during the startup sequence.
Max Retries	User programmable register setting indicating the maximum number of retries. The maximum value of retries can be 16. Retry is disabled when this value is 0.
Overvoltage Undervoltage Overcurrent Overspeed	Fault conditions.

Table 7. Sequencing Control Signals

Internally, the IRMCK203 has three states: Stand-By or STOP state, RUN state, and FAULT state. Transitioning to each state can be caused either by initiation of the I/O pins described above or internal drive conditions such as overcurrent, overvoltage, etc. The state diagram is shown in Figure 5.

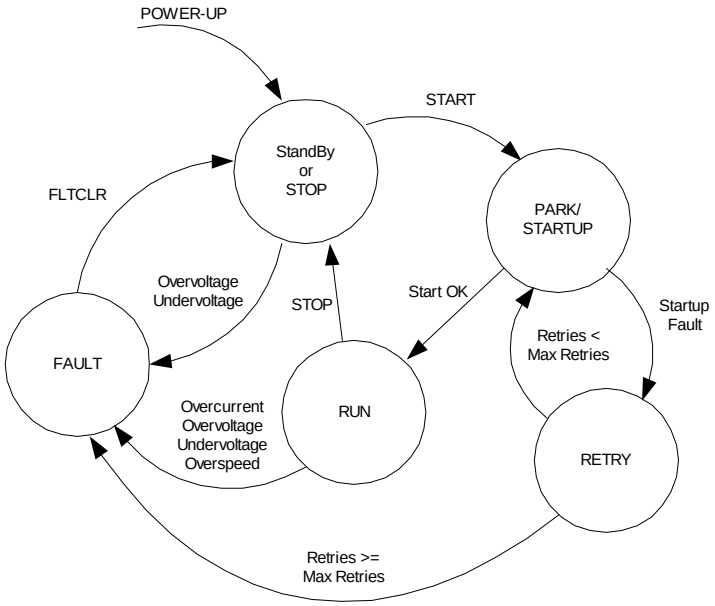


Figure 5. State Diagram and Sequencing

2.6 Fault Handling

The IRMCK203 system has built-in drive fault and protection features. Table 8 summarizes the types of drive fault conditions.

Fault	Status indication on Host Register Interface	Description
Overcurrent /Overtemperature	FltStatus Read Register, field GatekillFlt = 1	Overcurrent or overtemperature occurred. The IGBT gate driver (IR2136) disables gate drive outputs, momentarily latches a fault condition and asserts GATEKILL to the IRMCK203. This activates the fault latch inside the IRMCK203.
Overvoltage	FltStatus Read Register, field OvFlt = 1	Overvoltage of the DC bus occurred. Only the fault latch inside the IRMCK203 is activated.
Overspeed	FltStatus Read Register, field OvrSpdFlt = 1	The speed of the motor exceeded the maximum speed. Only the fault latch inside the IRMCK203 is activated.
Overrun	FltStatus Read Register, field ExecTmFlt = 1	The computation of algorithm exceeded the selected PWM carrier frequency period. Only the fault latch inside the IRMCK203 is activated.
Low voltage	FltStatus Read Register, field LvFlt = 1	The bus voltage dropped below a certain level (determined by the dc bus feedback scaling). Only the fault latch inside the IRMCK203 is activated.
Zero speed	FltStatus Read Register, field ZeroSpdFlt = 1	When speed is less than MinSpd/2 (half minimum speed) for a continuous period of 2 seconds, the zero speed fault occurs. Only the fault latch inside the IRMCK203 is activated.
Startup retry failure	FltStatus Read Register, field RetryFlt = 1	After a configured number of start-up failures (determined by register NumRetries in the StartupRetrial write register group), this fault occurs. Only the fault latch inside the IRMCK203 is activated.
Phase loss	FltStatus Read Register, field PhsLossFlt = 1	This fault indicates that the drive to motor phase connection may be loose. Only the fault latch inside the IRMCK203 is activated.

Table 8. Drive Fault Conditions

When any drive fault occurs, the PWM output is disabled and the gate signals from the IRMCK203 device are negated. This condition remains latched until Fault_Clear action is undertaken by the user. Fault_Clear, a level sensitive signal event, can be initiated either through the FltClr bit in the FaultControl host register or the FLTCLR discrete I/O external interface pin. For more information about the FaulControl and FaultStatus registers, refer to Sections 4.2.6 and 4.3.4, respectively.

When a fault occurs, the LED indication is as follows: REDLED = 1, GREENLED = 0.

2.6.1 Gatekill Structure and Overcurrent/Overtemperature Fault

For example, the IRMCS2031 design platform for IRMCK203 has an advanced intelligent power module (IRAMX16UP60A) rated at a 600V/16A. This IGBT module contains an integrated high voltage gate drive IC (IR2136) with a thermistor.

A ground fault protection circuit is also equipped on the IRMCS2031. The signal is fed to an opto-coupler device to trigger the signal to IRMCK203 pin 37, GATEKILL.

When an overcurrent condition occurs, GATEKILL is asserted and momentarily latched within the IR2136 for the programmed period, which is approximately 9 milliseconds. After this period, the pending fault is automatically cleared. Meanwhile, the triggered GATEKILL assertion latches and inhibits all PWM output gate signals off the IRMCK203 until the user initiates a FAULT CLEAR action.

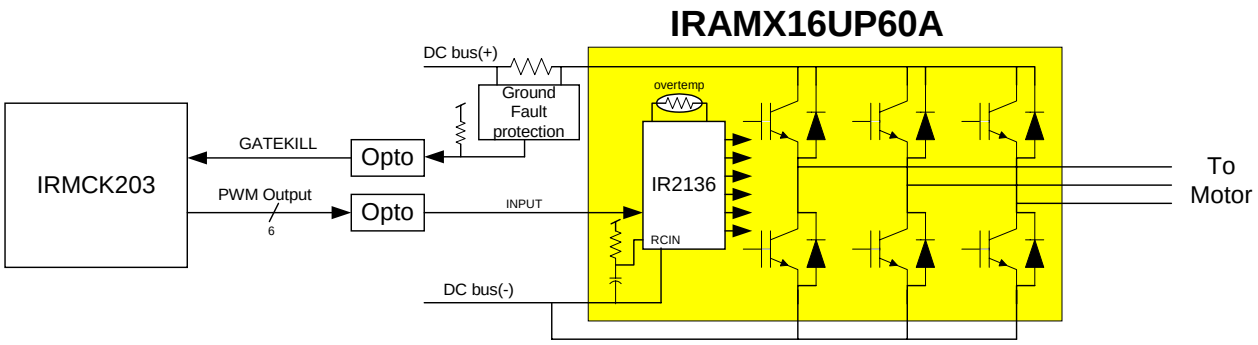


Figure 6. Protection Circuit Block Diagram

Figure 6 shows the protection circuit diagram. The IGBT module contains an RC circuit connected to RCIN input of the IR2136, which automatically initiates FAULT CLEAR in 9 milliseconds after assertion of FAULT. The IGBT module also contains an overtemperature protection circuit, which shuts down all IGBTs and performs automatic FAULT CLEAR as well. Overtemperature protection can be enabled by adding a 6.8 kOhm external resistor. The threshold level is set at approximately 110°C. The IRMCS2031 contains the ground fault protection circuit on the high side DC bus (+) node. The circuit senses positive ground fault current and sends a trigger signal to GATEKILL via a wired-OR FAULT signal.

Once any fault condition is detected, the IR2136 inside of the IGBT module momentarily latches the condition and initiates FAULT output and shutdown of all six IGBTs. Upon receiving the FAULT signal at its GATEKILL input, the IRMCK203 disables all PWM gate signals and latches GATEKILL. It also disables PWM output.

To reset a fault condition, first write a "1" to the "FltClr" bit of the Fault Control register (see Section 4.2.6). This clears the fault in the IRMCK203. Then write a "0" to the FltClr bit to re-enable fault processing. Note that PWM output does not automatically restart after a fault condition is cleared.

2.6.2 DC Bus Faults and DC Bus Braking

The DC bus signal is employed for dc bus overvoltage, undervoltage protection and Brake control. It is also used for compensation of motor controller scaling internal to the IRMCK203.

It is crucial to design a suitable dc bus feedback scaling for proper drive protection. The dc bus voltage can be acquired via the ADS7818 A/D converter. The input of ADS7818 maps 0 - 5 Volts into 0 - 4095 digital counts. The overvoltage and undervoltage trip levels are given in Table 9. The analog scaling (amplifier gain) of the dc bus is restricted by the desired voltage trip levels. Therefore, the signal conditioning (amplifier gain) of dc bus voltage feedback needs to be considered carefully.

	Dc bus feedback IRMCK203 internal digital counts (Fixed)	ADS7818 input voltage	Actual DC bus voltage (assumption: amplifier gain 1/100)
Overvoltage trip fault	3360	4.1 V	410V
Undervoltage trip fault	976	1.2 V	120V
Undervoltage clear level	1152	1.4 V	140V

Table 9. Overvoltage and Undervoltage Trip Levels

Some applications may require power regeneration. Under such circumstances, an external braking circuit (for dynamic braking) can be used to absorb regeneration energy from the motor. The IRMCK203 provides braking control. The braking control utilizes dc bus voltage feedback to determine when to activate and release the braking circuit. The dynamic braking voltage level is given in Table 10. The analog scaling (amplifier gain) of the dc bus presets the brake on-off levels.

Brake Condition	Dc feedback digital counts (Fixed)	ADS7818 Input voltage	Actual DC bus voltage (assumes amplifier gain 1/100)
Brake turn on	3120	3.8 V	380V
Brake turn off	2944	3.6 V	360V

Table 10. Dynamic Braking Voltage Levels

2.7 LED Modes

The operating state of the IRMCK203 is indicated by the LED module. There are three indication modes. Mode 1 indicates successful configuration of the IRMCK203. The LED is green in this mode. Thus, a green LED appears automatically right after power up.

A red LED indicates a drive fault condition. This is Mode 2.

The LED is not lit in Mode 3. This is a hardware fault condition. This means that either configuration data was not transferred to IRMCK203 correctly or the IRMCK203 itself has a hardware problem.

Mode	LED Indication	Description
Mode 1	Green	IRMCK203 configuration has been done correctly and IRMCK203 is functioning normally.
Mode 2	Red	A drive fault condition is pending.
Mode 3	Off	IRMCK203 is not functioning indicating either configuration is not completed correctly and/or IRMCK203 has a hardware problem itself.

Table 11. LED Modes

3 Motor Start-up Supporting Tools

3.1 Start-up Flow

After peripheral circuitry has been implemented for the control IC, a Start-up procedure is provided to guide the user through the commissioning of the user's motor application. Configurable parameters are required to tailor the design to various applications (motor and load). These configurable parameters can be modified via the host register interface (using the ServoDesigner tool) through the communication interface. A design Spreadsheet (Drive parameters translator) is provided to aid the user for ease of drive start-up. Using the Spreadsheet, the user enters high-level parameters such as motor nameplate information, maximum application speed, current limit, speed and speed regulator bandwidth. This high-level user information is translated to engineering parameters (directly used by the drive). Figure 7 gives an overview of the commissioning steps.

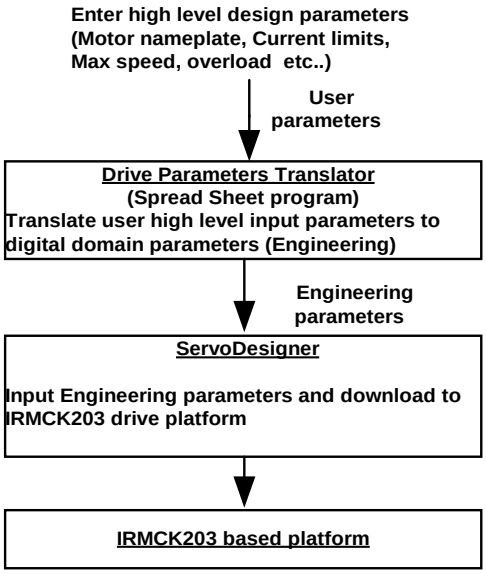


Figure 7. Overview of Drive Commissioning

3.1.1 Drive Parameter Setup

The IRMCK203 support software includes an Excel workbook file that partially automates the procedure of calculating the appropriate values for configuration and tuning parameters. In the workbook, the user enters motor nameplate data and parameters specific to his application, and Excel formulas calculate the appropriate values for certain write registers. In Excel, the "Save As…" function is used to export the register values to a text file, and in ServoDesigner, the text file can be imported to fill in the register values. Then, when the Configure Motor function is executed in ServoDesigner, the values are written to the IRMCK203 based platform.

The Excel Workbook File

The Excel workbook file is named IRMCS2031-DriveParams.xls. Double click the file to open it in Excel.

At the bottom of the workbook window, there are two sheet tabs, which select the worksheet to be displayed. The first tab selects the "User Entries" worksheet used to set up motor and application parameters. This sheet is pre-initialized with values appropriate for the Sanyo Denki 400W 3000rpm motor and is provided as an example. The "User Entries" worksheet can be customized for any motor. To calculate settings for more than one motor, make copies of the IRMCS2031-DriveParams.xls file and modify each copy to define a different motor.

The second tab is labeled "Parameter Export." This worksheet shows the calculated write register values and is the sheet that needs to be exported for use in ServoDesigner.

Enter Motor and Application Parameters in Excel

The first stage of configuring drive parameters involves entering the correct settings for a specific motor and custom application requirements.

Step 1. Initialize a motor setup sheet for the motor.

Click on the "User Entries" sheet tab to select the motor setup worksheet. If desired, double-click the sheet tab and change the tab title to identify the motor. The first line of the motor setup worksheet describes the motor. Double click on column B and enter a description of the motor. (The description is optional; it's not used in the calculations and is not exported to ServoDesigner.)

Step 2. Enter Motor Information.

The motor information section of the "User Entries" worksheet contains parameter settings that should be available in the motor's datasheet or on its nameplate. To enter a value for each parameter, double click in column B on the same line as the parameter name. When the mouse is moved over column B for each parameter, a short description of the parameter is shown in a help bubble.

	A	B	C	D	E
1	Motor:	Sanyo Denki P30B06040DX3			
2					
3	"========== Motor Information ==========				
4					
5	Hz =	200	Hz		
6	RPM =	3000	rpm		
7	Lq =	0.006	H		
9	R_Stator =	1.6	ohms/ph		
10	Amps =	2.7	Arms		
11	Inertia =	1.76E-03	Kg-m2		
12	Kt	0.525	N-m/Arms		
13	Ke	18.78	V ln-rms/krpm		
14	Poles	8			
15					

Figure 8. Motor Information in User Entries Worksheet

More detailed descriptions are provided below.

