International
IOR Rectifier

DEVELOPER'S GUIDE

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

IRMCK201
Application Developer's Guide

January 10, 2004
Version 2.0

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 datasheets for the IRMCK201 Motion Control IC product. It provides detailed information about the internal design and external interfaces of the products and describes how to configure the operation to conform to the requirements of your custom application. You should read this document if you are developing an application using the IRMCK201 product.

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 will give you the background you need to begin your application development. The Techniques section provides practical "how-to" information, tips and examples to help you during your 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 Control Structure Configuration

This section introduces some of the key configurable parameters used in the IRMCK201 system. Table 1 summarizes the system's configuration parameters and tunable gains. Determining proper settings for the parameters and gain values is the main focus of this document, with the necessary theory explained in the Concepts section and specific examples provided in the Techniques section.

Note: Changing some parameters may require the drive system to be in stand-by mode as opposed to PWM active mode.

Velocity Control Enable/Disable

With this switch, control mode can be configured as either Torque control or Velocity control. When disabling velocity control, reference input becomes torque command. When the velocity control is enabled, the reference input becomes the velocity command.

Slip Gain Enable/Disable (PMAC or induction machine)

The closed loop current control structure can be configured either for induction machine or permanent magnet AC machine. When slip gain is enabled, the feedforward slip gain path is added to a rotation angle to support indirect field orientation of induction machine. Torque producing current reference (Iq command) serves as the input to the slip gain.

Reference Select

The reference, whether it is for velocity reference or torque reference, can be connected to either ±10V analog input via a 12-bit A/D converter, or the host register interface. When analog input is selected, ±10V is converted bipolar digital value, which can be scaled by REF_scale in the associated multiply-divide block.

Current Feedback Interface Select (IR2175 or generic analog input interface)

The IRMCK201 has a built-in interface circuit for IR2175 motor current sensing high voltage ICs. With IR2175 and a shunt resistor, a 10-bit resolution of current feedback data can be obtained within 8.5 microseconds of conversion time. IR2175 IC is a simple yet high performance and low cost solution for servo drive application up to 3.7 kW output power level. For a higher horsepower application, IR2175 becomes impractical due to power dissipation associated with a shunt resistor in series with a motor phase output.

CEMF Feedforward Gain Enable/Disable

The output of q-axis current PI controller has an optional summing junction that adds a feedforward gain block. This gain block input is fed by the host interface register and can be connected to a velocity feedback to achieve CEMF feedforward control. This switch is also useful when the user needs V/Hz open loop control to disable current control PI