Hz — The rated frequency of the motor (in Hertz).

RPM — The rated speed of the motor (in RPM).

Lq — Motor per phase inductance (in Henry).

R_Stator — Per phase resistance of the motor **plus** cable (in ohms).

Amps — The rated current of the motor (in Amps rms).

Inertia — Total inertia (motor inertia plus load in Kg-m^2). If total load inertia is not specified in the available design data, use a best estimate and adjust the value later when fine-tuning drive operation (refer to Section 3.1.2).

Kt — Motor torque constant (in Newton-Meter per Amps rms).

Ke — Motor voltage constant (in line-to-neutral rms volts per thousand rpm). Note that some motor manufacturers provide data in line-to-line rms volts, in which case the value must be converted to line-to-neutral voltage.

Poles — The number of motor poles.

Step 3. Enter Application Information

The application information section of the "User Entries" worksheet contains parameter settings that describe the requirements of a specific application. The parameters are described below. To enter a value for each parameter, double click in column D on the same line as the parameter name.

	A	B	C	D	E
15					
16	"================== Application Information ==============				
17					
18	"------------------------- General --------------------------"				
19	Max RPM			3500	rpm
20	(Vdc_Nom) Nominal Vdc			310	Volts
21	(OvLoad) Max pu motor current at rated speed			1	pu
22					
23	"---------- Speed Regulator Tuning ----------"				
24	Speed Regulator BW			50	rad/sec
25	Acceleration Rate			1	sec to rate speed
26	Deceleration Rate			1	sec to rate speed
27					
28	"--------------------- Current Limits ---------------------"				
29	Motoring Limit			100	%
30	Regen Limit			20	%
31					
32	"---------- Start-Up Current Limits ----------"				
33	Drive Start current limit			100	%
34					
35	"------------------ Minimum Speed ------------------"				
36	Minimum drive running speed			300	rpm
37					
38	"---------- Inverter Switching Frequency ----------"				
39	(fc) Pwm carrier freq			10	KHz
40					
41	"-------- Max anticipated Drive Peak Amps ----------"				
42	"------- (Purpose: for Current Shunt sizing) ---------"				
43	(Max_Amps)	Drive peak amps		13.00	Apk
44					
45	"--------------------- Stopping Mode ---------------------"				
46	Stop Mode			1	0 - Coast Stop
47					1 - Ramp Stop

Figure 9. Application Information in User Entries Worksheet

Max RPM — This is the maximum speed (in rpm) required for the application. When motor speed exceeds this value, the system will generate an Overspeed trip fault. It is suggested that this value be set to the rated speed of the motor plus 20 percent.

Nominal Vdc	Nominal DC bus voltage (in volts). For use with the IRMCS2031 development platform, the nominal dc bus voltage should be set to 1.414 * ac input voltage (ac input voltage: USA 110V, JAP 100V, UK 220V etc.).
Max overload current	This is the anticipated maximum current in per unit drawn by the motor at the motor's rated speed. Setting this parameter to 1 pu means that the system drives 100% rated current at the rated speed.
Speed Regulator BW	Speed regulator bandwidth (in rad/sec). The system may not tolerate high speed regulator bandwidth (due to mechanical coupling, gear box etc.), resulting in load mechanical resonance. If the correct setting for this parameter is not known, start with a value of 10 rad/sec and raise it gradually as the system is tuned. Typical values would range between 10 and 25 rad/sec.
Acceleration Rate	This parameter defines the number of seconds required for the motor to accelerate from 0 speed to the motor's rated speed.
Deceleration Rate	This parameter defines the number of seconds required for the motor to decelerate from rated speed to 0 speed.
Motoring Limit	Positive torque current limit (in percentage of rate current). Motoring power is energy transferred from the inverter to the motor while the motor is running.
Regen Limit	Negative torque current limit (in percentage of rate current). Regenerative energy is transferred from the motor to the inverter when the motor decelerates. If the system does not contain a breaking resistor to absorb the regenerative energy, an increase in DC bus voltage (and potential trip fault) results. This parameter should be set to zero if the system cannot absorb regenerative power, which is the case for the IRMCS2031 development platform as shipped.
Drive start current limit	Drive start-up current. During initial drive start-up, this current limit will be applied to ensure robust start-up. Input as percentage of rated motor current.
Minimum running speed	This is the minimum allowable operating speed for the Sensorless drive. Typical values range between 5% and 10% of rated motor speed.
Pwm carrier freq	PWM carrier frequency. 10 KHz is the default setting for the IRMCS2031 product. The setting of this parameter is a tradeoff between current ripple, inverter loss and EMI noise.
Drive peak amps	This parameter defines the anticipated maximum drive current. This parameter should be chosen to accommodate the anticipated full current range. The current feedback resolution will degrade as a consequence of using a higher drive peak amps value. Therefore, it is best to choose the minimum value that satisfies the requirements of the application. It may be necessary to change the current feedback shunt resistor on the IRMCS2031 development platform to conform to the setting of this parameter. A shunt value calculated and displayed on the worksheet to the right of the Drive peak amps entry (column F) shows the recommended resistor value. It may be necessary to adjust the setting of the Drive peak amps parameter slightly to obtain a shunt recommendation that corresponds to a commercially available resistor value (1% or less tolerance recommended).

Stopping Mode

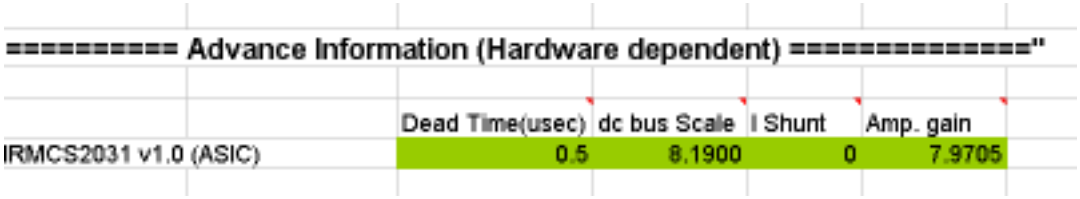
The drive stopping mode can be configured using this parameter.

Coast Stop (enter 0) : when stop command is issued, the inverter will switch off immediately. The motor speed will be decreased by windage and friction.

Ramp Stop (enter 1): When stop command is issued, the inverter will control the motor speed down to zero. The rate of stopping is determined by the setting of deceleration rate and Regen current limit.
Note: Ramp stop will regenerate energy back to the dc bus, hence will increase dc bus voltage during fast deceleration, please ensure brake is installed if Ramp stop mode is used.

Step 4. Enter Advance Information (Hardware Dependent)

The advanced information section of the motor setup worksheet contains parameter settings that are specific to the hardware platform. **It is not necessary to modify these settings for use with the IRMCS2031 development platform.**



========== Advance Information (Hardware dependent) ==========="

	Dead Time(usec)	dc bus Scale	I Shunt	Amp. gain
IRMCS2031 v1.0 (ASIC)	0.5	8.1900	0	7.9705

Deadtime

This parameter sets the inverter dead time delay. The enter unit is in usec. The setting depends on what type (IGBT, MOSFET etc..) of main power switches being used in the inverter. Users should refer to the Deadtime value suggested by the power device manufacturer.

Dc Bus Scale

This is the dc bus scaling in digital counts per volt of dc bus. The information is the hardware scaling of dc bus feedback. For instance: if user's hardware scales down the dc bus 100 times, then at 500 V dc bus level, 5V will appear at the A/D converter (ADS7818) input. The ADS7818 maps 5V (at Vcc = 5) to 4095 digital counts. Therefore, the Bus scaling is 4095/500 = 8.19 cts/V.

I Shunt

This parameter provides current feedback selection. Please enter "0" if using IR2175 , "1" if using Inverter Leg Shunt

Amp Gain

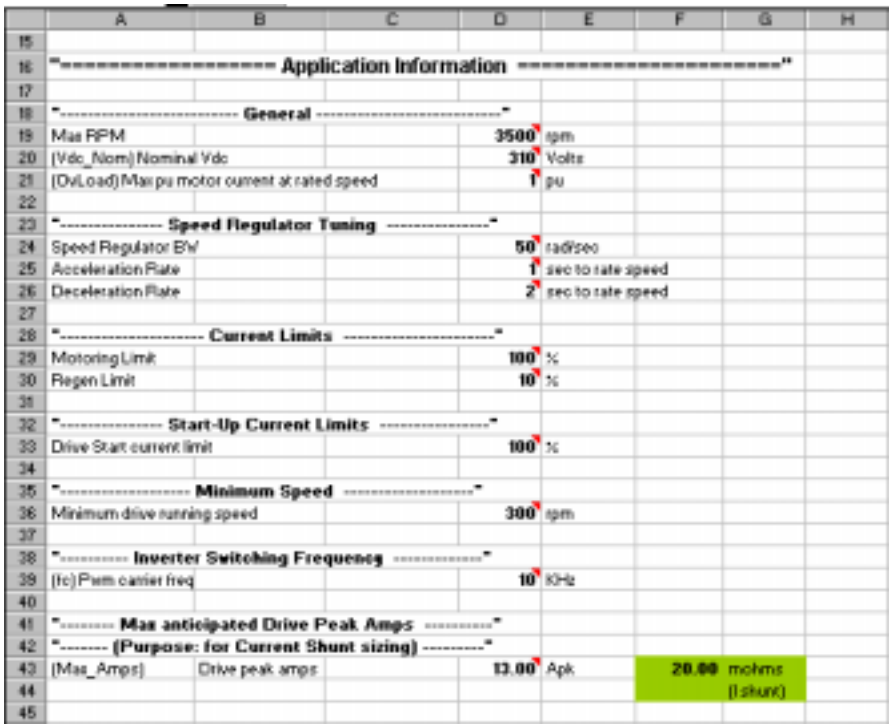
This parameter is only required if Leg Shunt is selected as the current feedback choice, and specifies the amplifier gain from Leg shunt sampling resistor to input of A/D converter. For instance, in the hardware example given in Section 2.4.2, the Leg shunt amplifier gain is 7.97.

Export Drive Parameters in Excel

In the second stage of configuring drive parameters, the parameter settings selected in the previous section are used to calculate values for a number of IRMCK203 write registers. The write register values are written to a text file in a specific format defined for use with ServoDesigner.

Step 1. Note Shunt Resistor Value

At the bottom of the "User Entries" worksheet, note the calculated current feedback shunt resistor value shown in column F (see Figure 10). If the value shown does not correspond to an available resistor, it may be necessary to modify the "Drive peak amps" setting. After modifying the value, check the shunt resistor value again.



	A	B	C	D	E	F	G	H
15								
16	"---------- Application Information ----------"							
17								
18	"---------- General ----------"							
19	Max RPM			3500	rpm			
20	(Vdc_Nom) Nominal Vdc			310	Volts			
21	(OvLoad) Max pu motor current at rated speed			1	pu			
22								
23	"---------- Speed Regulator Tuning ----------"							
24	Speed Regulator BW			50	rad/sec			
25	Acceleration Rate			1	sec to rate speed			
26	Deceleration Rate			2	sec to rate speed			
27								
28	"---------- Current Limits ----------"							
29	Motoring Limit			100	%			
30	Regen Limit			10	%			
31								
32	"---------- Start-Up Current Limits ----------"							
33	Drive Start current limit			100	%			
34								
35	"---------- Minimum Speed ----------"							
36	Minimum drive running speed			300	rpm			
37								
38	"---------- Inverter Switching Frequency ----------"							
39	(fc) Pwm carrier freq			10	KHz			
40								
41	"-------- Max anticipated Drive Peak Amps ---------"							
42	"------- (Purpose: for Current Shunt sizing) ---------"							
43	(Max_Amps)	Drive peak amps		13.00	Apk	20.00	mohms (I shunt)	
44								
45								

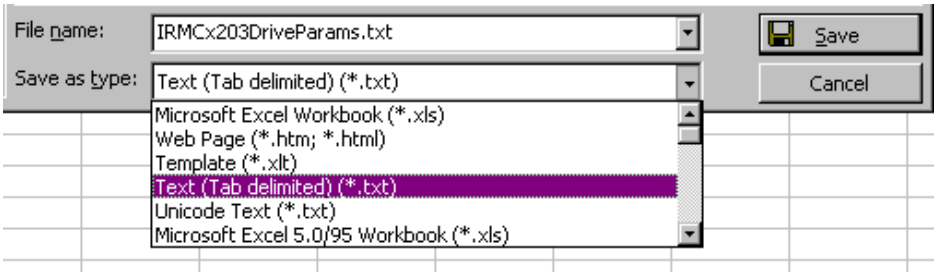
Figure 10. Shunt Resistor Value in User Entries Worksheet

Step 2. Save the Settings

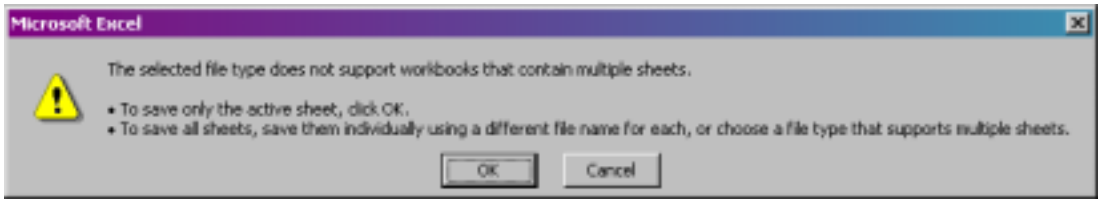
When all parameters are set appropriately, select Save from Excel's File menu to save the workbook file in ".xls" format.

Step 3. Export Drive Parameters

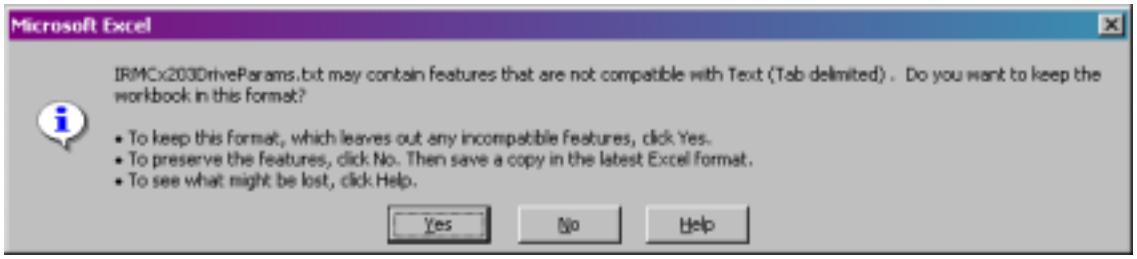
Click on the Parameter Export tab at the bottom of the workbook window. This worksheet shows the register values that were calculated settings were changed in the motor setup worksheet. From Excel's File menu, select "Save As…". In the Save As dialog, select Save as type: "Text (Tab delimited) (*.txt)" as shown below. Then browse to the folder where the exported drive parameters file is to be saved, specify a file name, and click Save.



Click OK when the following warning message appears:



Click Yes when this warning message appears:



Import Drive Parameters in ServoDesigner

The final stage of drive parameter configuration involves loading the drive parameter settings into a ServoDesigner database and writing the registers to the IRMCK203. For information about how to use ServoDesigner, refer to the ServoDesigner User's Guide. In particular, Section 10.3 of that document describes the Import Drive Parameters feature.

The text file exported from the Excel workbook contains two sections: Parameters and Registers.

The Parameters Section

The Parameters section specifies motor configuration parameters, which are saved in the ServoDesigner configuration file (.irc file). In ServoDesigner, the settings can be viewed and modified by selecting Motor Configuration from the Preferences menu. When the drive parameters text file is imported into ServoDesigner, the motor configuration parameters in the import text file always replace the current settings in the ServoDesigner database.

The Registers Section

Each of the entries in the Register section of the file identifies a write register and a value to be stored in the register. In a ServoDesigner database, there are several locations where each register value can be used:

- In the register definition, the Value to Write is written to the corresponding IRMCK203 register when the register entry is double clicked.
- Also in the register definition, the EEPROM Value to Write can be saved to EEPROM and used to initialize the IRMCK203 register on power up.
- In the Function Definitions section, one or more functions may write the register value to the IRMCK203. (A function is set up to perform a sequence of operations automatically.)

When the drive parameters text file is imported into ServoDesigner, there are several options for updating any or all of these register settings with the value specified in the file.

Step 1. Run ServoDesigner and Open a Database

Start ServoDesigner and select Open from the File menu. ServoDesigner configuration files have the file extension ".irc". Browse to locate a ServoDesigner configuration file and open it. (To create a custom configuration file to use with a specific project, it's best to make a copy of the example file included with the release.)

Step 2. Import Drive Parameters
From the File menu, select Import, and from the Import sub-menu, select Drive Parameters. Browse to locate the text file that was exported from Excel and click Open to open it. In the Import Drive Parameters dialog, select one of the three available modes and click OK. Depending on the selected mode, ServoDesigner may prompt for confirmation before modifying each register setting or group of settings. Refer to the ServoDesigner User's Guide for more information about the available modes of operation.

Step 3. Save the New Settings
The Import Drive Parameters function in ServoDesigner updates register values in the database that's currently open. To save the new settings in the configuration file, select Save from the File menu before exiting ServoDesigner. If this is not done, the updates will be lost, and the Import Drive Parameters function will need to be repeated next time the configuration file is opened.

Step 4. Write the Settings to the IRMCK203
The Import Drive Parameters function does not write any values to the IRMCK203; it simply updates the register settings in the database. To transfer the register settings to the IRMCK203, it is necessary to either double click each write register individually (not recommended) or execute a function that writes the registers automatically. The Configure Motor function is pre-defined for this purpose. To execute the Configure Motor function, click the Configure Motor icon on the toolbar, or double click Configure Motor in the Function Definitions section of the tree view.

3.1.2 Evaluating Drive Performance
The drive parameter translation as described in the previous section is the first step of drive commissioning. It is expected that the user parameter entries such as motor nameplate information and load inertia will have at least 10% error. This is typical due to the inaccuracy in motor datasheet and load information. The drive performance can be further refined by going through drive diagnostics as described in Section 3.1.3.

For motor control purposes, the rotor angle information is required to optimally control a Permanent Magnet AC motor. In the IRMCK203, the control is performed without a shaft encoder (Sensorless). The rotor angle is estimated utilizing motor phase (V, W) current and DC bus voltage feedback information.

In the IRMCK203 drive controller, there are 3 control modes (Figure 11) for estimation of the rotor angle for the entire speed range including zero speed. During motor start-up phase, the controller will go through these three control modes in sequence. These control modes are described below.

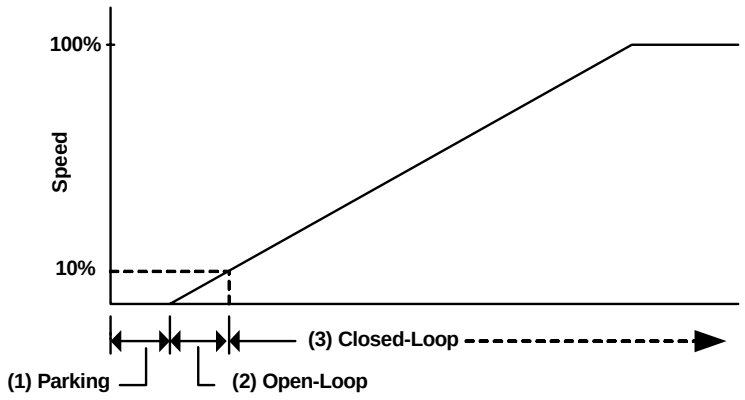


Figure 11. Drive Control Modes

1) Parking – The initial rotor angle is identified by forcing DC current into the motor and hence forcing the motor shaft to park at a certain prescribed angle.

2) Open-loop – Immediately after Parking stage, the rotor angle is estimated in an open-loop fashion, which utilizes a simple motor-load mechanical model to estimate the rotor angle (estimate load characteristics). If the mismatch between the external load characteristics and the internal motor-load model is exceedingly large, start-up performance will suffer.

3) Closed-loop – motor speed increases during start-up; the motor voltage also builds up due to the increase in speed. Useful information for rotor angle estimation can be then be extracted from the motor voltage (estimated by using motor current and DC bus voltage). The drive will enter Closed-loop control mode as shown in Figure 11.

3.1.3 Diagnostic Mode Functions

Diagnostic mode functions are provided in the ServoDesigner tool to fine tune drive performance. It is recommended to go through diagnostic mode in the proper order (Parking Diagnostic then Start-up Diagnostic).

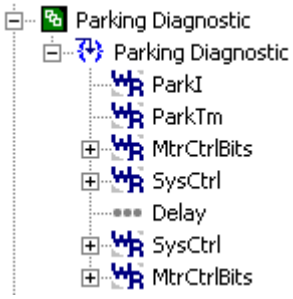


Figure 12. Parking Diagnostic Function

Parking Diagnostic

With the Parking diagnostic (shown in Figure 12), the optimal Parking current (ParkI) and the Parking time duration (ParkTm) can be readily determined. In addition, current controller and current feedback can also be verified.

When the Parking Diagnostic function is executed, the drive is forced to stay in Parking state for five seconds, followed by a stop. The diagnostic can be stopped anytime by executing the Stop Motor function. The characteristic of parking depends on the amount of dc current injection.

It is possible to verify current control by observing the actual current flowing through the motor windings using current sensing instrumentation (current probe). The amount of parking current injected to the motor is controlled by parameter ParkI as shown in Figure 12. The full scale of ParkI is 255 digital counts, which represents 86.7% rated motor current (in peak Amps). During motor parking, dc current is injected (by inverter) into W-phase and V-phase of the motor; the current in U-phase is regulated to zero. For instance, if motor rated current is 2.7Arms, a value of 77 digital counts in ParkI will produce 1 Amp dc current in W-phase (- 1 Amp) and V-phase (+ 1Amp).

In practice, it is much more than sufficient to park a motor with rated motor current. If an exceedingly large value of ParkI is used, the motor shaft will hunt during parking. This will increase the time for the rotor shaft to settle and hence increase parking time. Systems with a higher inertia to friction ratio will tend to hunt more. Therefore, it is recommended to start with a lower value (say 4% ParkI = 12).

The user can experience the effect of using different ParkI values. It may be noticed that the parking characteristics will also depend on the initial rotor angle (when drive is off). Therefore, the shaft should be rotated (manually, while the drive is off) to a different position before each parking evaluation. It is recommend to use the highest possible value (not to cause excessive motor hunting) of ParkI such that the duration of parking can be minimized. Once the optimal parking current (ParkI) is determined, please note the time required for the motor shaft to settle

during parking. This time duration is the optimal parking time and should be converted to digital drive units and entered into register ParkTm. The full scale of ParkTm is 255, which represents 4 seconds.

After the parking diagnostic has been accomplished, please enter these two parameters (ParkI and ParkTm) into the Sensorless Ctrl Start-up subfunction inside the Configure Motor function, as shown in Figure 13.

Note: To resume normal mode operation (out of diagnostic), the Configure Motor function must be executed again.

Figure 13. Enter Optimal Parking Parameters

Start-up Diagnostic

This diagnostic mode is provided to fine tune open-loop start-up performance. During open-loop start-up, the IRMCK203 control IC estimates the rotor angle based on a simple motor load model, which uses only one configurable parameter (KTorque). The user parameter translator (Excel spreadsheet) also generates this parameter based on user input load inertia. Use this value as a starting point for fine tuning. The goal of this start-up diagnostic is to fine tune this parameter (KTorque) for optimal open-loop drive control performance. If a correct value of KTorque is used the drive will produce the highest torque per ampere ratio during open-loop start-up. The drive may fail to start if excessive error is present in KTorque gain.

When the Start-up Diagnostic function is executed, the drive will enter parking mode followed by open-loop start-up. The drive will coast to a stop as soon as open-loop is accomplished (determined when motor frequency exceeds the level prescribed by write register WeThr). If an optimal value of KTorque is used, the drive will accelerate the motor to a higher speed since maximum torque per ampere is achieved.

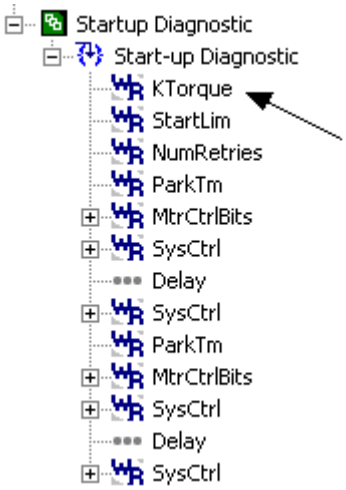


Figure 14. Start-up Diagnostic Function

Users can observe motor shaft movement during the Start-up diagnostic to determine an optimal value of KTorque. As mentioned earlier, when the Start-up diagnostic is initiated, the drive will enter parking mode for 4 seconds; thereafter open-loop start up will be initiated. There will be shaft movement due to parking of the motor during the initial 4 seconds. It is important to observe the shaft movement only in the open-loop startup period. An optimal KTorque value will generate higher starting torque and hence increased motor shaft rotation during open-loop duration.

If measurement instrumentation (oscilloscope and voltage probe) is available, it can be used to observe the motor back EMF to determine the optimal KTorque value. The motor back emf is proportional to motor speed. By observing the motor line-line voltage at the end of the open-loop period (indicated by a momentary high pulse on D/A converter channel 4), it is possible to determine an optimum value of KTorque.

Figure 15 and Figure 16 illustrate two example runs of the start-up diagnostic with two different values of KTorque being used. As can be seen in these figures, after open-loop terminates, the speed of the motor coasts down. It is apparent that the KTorque value used in Figure 16 provides higher voltage and frequency; hence the motor speed is also higher.

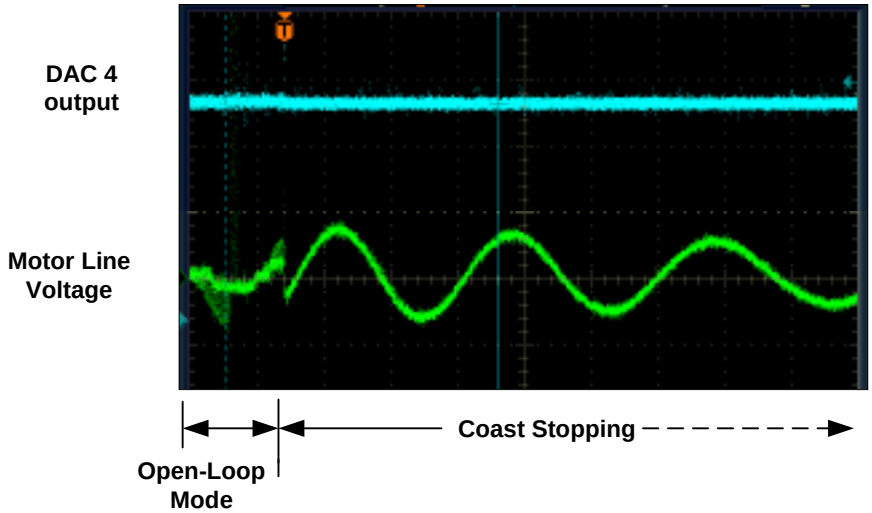


Figure 15. Open-loop start (KTorque = 400)

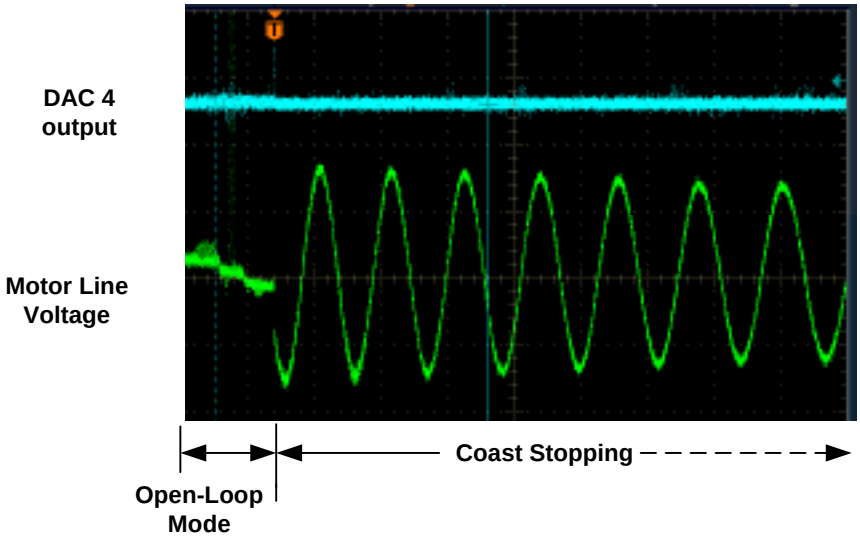


Figure 16. Open-loop start-up (KTorque = 520)

After the optimal value for KTorque has been determined, please enter the value into the Sensorless Ctrl Open-loop subfunction inside the Configure Motor function as shown in Figure 17.

Resume Normal Operation

After completing the diagnostic tests described in this section, the Configure Motor function must be executed in order to resume normal drive mode.

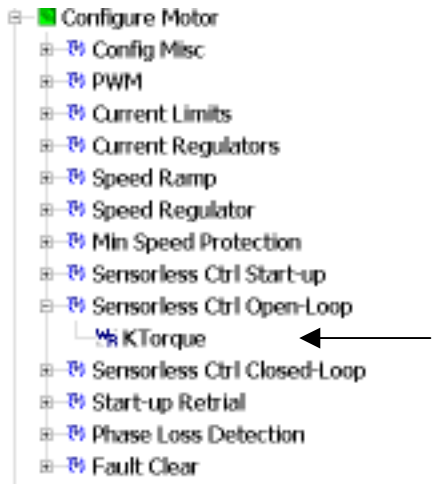


Figure 17. Enter Optimal KTorque Parameter

3.1.4 Miscellaneous Functions

Miscellaneous functions provided in the ServoDesigner tool are described in this section.

Phase Loss Detection

This function provides detection (during start-up) of a loose wire (u, v, w) between drive and motor.

During motor parking (first stage of motor startup), a certain amount of dc current is injected into the motor windings for the purpose of initialization of rotor position. If motor feedback currents do not match the expected dc injection current level, a phase loss fault (PhsLossFlt) is triggered. This fault can be disabled via bit 4 (PhsLosFltDisable field) of the MtrCtrlBits write register.

Start-up Retrial

This function provides start-up retrials upon a start-up failure. Start-up failure may occur if the motor shaft is jammed or motor starting torque cannot overcome shaft friction during startup.

Motor starting torque can be controlled by motor starting current limit (StartLim) as shown in Figure 18. The scaling of StartLim is 4095 = rated motor current.

The write registers written on execution of the Startup Retrial function are described below. The function is shown in
Figure 19.

NumRetries – This parameter determines the number of start-up retries. A value of zero will disable startup retry. The maximum number of retries is 15.

FlxThrL - The low flux threshold level for determining a successful startup (scaling: 129 = 100% flux). Please do not modify this parameter without consulting a motor drive FAE.

FlxThrH - The upper flux threshold level for determining a successful startup (scaling: 64 = 100% flux). Please do not modify this parameter without consulting a motor drive FAE.

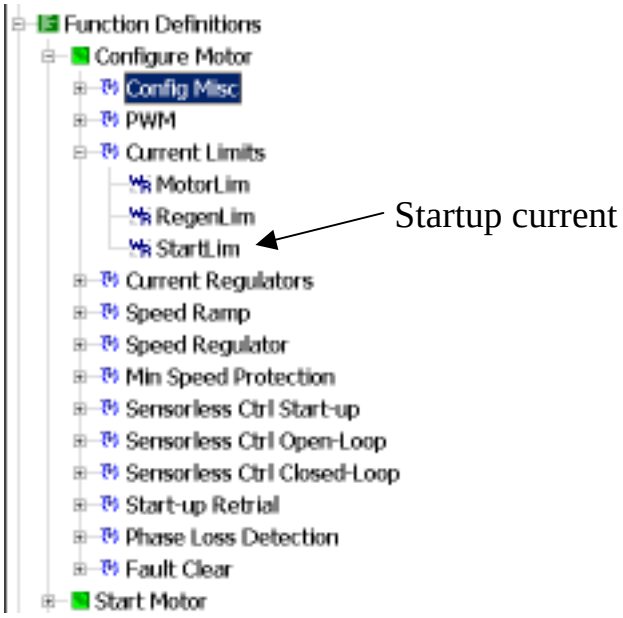


Figure 18. Configuring the Startup Current

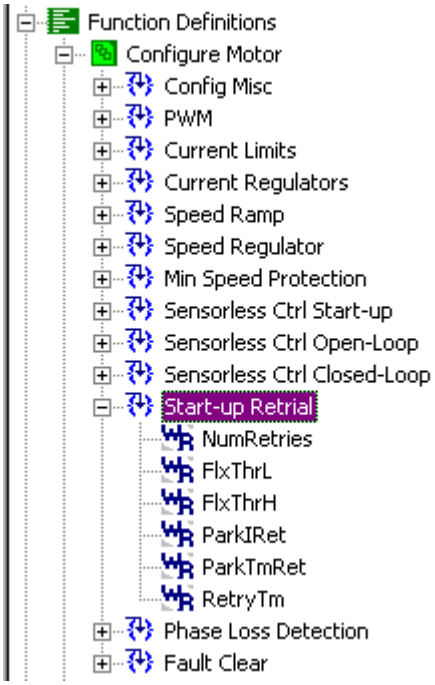


Figure 19. Start-up Retrial Function

ParkIRet - During motor start-up, dc current is injected to the motor for maximization of startup torque per ampere rating. Users are able to use a higher level of dc current injection (ParkIRet scaling 255 = Motor Rated Amp * 0.866) after two or more restarts. This is done to increase the chance of a successful start-up.

ParkTmRet - During motor start-up, dc current is injected to the motor for maximization of startup torque per ampere rating. ParkTm controls the duration of dc current injection. However, users are able to use a longer duration after two or more restarts by setting this parameter (ParkTmRet scaling 255 = 4 secs.). This is done to increase the chance of a successful start-up.

RetryTm - This parameter provides the adjustment to the sampling instant for determination of start failure. The sampling instant starts when Closed_Loop = 1. Scaling 1 count = 1.966 msec. Please do not modify this parameter without consulting a motor drive FAE.

3.2 Standalone Operation and Register Initialization via Serial EEPROM

This section describes the register-controlled configuration and operation for an example system that uses the IRMCK203 system in standalone mode, which requires no initialization by a host microprocessor. In standalone mode, the IRMCK203 initializes the host write registers from an I^2C serial EEPROM at power up and receives control commands from the external hardware user interface signals during motor operation. The system described in this example is shown in Figure 20.

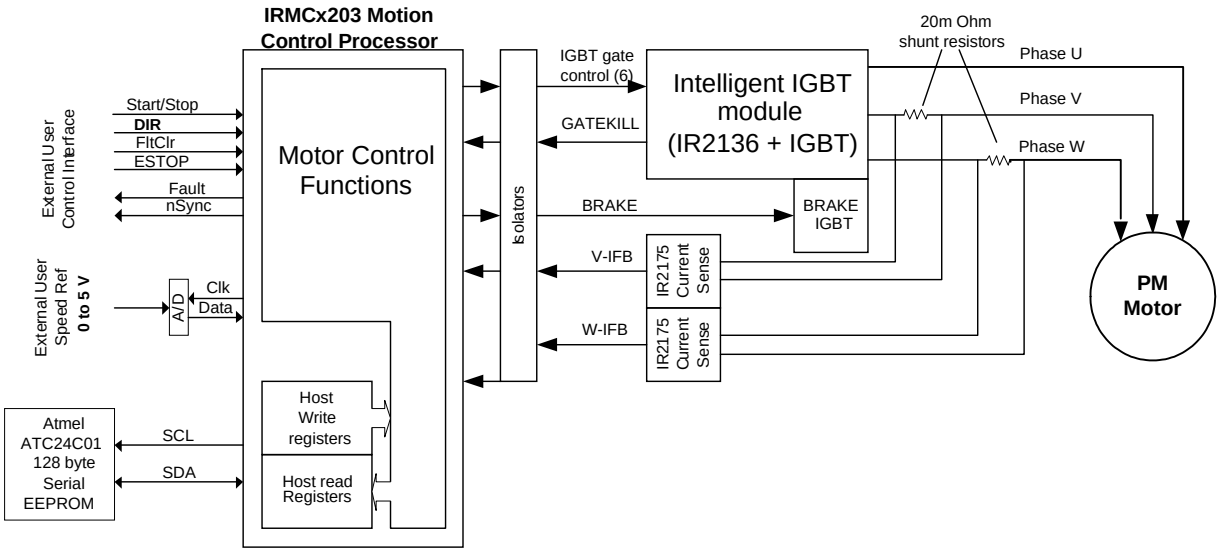


Figure 20. IRMCK203 Standalone System

3.2.1 Register Initialization via EEPROM

Each time the IRMCK203 powers up, it checks for valid EEPROM data by reading a single byte from EEPROM address 0x5D, which represents the IRMCK203 register map version code. If this value matches the IRMCK203 internal register map version, the IRMCK203 EEPROM initialization sequencer reads 128 sequential bytes from the EEPROM and stores them in host write registers 0 - 0x7F.

If the user sets the location in the EEPROM that corresponds to the SystemConfig register group's ExtCtrl field to "1", motor operation can be controlled directly using the external user interface immediately after power-on host write register initialization.

3.2.2 Starting and Stopping the Motor

To start or stop the motor in standalone mode, the user simply drives the Start/Stop signal. Section 2.4.1 describes the required I/Os for Standalone mode operation.

3.2.3 Fault Processing

When the IRMCK203 detects a fault condition, it disables PWM and asserts the "Fault" signal. In standalone mode, the user clears the fault condition using the "FltClr" signal. Fault processing is otherwise identical to that described Section 2.6

4 Reference

4.1 Register Access

A host computer controls the IRMCK203 using its slave-mode Full-Duplex SPI port, a standard RS-232 port or a 8-bit parallel port for connection to a microprocessor. All interfaces are always active and can be used interchangeably, although not simultaneously. Control/status registers are mapped into a 128-byte address space.

4.1.1 Host Parallel Access

The IRMCK203 contains an address register that is updated with the Host Register address when HP_A = 1. After each subsequent data byte is either read or written, the internal address register is incremented. The diagram below shows that Data Bytes 0 to N would access register locations initially specified by the Address Byte. The Address Bye with the HP_A signal can be asserted at any time.

Address Byte HP_A = 1	Data Byte 0 HP_A = 0	……. HP_A = 0	Data Byte N HP_A = 0

Host Parallel Data Transfer Format

4.1.2 SPI Register Access

When configured as an SPI device read only and read/write operations are performed using the following transfer format:

Command Byte	Data Byte 0	…….	Data Byte N

Data Transfer Format

Bit Position							
7	6	5	4	3	2	1	0
Read Only	Register Map Starting Address						

Command Byte Format

Data transfers begin at the address specified in the command byte and proceed sequentially until the SPI transfer completes. As in the Host Parallel Access, the internal address register is incremented after each SPI byte is transferred. Note that accesses are read/write unless the "read only" bit is set.

4.1.3 RS-232 Register Access

The IRMCK203 includes an RS-232 interface channel that provides a direct connection to the host PC. The software interface combines a basic "register map" control method with a simple communication protocol to accommodate potential communication errors.

RS-232 Register Write Access

A Register write operation consists of a command/address byte, byte count, register data and checksum. When the IRMCK203 receives the register data, it validates the checksum, writes the register data, and transmits and acknowledgement to the host.

Command / Address Byte	Byte Count	1-6 bytes of register data	Checksum

Register Write Operation

Command Acknowledgement Byte	Checksum

Register Write Acknowledgement

Bit Position							
7	6	5	4	3	2	1	0
1=Read/ 0=Write	Register Map Starting Address						

Command/Address Byte Format

Bit Position							
7	6	5	4	3	2	1	0
1=Error/ 0=OK	Register Map Starting Address						

Command Acknowledgement Byte Format

The following example shows a command sequence sent from the host to the IRMCK203 requesting a two-byte register write operation:

0x2F	Write operation beginning at offset 0x2F
0x02	Byte count of register data is 2
0x00	Data byte 1
0x04	Data byte 2
0x35	Checksum (sum of preceding bytes, overflow discarded)

A good reply from the IRMCK203 would appear as follows:

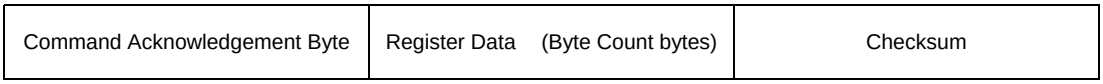
0x2F	Write completed OK at offset 0x2F
0x2F	Checksum

An error reply to the command would have the following format:

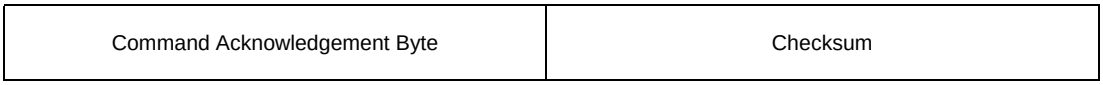
0xAF	Write at offset 0x2F completed in error
0xAF	Checksum

RS-232 Register Read Access
A register read operation consists of a command/address byte, byte count and checksum. When the IRMCK203 receives the command, it validates the checksum and transmits the register data to the host.

Register Read Operation



Register Read Acknowledgement (transfer OK)



Register Read Acknowledgement (error)

The following example shows a command sequence sent from the host to the IRMCK203 requesting four bytes of read register data:

0xA0	Read operation beginning at offset 0x20 (high-order bit selects read operation)
0x04	Requested data byte count is 4
0xA4	Checksum

A good reply from the IRMCK203 might appear as follows:

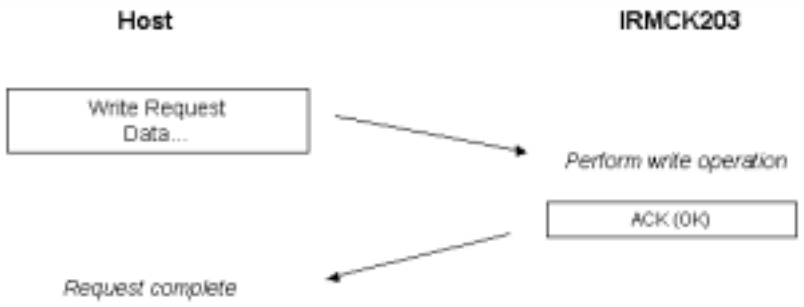
0x20	Read completed OK at offset 0x20
0x11	Data byte 1
0x22	Data byte 2
0x33	Data byte 3
0x44	Data byte 4
0xCA	Checksum

An error reply to the command would have the following format:

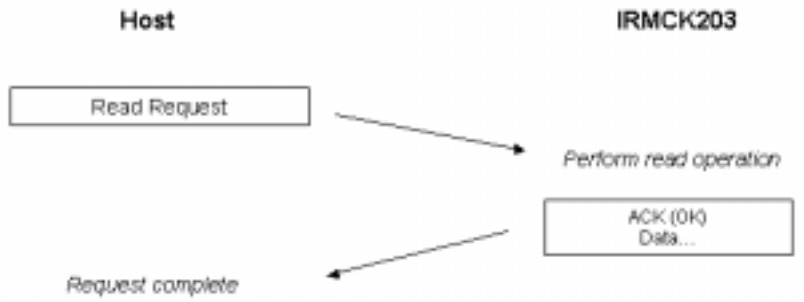
0xA0	Read at offset 0x20 completed in error
0xA0	Checksum

RS-232 Timeout
The IRMCK203 receiver includes a timer that automatically terminates transfers from the host to the IRMCK203 after a period of 32 msec.

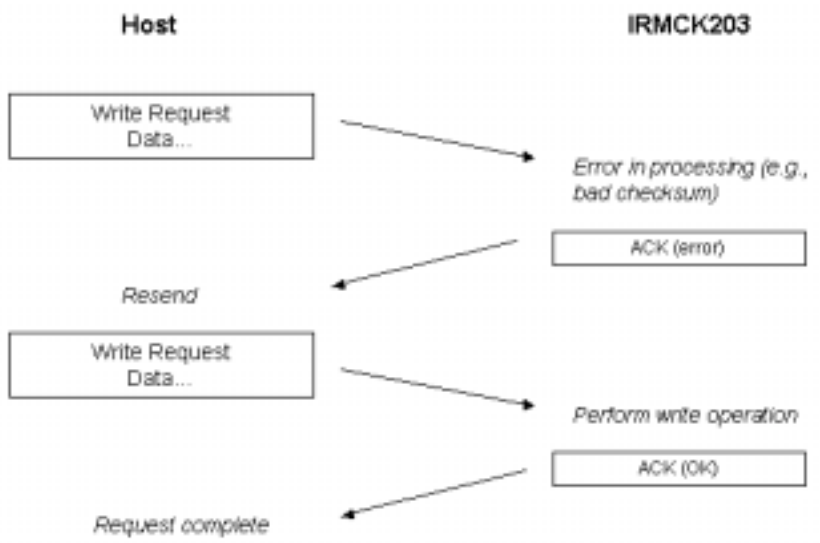
RS-232 Transfer Examples
The following example shows a normal exchange executing a register write access.



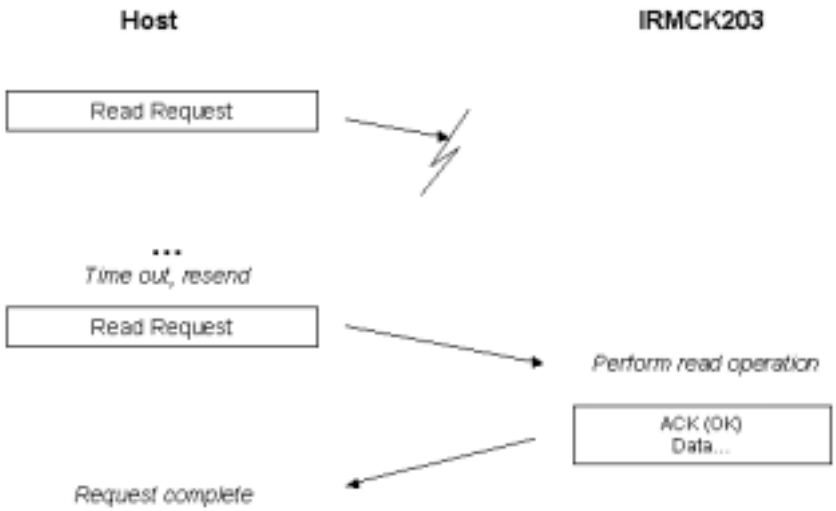
The example below shows a normal register read access exchange.



The following example shows a register write request that is repeated by the host due to a negative acknowledgement from the IRMCK203.



In the final example, the host repeats a register read access request when it receives no response to its first attempt.



4.2 Write Register Definitions

4.2.1 PwmConfig Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0xC	Gatekill Sns (W)	SPARE	Gate SnsL (W)	Gate SnsU (W)	SyncSns	BrakeSns	SD (W)	SPARE
0xD	PwmPeriod (LSBs) (W)							
0xE	TwoPhs Pwm (W)	TwoPhs Type (W)	PwmConfig (W)		PwmPeriod (MSBs) (W)			
0xF	PwmDeadTm (W)							
0x44	ModScl (LSBs) (W)							
0x45	ModScl (MSBs) (W)							
0x51	PwmGuardBand (W)							

PwmConfig Write Register Map

Field Name	Access (R/W)	Field Description
SD	W	Shutdown control output to IR2137.
BrakeSns	W	Logic Sense for BRAKE signal output to gate driver IC. 0 = Active low, 1 = active high.
SyncSns	W	Logic Sense for PWM SYNC signal output to microprocessor. 0 = Active low, 1 = active high.
GateSnsU	W	Upper IGBT gate sense. 1 = active high gate control, 0 = active low gate control.
GateSnsL	W	Lower IGBT gate sense. 1 = active high gate control, 0 = active low gate control.
GatekillSns	W	GATEKILL signal sense. 1 = active high GATEKILL, 0 = active low GATEKILL.
PwmPeriod	W	PWM Carrier period. Actual PWM carrier period is 2 * (PwmPeriod + 1) * (System Clock Period).
PwmConfig	W	PWM Configuration. 0 = Asymmetrical center aligned PWM, 1 = Symmetrical Center aligned PWM.
TwoPhsType	W	Used only for two-phase PWM modulation mode: 0 = Type 1 2-phase PWM 1 = Type 2 2-phase PWM
TwoPhsPwm	W	Selects PWM modulation mode: 0 = Enable 3-phase space vector PWM modulation 1 = Enable 2-phase space vector PWM modulation

Field Name	Access (R/W)	Field Description
PwmDeadTm	W	Gate drive dead time in units of system clock cycles (e.g., 30 ns with 33 MHz clock).
ModScl	W	Space vector modulator scale factor. This register, which depends on the PWM carrier frequency, should be set as follows: ModScl = PwmPeriod * sqrt(3) * 4096 / 2355 where PwmPeriod is the value in the PwmConfig write register group's PwmPeriod register.
PwmGuardBand	W	This parameter provides a guard band (scaling: 1 = 30nsec) such that PWM switching will not migrate into the current feedback sampling instant (Sync Pulse region). This guard band is provided to improve feedback noise. The parameter only applies to the 3-phase Space Vector modulation scheme. Please do not modify this parameter without consulting a motor drive FAE.

PwmConfig Write Register Field Definitions

4.2.2 CurrentFeedbackConfig Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x15	IfbkScl (LSB) (W)							
0x16	IfbkScl (MSB) (W)							
0x7D	OffsetCalDelay (W)							

CurrentFeedbackConfig Write Register Map

Field Name	Access (R/W)	Field Description
IfbkScl	W	Rotating frame Iq component and Id component current feedback scale factor. Constant used to scale current measurements before they are used in the field orientation calculation. This is a 15-bit fixed-point signed number with 10 fractional bits that ranges from –16 to + 16 + 1023 / 1024.
IfbOffsVOffsetCal Delay	W	This parameter specifies the delay time (1 = 1 sec) to restart current offset measurement after a stop command is issued. Only applies if Leg Shunt current feedback is selected.12-bit signed value for V phase current feedback offset. When the IfbOffsEnb bit in the SystemControl write register group is "0" this value is automatically added to each current measurement in hardware.

CurrentFeedbackConfig Write Register Field Definitions

4.2.3 SystemControl Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x17	SPARE					HostEstop	StartCmd	Rotation

SystemControl Write Register Map

Field Name	Access (R/W)	Field Description
Rotation	W	Direction of motor rotation: 0 = Reverse motor rotation; 1 = Forward motor rotation.
StartCmd	W	Start/Stop bit. Setting this bit to 1 issues a start command. Setting this bit to 0 stops the motor.
HostEstop	W	Emergency coast stop will take place when this bit is set to one.

SystemControl Write Register Field Definitions

4.2.4 TorqueLoopConfig Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x1A	KpIreg – Current Loop Proportional Gain (LSBs) (W)							
0x1B	KpIreg – Current Loop Proportional Gain (MSBs) (W)							
0x1C	KxIreg – Current Loop Integral Gain (LSBs) (W)							
0x1D	KxIreg – Current Loop Integral Gain (MSBs) (W)							
0x22	VqLim – Quadrature Current Output Limit (LSBs) (W)							
0x23	VqLim – Quadrature Current Output Limit (MSBs) (W)							
0x26	VdLim – Direct Current Output Limit (LSBs) (W)							
0x27	VdLim – Direct Current Output Limit (MSBs) (W)							

TorqueLoopConfig Write Register Map

Field Name	Access (R/W)	Field Description
KpIreg	W	15-bit signed current loop PI controller proportional gain. Scaled with 14 fractional bits for an effective range of 0 – 1.
KxIreg	W	15-bit signed current loop PI controller integral gain. Scaled with 19 fractional bits for an effective range of 0 - .03125.
VqLim	W	16-bit Quadrature current PI controller voltage output limit.
VdLim	W	16-bit Direct current PI controller voltage output limit.

TorqueLoopConfig Write Register Field Definitions

4.2.5 VelocityControl Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x32	KpSreg – Velocity loop proportional gain (LSBs) (W)							
0x33	KpSreg – Velocity loop proportional gain (MSBs) (W)							
0x34	KxSreg – Velocity loop integral gain (LSBs) (W)							
0x35	KxSreg – Velocity loop integral gain (MSBs) (W)							
0x36	MotorLim – Velocity loop Output Positive Limit (LSBs) (W)							
0x37	MotorLim – Velocity loop Output Positive Limit (MSBs) (W)							
0x38	RegenLim – – Velocity loop Output Negative Limit (LSBs)							
0x39	RegenLim – – Velocity loop Output Negative Limit (MSBs)							
0x3A	SpdScl – Speed Scale Factor (LSBs)							
0x3B	SpdScl – Speed Scale Factor (MSBs)							
0x3C	TargetSpd – Setpoint/target speed (LSBs)							
0x3D	TargetSpd – Setpoint/target speed (MSBs)							
0x3E	AccelRate							
0x3F	DecelRate							
0x7A	MinSpd							

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x18	StartLim (LSBs)							
0x19	StartLim (MSBs)							

VelocityControl Write Register Map

Field Name	Access (R/W)	Field Description
KpSreg	W	15-bit velocity loop proportional gain, in fixed point with 5 fractional bits. Range = 0 - 512.
KxSreg	W	15-bit velocity loop integral gain, in fixed point with 13 fractional bits. Range = 0 - 2.
MotorLim	W	Motoring torque current limit (4095 = rated motor current).16-bit speed PI controller output positive limit.
RegenLim	W	Regeneration torque current limit (4095 = rated motor current)16-bit speed PI controller output negative limit (2's complement)..
SpdScl	W	Motor Speed Scale factor. Spd value (in the VelocityStatus read register group) is maintained in SPEED units of SpdScl * (Encoder counts / Velocity Loop Execution) or SpdScl * (RATE * Encoder counts / PWM period). The user should set SpdScl = (64 * 16384) * 60 * PWMFREQ / (RATE * Max RPM * Encoder counts/revolution), which will result in a Spd value ranging ±16384 corresponding to ± Max RPM.
TargetSpd	W	Velocity loop speed setpoint in SPEED units, which are determined by the user via the SpdScl register setting.
AccelRate	W	Positive speedAcceleration rate limit.
DecelRate	W	Negative speedDeceleration rate limit.
MinSpd	W	Minimum speed protection. This parameter sets the minimum reference speed.
StartLim	W	Drive start-up current limit. (4095 = rated motor current).

VelocityControl Write Register Field Definitions

4.2.6 FaultControl Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x42	SPARE						FltClr	DcBusM Enb

FaultControl Write Register Map

Field Name	Access (R/W)	Field Description
DcBusMEnb	W	DC Bus monitor enable. 1 = Monitor DC bus voltage and generate appropriate brake signal control and disable PWM output when voltage fault conditions occur. GatekillFlt and OvrSpdFlt faults cannot be disabled. DC bus voltage thresholds are as follows: Overvoltage – 410V Brake On – 380V Brake Off – 360V Nominal – 310V Undervoltage off – 140V Undervoltage – 120V
FltClr	W	This bit clears all active fault conditions. The user should monitor the FaultStatus read register group to determine fault status and set this bit to "1" to clear any faults that have occurred. A fault condition automatically clears the PwmEnbW and FocEnbW bits in the SystemControl write register group. Note that this bit also directly controls the output 2137 FLTCLR pin. After clearing a fault, the user must explicitly set this bit to "0" to re-enable fault processing.

FaultControl Write Register Field Definitions

4.2.7 SystemConfig Register Group (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x50	ExtCtrl	AdcIfbEnb	Ramp Stop	SPARE				

SystemConfig Write Register Map

Field Name	Access (R/W)	Field Description
RampStop	W	Selects the stopping mode: 0 - Configure for Coast stopping 1 - Configure for Ramp stopping
AdcIfbEnb	W	Selects the current feedback mode: 0 - Selects IR2175 current feedback 1 - Selects Leg-Shunt current feedback
ExtCtrl	W	Setting this bit to "1" enables direct control of basic motor operation via the external User Interface pins. When this bit is "1", the FocEnbW and PwmEnbW bits in the SystemControl write register group are ignored.

SystemConfig Write Register Field Definitions

4.2.8 EepromControl Registers (Write Registers)

At power up, the write registers can be optionally initialized with values stored in EEPROM. The EepromControl write register group and EepromStatus read register group are used to read and write these EEPROM values. Since the EeAddrW write register (which selects the EEPROM offset to read or write) does not require initialization at power up, the location corresponding to that register in EEPROM (at offset 0x5D) is used to store a register map version code. At power on, the IRMCK203 initializes the write registers from EEPROM only if the version code stored at this offset in EEPROM matches its internal register map version code (which can be read from the RegMapVer field of the EepromStatus read register group).

To enable write register initialization at power up, write the appropriate register map version code to EEPROM at offset 0x5D. To disable write register initialization at power up, write a zero (or any non-matching version code) to offset 0x5D of the EEPROM.

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x5C	SPARE					EeWrite	EeRead	EeRst
0x5D	EeAddrW / RegMapVersCode (W)							
0x5E	EeDataW (W)							

EepromControl Write Register Map

Field Name	Access (R/W)	Field Description
EeRst	W	Self-clearing EEPROM reset. Writing a "1" to this bit resets the I2C EEPROM interface.
EeRead	W	Self-clearing I2c EEPROM Read. Writing a "1" to this bit initiates an EEPROM read from the byte located at EEPROM address EeAddrW. After setting this bit the user should poll the EeBusy bit in the EepromStatus read register group to determine when the read completes and then read the data from EeDataR in the EepromStatus read register group.
EeWrite	W	Self-clearing EEPROM Write. Writing a "1" to this bit initiates an EEPROM write from the data byte in EeDataW to the EEPROM address EeAddrW .
EeAddrW	W	EEPROM Address Register. Contains the address for the next EEPROM read or write operation.
EeDataW	W	EEPROM Data Register. Contains the data for the next EEPROM write operation.

EepromControl Write Register Field Definitions

4.2.9 ClosedLoopAngleEstimator Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x60	IScl (LSBs) (W)							
0x61	IScl (MSBs (W)							
0x62	FlxBInit (LSBs) (W)							
0x63	FlxBInit (MSBs) (W)							
0x6A	PllKp (LSBs) (W)							
0x6B	SPARE		PllKp (MSBs (W)					
0x6C	PllKi (LSBs) (W)							
0x6D	SPARE		PllKi (MSBs (W)					
0x6E	VoltScl (LSBs) (W)							
0x6F	VoltScl (MSBs (W)							
0x70	Rs (LSBs) (W)							
0x71	Rs (MSBs (W)							
0x72	Ld (LSBs) (W)							
0x73	Ld (MSBs (W)							
0x74	AtanTau (LSBs) (W)							
0x75	AtanTau (MSBs (W)							
0x76	FlxTau (LSBs) (W)							
0x77	SPARE			FlxTau (MSBs) (W)				

ClosedLoopAngleEstimator Write Register Map

Field Name	Access (R/W)	Field Description
IScl	W	Current scaler for motor flux calculation.
FlxBInit	W	Initialization value of Beta flux at start.
PllKp	W	Flux phase lock loop proportional gain.
PllKi	W	Flux phase lock loop integral gain.
VoltScl	W	Voltage scaler for motor flux calculation.
Rs	W	Motor per phase resistance including cable (@25C).
Ld	W	Motor per phase inductance.
AtanTau	W	Rotor angle estimator phase compensation gain.
FlxTau	W	Rotor angle estimator flux model time constant.

ClosedLoopAngleEstimator Write Register Field Definitions

4.2.10 OpenLoopAngleEstimator Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x66	KTorque (LSBs) (W)							
0x67	KTorque (MSBs (W)							
0x5F	VFGain (W)							

OpenLoopAngleEstimator Write Register Map

Field Name	Access (R/W)	Field Description
KTorque	W	Motor mechanical model torque constant.
VFGain	W	Open-Loop Volts/Hz Flux gain. (for diagnostic use only).

OpenLoopAngleEstimator Write Register Field Definitions

4.2.11 StartupAngleEstimator Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x64	ParkI (W)							
0x65	SPARE	Zero SpdFlt Disable	Use2xFrq Scale	PhsLosFlt Disable	DiagnosticCtrl (W)			
0x68	WeThr (LSBs) (W)							

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x69	WeThr (MSBs (W)							
0x78	ParkTm (W)							

StartupAngleEstimator Write Register Map

Field Name	Access (R/W)	Field Description
ParkI	W	DC current injection level during motor parking (start-up mode).
DiagnosticCtrl	W	1 (0001) – Enable Parking diagnostic 2 (0010) – Enable start-up diagnostic 5 (0101) – Enable current regulator diagnostic 9 (1001) – Enable volts Hertz diagnostic
PhsLosFlt Disable	W	Enable/disable phase loss fault: 0 = Enable Phase Loss Fault; 1 = Disable Phase Loss Fault
Use2xFrqScale	W	Selects speed scaling: 0 - Norminal speed scale 1 - Reduce speed feedback scaling by half Please do not modify this parameter without consulting motor control FAEs
ZeroSpdFlt Disable	W	Zero speed fault enable/disable: 0 - Enbale Zero Speed Fault 1 - Disable Zero Speed Fault
WeThr	W	Frequency threshold level (switch over from open-loop to closed-loop mode).
ParkTm	W	Time duration of parking mode. 255 = 4 sec

StartupAngleEstimator Write Register Field Definitions

4.2.12 StartupRetrial Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x1E	RetryTm (LSBs)							
0x1F	RetryTm (MSBs)							
0x79	ParkTmRet							
0x7B	FlxThrL							
0x7C	FlxThrH							

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x7E	NumRetries							
0x7F	ParkIRet							

StartupRetrial Write Register Map

Field Name	Access (R/W)	Field Description
RetryTm	W	This parameter provides the adjustment to the sampling instant for determination of start failure. The sampling instant starts when Closed_Loop = 1. Scaling 1 count = 1.966 msec. Please do not modify this parameter without consulting a motor drive FAE.
ParkTmRet	W	During motor start-up, dc current is injected to the motor for maximization of startup torque per ampere rating. ParkTm controls the duration of dc current injection. However, users are able to use a longer duration after two or more restarts by setting this parameter (ParkTmRet scaling 255 = 4 secs.). This is done to increase the chance of a successful start-up.Start-up failure may be caused by increased shaft friction. After first start-up retry, the parking time can be increased to improve parking performance.
FlxThrL	W	The low flux threshold level for determining a successful startup (scaling: 129 = 100% flux). Please do not modify this parameter without consulting a motor drive FAE.The low flux threshold level for determining a successful startup.
FlxThrH	W	The upper flux threshold level for determining a successful startup (scaling: 64 = 100% flux). Please do not modify this parameter without consulting a motor drive FAE.The high flux threshold level for determining start-up failure.
NumRetries	W	If start-up fails, the user can program start-up retrial. This parameter determines the number of start-up retries. A value of zero will disable startup retrial. The maximum number of retries is 15.
ParkIRet	W	During motor start-up, dc current is injected to the motor for maximization of startup torque per ampere rating. Users are able to use a higher level of dc current injection (ParkIRet scaling 255 = Motor Rated Amp * 0.866) after two or more restarts. This is done to increase the chance of a successful start-up.Start-up failure may be caused by increased shaft friction. After first start-up retry, the parking current can be increased to improve parking performance.

StartupRetrial Write Register Field Definitions

4.2.13 PhaseLossDetect Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x79	ParkTmRet							
0x28	AdjPark1							
0x29	AdjPark2							
0x2A	RetryTm							

PhaseLossDetect Write Register Map

Field Name	Access (R/W)	Field Description
AdjPark1	W	Anticipated W-phase motor current gain scaler used during initial stage of Phase Loss detection.
AdjPark2	W	Anticipated W-phase motor current gain scaler used during final stage of Phase Loss detection.
PhsLosThr	W	Phase Loss detection current error thershold.

PhaseLossDetect Write Register Field Definitions

4.2.14 D/AConverter Registers (Write Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x4F	DacSel							

D/AConverter Write Register Map

Field Name	Access (R/W)	Field Description
DacSel	W	Selects D/A converter diagnostic outputs 0 - 3. A value of 0 selects: Data 0 = Alpha fluxFlux Data 1 = Electrical Rotor angle Data 2 = Alpha voltageTorque current Data 3 = Closed loop/open loop status (0 = open, 1 = closed) A value of 1 selects: Data 0 = Alpha currentDC bus voltage Data 1 = Torque current feedbackAlpha voltage Data 2 = IQ refTorque current reference Data 3 = Motor speed A value of 2 selects: Data 0 = Q-axis command voltage Data 1 = D-axis command voltage Data 2 = Alpha current Data 3 = Beta current A value of 3 selects: Data 0 = Flux magnitude Data 1 = Current error at parking Data 2 = Parking diagnostic flag Data 3 = W-phase current

D/AConverter Write Register Field Definitions

4.2.15 Factory Test Register (Write Register)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x58	Test							

Factory Write Register Map

Field Name	Access (R/W)	Field Description
Test	W	Reserved for factory use. Data written to this register could be read from a read register at location 0x58.

Factory Write Register Field Definitions

4.3 Read Register Definitions

4.3.1 SystemStatus Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x7	StartStop	FwdRev	ESTOP	PwrID		ExtCtrlR	Foc EnbR	Pwm EnbR

SystemStatus Read Register Map

Field Name	Access (R/W)	Field Description
PwmEnbR	R	PWM Enable bit status.
FocEnbR	R	FOC Enable bit status.
ExtCtrlR	R	Reflects the status of the ExtCtrl bit in the System Configuration write register (address 0x50).
PwrID	R	Power ID. 0 = 3 kW, 1 = 2 kW, 2 = 500 W.
ESTOP	R	User Interface emergency stop signal (1 – emergency stop)
FwdRev	R	User Interface "FWD/REVDIR" digital input status. 1 - Forward rotation request 0 - Reverse rotation request
StartStop	R	User Interface "START/STOP" digital input status. 1 - Start 0 - Stop

SystemStatus Read Register Field Definitions

4.3.2 DcBusVoltage Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0xA	DcBusVolts (LSBs)							
0xB	SPARE			Brake	DcBusVolts (MSBs)			

DcBusVoltage Read Register Map

Field Name	Access (R/W)	Field Description
DcBusVolts	R	DC Bus Voltage. Data range is 0 - 4095, which corresponds to a DC bus voltage between 0 and 500 volts.
Brake	R	Brake signal status. 1 = Brake signal active.

DcBusVoltage Read Register Field Definitions

4.3.3 FocDiagnosticData Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0xC	RotatorAngle (LSB)							
0xD	SPARE	Parking Done	Start_ Fail	Closed_ Loop	RotatorAngle (MSB)			
0xE	Id – Synchronous Frame Direct Current (LSBs)							
0xF	Id – Synchronous Frame Direct Current (MSBs)							
0x10	Iq – Synchronous Frame Quadrature Current (LSBs)							
0x11	Iq – Synchronous Frame Quadrature Current (MSBs)							
0x12	IqRef_C – Synchronous Frame Quadrature Current command (LSB)							
0x13	IqRef_C – Synchronous Frame Quadrature Current command (MSB)							
0x14	Flx_Alpha – Estimated Motor Flux (LSB)							
0x15	Flx_Alpha – Estimated Motor Flux (MSB)							
0x16	I_Alpha – Alpha Frame Current (LSB)							
0x17	I_Alpha – Alpha Frame Current (MSB)							
0x18	V_Alpha – Alpha Frame Voltage (LSB)							
0x19	V_Alpha – Alpha Frame Voltage (MSB)							

FocDiagnosticData Read Register Map

Field Name	Access (R/W)	Field Description
RotatorAnlge	R	Estimated rotor angle (electrical), which is used for synchronous frame to stationary frame transformation. The scaling is 4096 = 2PI. The range is 0 – 4095.
Closed_Loop	R	This is a drive control status flag which indicates that the drive has switched from open-loop to closed-loop operation. The switch over is done during drive start-up (initial speed ramping)
Start_Fail	R	This is a drive control status flag indicating that the drive has failed to start due to various reasons (for instance: shaft jam). The start-stop sequencer uses this bit and parameter NumRetry to determine whether a start-up retry should be activated.
Parking Done	R	This is a status flag indicating that the drive has finished obtaining the initial rotor angle (parking) for motor startup. During drive start-up, the first start-up stage is parking stage.
Id, Iq	R	Synchronous or rotating frame direct and quadrature current values in 2's complement representation. The full scale current values range from –16384 to 16383. (Scaling: 4095 = rated motor current)
IqRef_C	R	Synchronous or rotating frame quadrature current command values in 2's complement representation. The full scale current values range from –16384 to 16383.
Flx_Alpha	R	Estimated motor flux value. Scaling is 5000 = rated motor flux.
I_Alpha	R	Stationary frame current. Scaling is platform dependent (current shunt resistor). Drive commissioning tool (Spreadsheet) provides the scaling of I_Alpha (AiBi scale).
V_Alpha	R	Stationary frame Alpha voltage. This voltage is constructed by dc bus voltage and modulation index in the Stationary frame. The scaling is platform dependent.

FocDiagnosticData Read Register Field Definitions

4.3.4 FaultStatus Register Group (Read Registers)

The Fault Status register records fault conditions that occur during drive operation. When any of these fault conditions occur, the PWM output is automatically disabled. The user should monitor this register continuously for fault conditions. A fault condition can be cleared by writing a "1" to the FaultClr bit in the FaultControl write register group. (This does not automatically re-enable PWM output.)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x1E	PhsLoss Flt	RetryFlt	ZeroSpd Flt	ExecTm Flt	OvrSpdFlt	OvFlt	LvFlt	GatekillFlt

FaultStatus Read Register Map

Field Name	Access (R/W)	Field Description
GatekillFlt	R	Filtered and latched version of IR2137 FAULT output.
LvFlt	R	DC bus low voltage fault. This fault occurs if the DC bus drops below 120V.
OvFlt	R	DC bus overvoltage fault. This fault occurs if the DC bus voltage exceeds 410V.
OvrSpdFlt	R	Over speed fault. This fault occurs whenever the motor reaches the positive or negative limits. The user should use the scale factor in the SpdScl field of the VelocityControl write register group to scale the motor speed so that it falls between -16384 and +16383 with these limits as the over speed condition.
ExecTmFlt	R	Execution time fault.
ZeroSpdFlt	R	Zero Speed fault. When speed is less than MinSpd/2 (half minimum speed) for a continuous period of 2 4 seconds, the zero speed fault will be set.
RetryFlt	R	Start-up retry fault. After a certain number (determined by parameter NumRetries) of start-up failures, this fault will be set.
PhsLossFlt	R	Phase loss fault. Drive to motor phase connection may be loose.

FaultStatus Read Register Field Definitions

4.3.5 VelocityStatus Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x26	Spd (LSBs)							
0x27	Spd (MSBs)							

VelocityStatus Read Register Map

Field Name	Access (R/W)	Field Description
Spd	R	Current motor speed in SPEED units. (See the description of SpdScl in the VelocityControl write register group.)

VelocityStatus Read Register Field Definitions

4.3.6 CurrentFeedbackOffset Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x30	IfbVOffs (LSBs) (R)							
0x31	IfbWOffs (LSBs) (R)				IfbVOffs (MSBs) (R)			
0x32	IfbWOffs (MSBs) (R)							

CurrentFeedbackOffset Read Register Map

Field Name	Access (R/W)	Field Description
IfbVOffs, IfbWOffs	R	Current feedback offset values from the last IFB Offset calculation. These values are automatically applied to each current feedback measurement value whenever the IfbOffsEnb bit in the SystemControl write register group is set.

CurrentFeedbackOffset Read Register Field Definitions

4.3.7 EepromStatus Registers (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x38	SPARE							EeBusy
0x39	EdDataR (R)							
0x3A	EeAddrR (R)							

EepromStatus Read Register Map

Field Name	Access (R/W)	Field Description
EeBusy	R	I2C EEPROM Interface busy bit. The user should wait for this bit to clear before initiating EEPROM read or write operations.
EeDataR	R	EEPROM Data Register. Contains the data from the last EEPROM read operation. Note that writing to the EeRst field in the EepromControl write register group invalidates this register.

Field Name	Access (R/W)	Field Description
EeAddrR	R	EEPROM Address read register shows the value stored in EEPROM at the offset of the EeAddrW write register (0x5D). Since this address in the EEPROM contains the BPIRMCK203 register map version, the user can read this field to determine whether or not the write registers were initialized at power on.

EepromStatus Read Register Field Definitions

4.3.8 FOCDiagnosticDataSupplement Register Group (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x3C	ElecAngR (LSBs) (R)							
0x3D	SPARE				ElecAngR (MSBs) (R)			
0x3E	SpdRef (LSBs) (R)							
0x3F	SpdRef (MSBs) (R)							
0x40	SpdErr (LSBs) (R)							
0x41	SpdErr (MSBs) (R)							
0x42	IqRefR (LSBs) (R)							
0x43	IqRefR (MSBs) (R)							

FOCDiagnosticDataSupplement Read Register Map

Field Name	Access (R/W)	Field Description
ElecAngR	R	Electrical angle.
SpdRef	R	Speed PI controller reference input.
SpdErr	R	Speed PI controller error.
IqRefR	R	Speed PI controller output.

FOCDiagnosticDataSupplement Read Register Field Definitions

4.3.9 ProductIdentification Registers (Read Registers)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x7C	ProductID (R)							
0x7D	RegMapVerID (R)							
0x7E	RevCodeID (LSBs) (R)							
0x7F	RevCodeID (MSBs) (R)							

ProductIdentification Read Register Map

Field Name	Access (R/W)	Field Description
ProductID	R	Product identification code.
RegMapVerID	R	Current register map version code.
RevCodeID	R	IRMCK203 Revision Code. Revision code format is "XX.XX", where each "X" is a 4-bit hexadecimal number.

ProductIdentification Read Register Field Definitions

4.3.10 Factory Register (Read Register)

Byte Offset	Bit Position							
	7	6	5	4	3	2	1	0
0x58	Test (R)							

Factory Read Register Map

Field Name	Access (R/W)	Field Description
Test	R	Data value resulting from a write to write register 0x58. Used for factory use only.

Factory Read Register Field Definitions

Appendix A Space Vector PWM Module

The Space Vector PWM generation module accepts modulation index commands and generates the appropriate gate drive waveforms for each PWM cycle. This section describes the operation and configuration of the SVPWM module.

SVPWM Basic Theory and Transfer Characteristics

A three-phase 2-level inverter with dc link configuration can have eight possible switching states, which generates output voltage of the inverter. Each inverter switching state generates a voltage Space Vector (V1 to V6 active vectors, V7 and V8 zero voltage vectors) in the Space Vector plane (Figure 21). The magnitude of each active vector (V1 to V6) is 2/3 Vdc (dc bus voltage).

The Space Vector PWM (SVPWM) module inputs modulation index commands (U_Alpha and U_Beta) which are orthogonal signals (Alpha and Beta) as shown in Figure 21. The gain characteristic of the SVPWM module is given in Figure 22. The vertical axis of Figure 22 represents the normalized peak motor phase voltage (V/Vdc) and the horizontal axis represents the normalized modulation index (M).

Where : $M = Umag * Mod\_Scl * 10^{-4}$

$Umag = \sqrt{(U\_Alpha^2 + U\_Beta^2)}$ (-32768 <= U_Alpha, U_Beta <= 32767)

Mod_Scl : Input scaling factor (0 to 32767 range)

The inverter fundamental line-to-line Rms output voltage (Vline) can be approximated (linear range) by the following equation:

$Vline = Umag * Mod\_Scl * Vdc / \sqrt{6} / 2^{25}$ where dc bus voltage (Vdc) is in volts

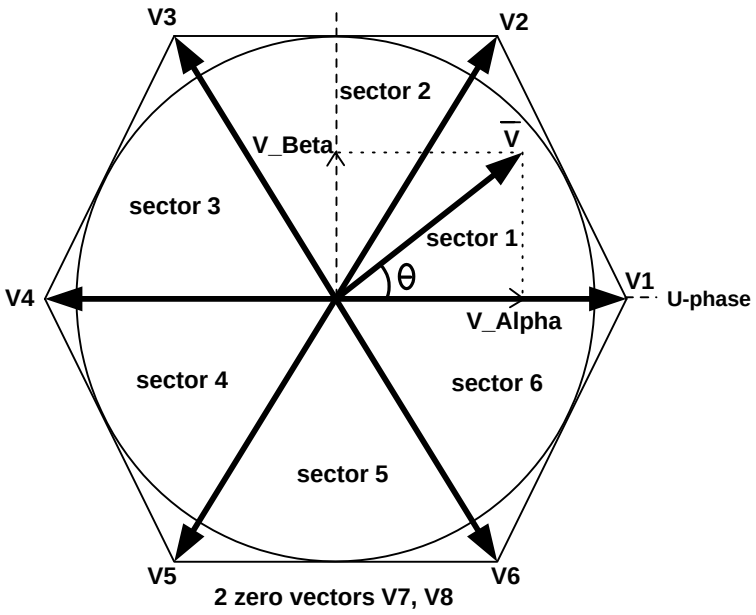


Figure 21. Space Vector Diagram

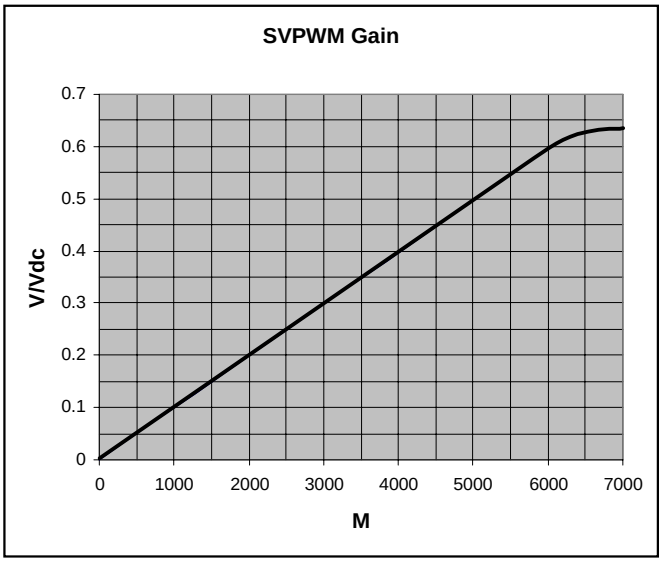


Figure 22. Transfer Characteristics

The maximum achievable modulation (Umag_L) in the linear operating range is given by:

$$Umag\_L = 2^{25} * \sqrt{3} / Mod\_Scl$$

Over modulation occurs when modulation Umag > Umag_L. This corresponds to the condition where the voltage vector in Figure 23 increases beyond the hexagon boundary. Under such circumstance, the Space Vector PWM algorithm will rescale the magnitude of the voltage vector to fit within the Hexagon limit. The magnitude of the voltage vector is restricted within the Hexagon; however, the phase angle (θ) is always preserved. The transfer gain (Figure 22) of the PWM modulator reduces and becomes non-linear in the over modulation region.

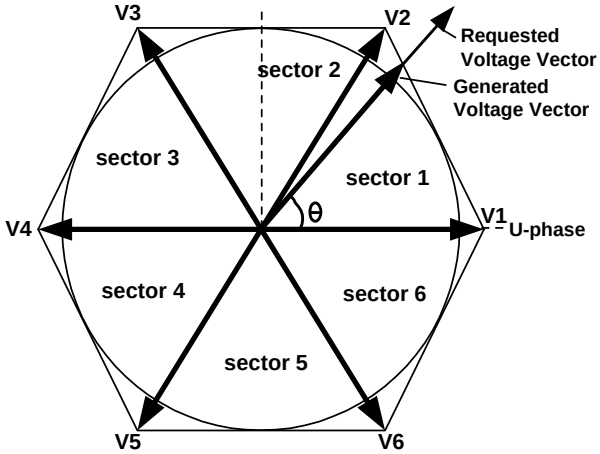


Figure 23. Voltage Vector Rescaling

PWM Operation

Referring to Figure 24, upon receiving the modulation index commands (U_Alpha and U_Beta) the sub-module SVPWM_Tm starts its calculations at the rising edge of the PwmLoad signal. The SVPWM_Tm module implements an algorithm that selects (based on sector determination) the active space vectors (V1 to V6) being used and calculates the appropriate time duration (w.r.t. one PWM cycle) for each active vector. The appropriated zero vectors are also being selected. The SVPWM_Tm module consumes 11 clock cycles typically and 35 clock cycles (worst case Tr) in over modulation cases. At the falling edge of nSYNC, a new set of Space Vector times and vectors are readily available for actual PWM generation (PhaseU, PhaseV, PhaseW) by sub module PwmGeneration. It is crucial to trigger PwmLoad at least 35 clock cycles prior to the falling edge of nSYNC signal; otherwise new modulation commands will not be implemented at the earliest PWM cycle.

Figure 24 (3-phase modulation) and Figure 25 (2-phase modulation) illustrates the PWM waveforms for a voltage vector locates in sector I of the Space Vector plane (Figure 21). The gating pattern outputs (PWMUH … PWMWL) include deadtime insertion (describe in later section).

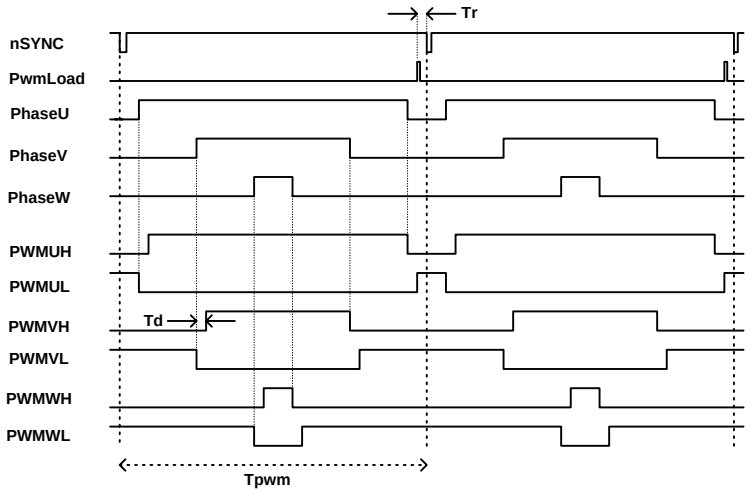


Figure 24. 3-phase Space Vector PWM

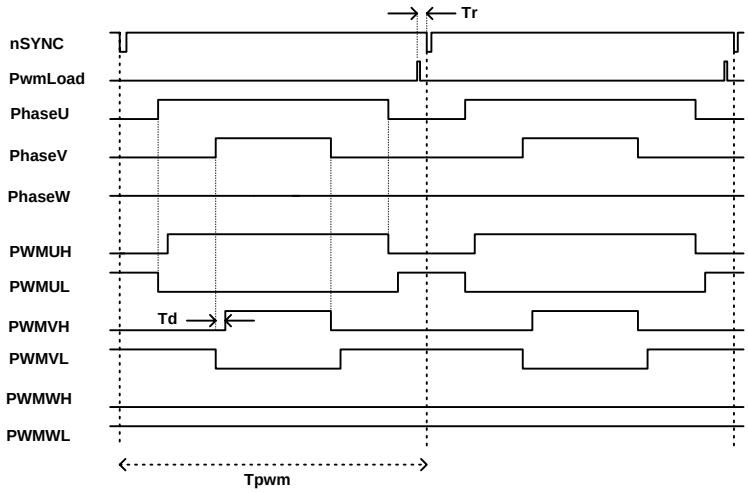


Figure 25. 2-phase (6-step PWM) Space Vector PWM

PWM Carrier Period

Input variable PwmCval controls the duration of a PWM cycle. It should be populated by the system clock frequency (Clk) and Pwm frequency (PwmFreq) selection. The variable should be calculated as:

$$PwmCval = Clk/(2 * PwmFreq) - 1$$

The input resolution of the Space Vector PWM modulator signals U_Alpha and U_Beta is 16-bit signed integer. However, the actual PWM resolution (PwmCval) is limited by the system clock frequency.

Deadtime Insertion Logic

Deadtime is inserted at the output of the PWM Generation Module. The resolution is 1 clock cycle, or 30 nsec at a 33.3 MHz clock and is the same as those of the voltage command registers and the PWM carrier frequency register.

The deadtime insertion logic chops off the high side commanded volt*seconds by the amount of deadtime and adds the same amount of volt*seconds to the low side signal. Thus, it eliminates the complete high side turn on pulse if the commanded volt*seconds is less than the programmed deadtime.

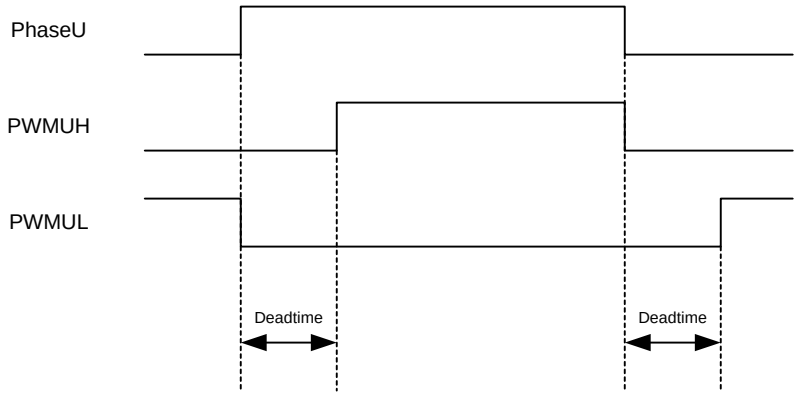


Figure 26. Deadtime Insertion

The deadtime insertion logic inserts the programmed deadtime between two high and low side of the gate signals within a phase. The deadtime register is also double buffered to allow "on the fly" deadtime change and control while PWM logic is inactive.

Symmetrical and Asymmetrical Mode Operation

There are two modes of operation available for PWM waveform generation, namely the Center Aligned Symmetrical PWM (Figure 24) and the Center Aligned Asymmetrical PWM (Figure 27). The volt-sec can be changed every half a PWM cycle (Tpwm) since PwmLoad occurs every half a PWM cycle (compare Figure 24 and Figure 27). With Symmetrical PWM mode, the inverter voltage Config = 0), the inverter voltage can be changed at two times the rate of the switching frequency. This will provide an increase in voltage control bandwidth, however, at the expense of increased current harmonics.

The mode of operation is selected using the PwmConfig field of the PwmConfig write register group (described in Section 4.2.1). To select Center Aligned Asymmetrical PWM, set the PwmConfig field to '0'. To select Center Aligned Symmetrical PWM, set the field to '1'.

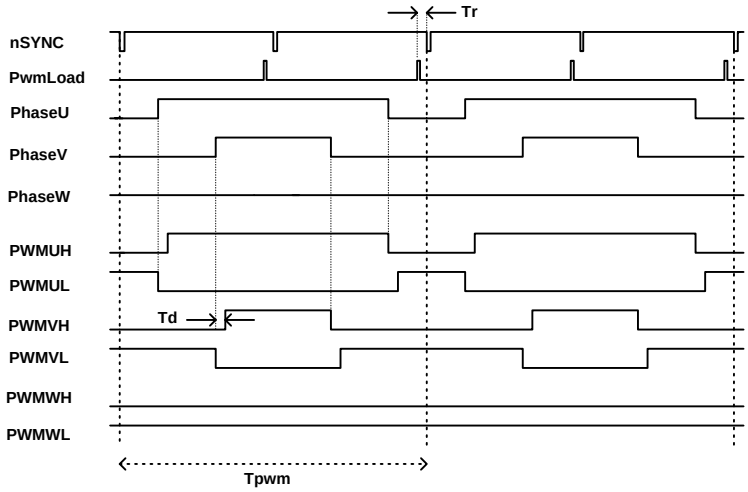


Figure 27. Asymmetrical PWM Mode

Three-Phase and Two-Phase Modulation

Three-phase and two-phase Space Vector PWM modulation options are provided for the IRMCx203. The Volt-sec generated by the two PWM strategies are identical; however with 2-phase modulation the switching losses can be reduced significantly, especially when high switching frequency (>10Khz) is employed. Figure 28 shows the switching pattern for one PWM cycle when the voltage vector is inside sector 1.

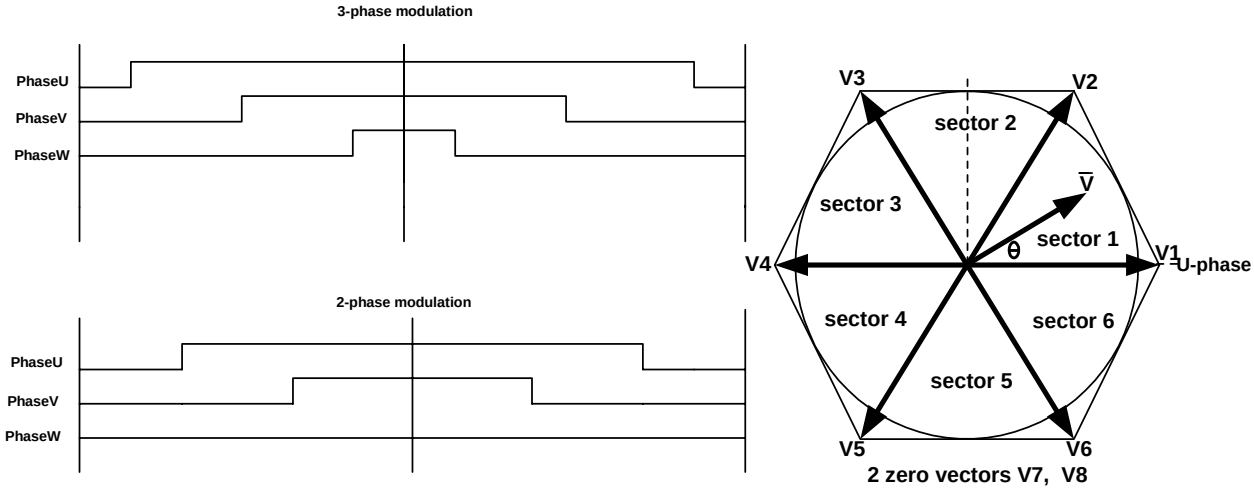
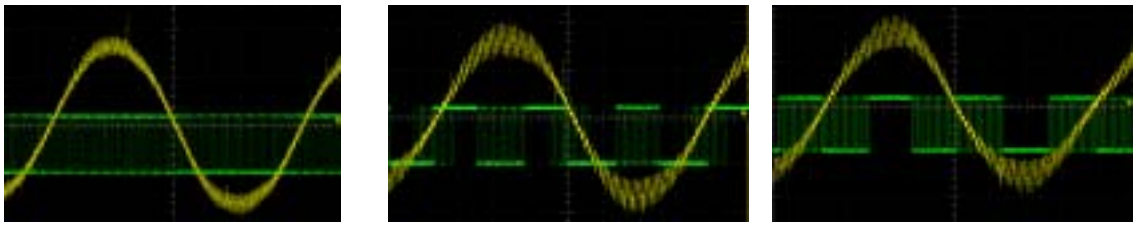


Figure 28. Three-Phase and Two-Phase Modulation

The field TwoPhsPwm of the PwmConfig write register group (described in Section 4.2.1) provides selection of three-phase or two-phase modulation. The default setting is three-phase modulation. Successful operation of two-phase modulation in the entire speed operating range will depend on hardware configuration. If the gate driver employs a bootstrap power supply strategy, misoperation will occur at low motor fundamental frequencies (< 2Hz) under two-phase modulation control.

There are two types of two-phase modulation schemes provided. The field TwoPhsType in the PwmConfig write register group is used to select the type, as described in Section 4.2.1. Figure 29 illustrates the different types of

Space Vector PWM strategies available for the IRMCx203 product. Inverter Pole voltage and motor current are displayed in Figure 29.



(a) Three-phase PWM (b) Two-phase (type 1) PWM (c) Two-phase (type 2) PWM

Figure 29. Different Types of Space Vector PWM

Appendix B IR2175 Current Sensing

Two channels of current feedback interface logic are provided in the IRMCx20x system. Each module measures the incoming varying duty period of the 130 kHz carrier frequency signal at the IR2175output. Measurement is performed for both carrier frequency period and on duty period at the same time using fast counters. Counting frequency is 133 MHz with a 33.3 MHz system clock.

The IR2175 are the unique high voltage ICs capable of measuring the motor phase current through an associated shunt resistor, which can generate ±260mV voltage range. The output of the IR2175 is an open drain with a 130 kHz fixed carrier frequency where the duty variance is linearly proportional to ±260 mV input voltage. The counting frequency is 133.3 MHz when the system clock crystal frequency is 33.3 MHz, which yields 10-bit resolution of the current measurement data from the IR2175.

The offset measurement is automatically added after the 10-bit current measurement has been calculated. The offset value must be calculated and supplied by external hardware or software.

The period measurement of both the carrier frequency period and the duty period of the IR2175 output signal are performed. For carrier frequency period measurement, there is a 16-stage averaging filter to smooth out the 130 kHz carrier period of the IR2175. The multiply/divide computation follows after completing both period measurements. Divide computation between the carrier frequency period and the duty period alleviates temperature drift of the incoming data off the IR2175, since variation of these periods uniformly moves in same direction as temperature changes.

The measured and adjusted data is coherently updated to the host digital system such as a microcontroller, DSP, or FPGA. A block diagram of the current measurement block is shown in Figure 30.

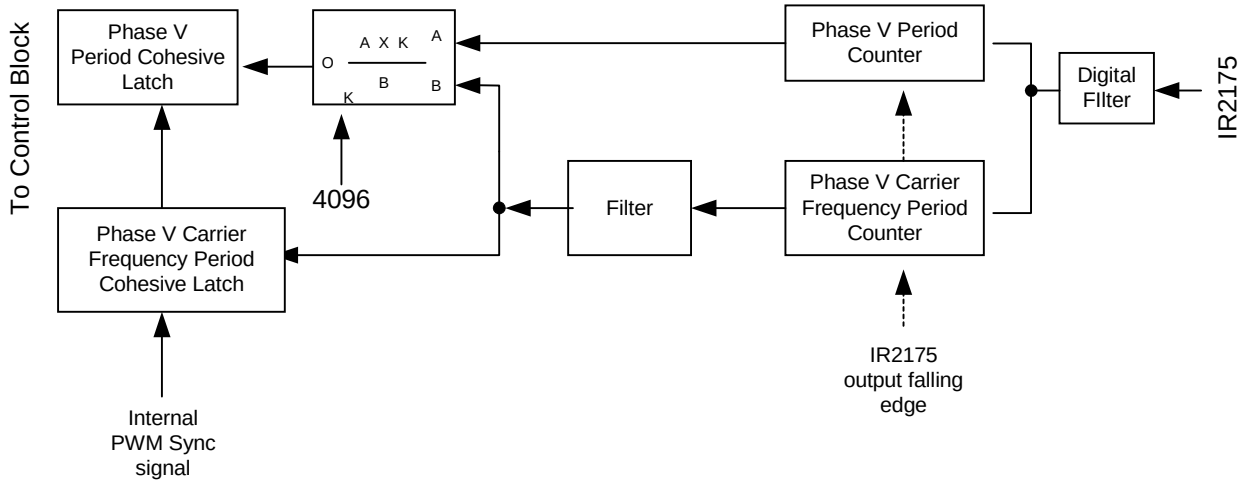


Figure 30. Current Feedback Measurement Block

The current feedback module requires a faster clock to count the duty period of the incoming pulse width modulated signal from the IR2175. This clock rate is designed to work with a frequency between 120 MHz and 133.3 MHz. Figure 31 depicts a simple time chart of counting.

The duty counter, shown as "TA" in Figure 31, captures/latches the value at the falling edge of IR2175 and reset. Then the counter waits for the next rising edge to start counting up. The carrier frequency counter ("TB") captures/latches the value at the rising edge of IR2175 and is immediately followed by re-counting at each IFB event.

At each IFB event, a multiply/divide operation is performed to cancel the temperature drift error of measurement. The following is the basic multiply/divide operation:

$$\frac{TA(n) \times 4096}{Filtered\_TB(n)}$$

Calculation starts immediately after the rising edge of the IR2175 signal as shown in Figure 31. This look-ahead calculation is required to minimize the latency of data availability of the calculation result.

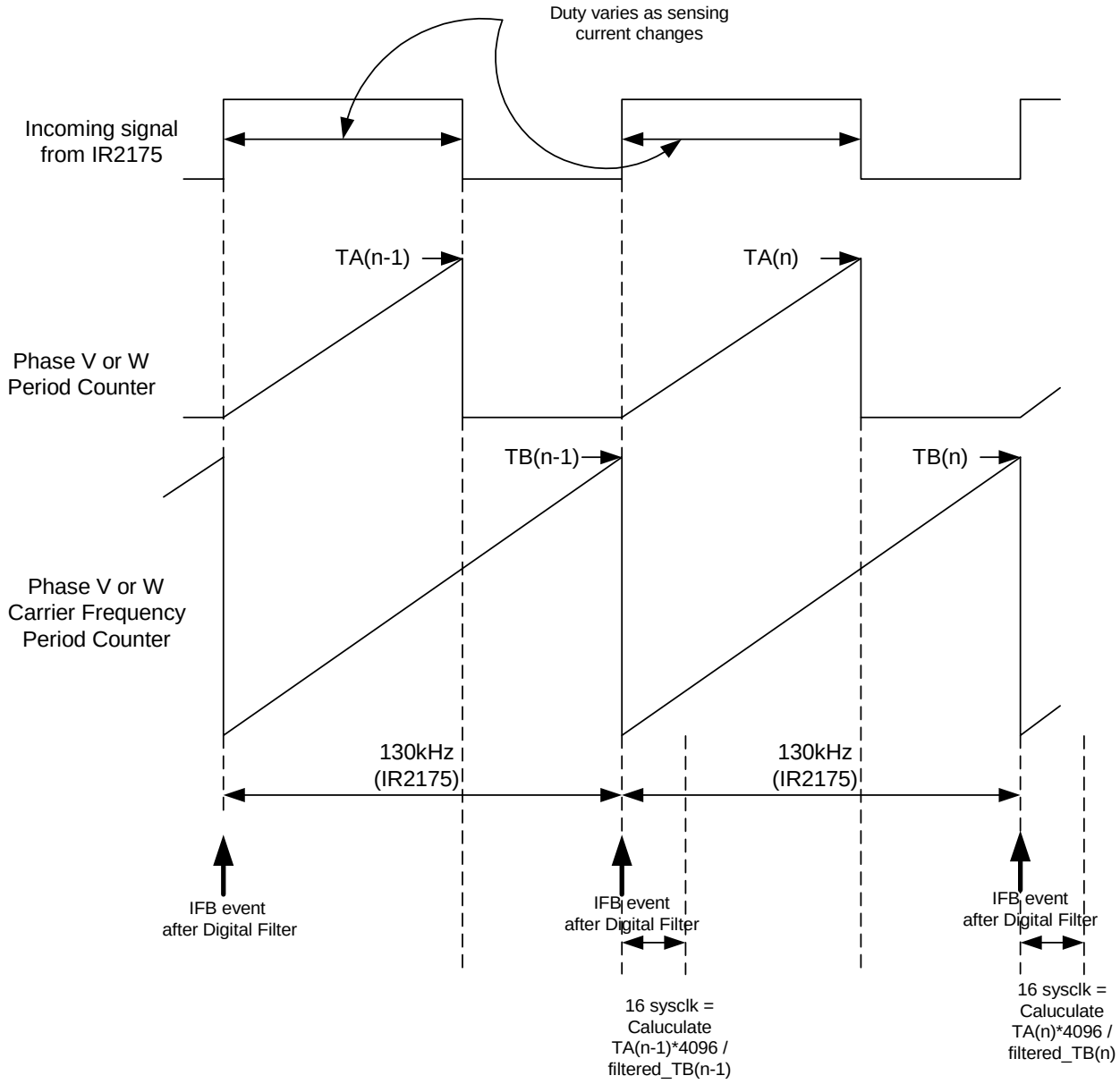


Figure 31. Current Feedback Calculation Timing

International
IOR Rectifier

IR WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245, Tel: (310) 252-7105
http://www.irf.com ***Data and specifications subject to change without notice.***

Sales Offices, Agents and Distributors in Major Cities Throughout the World.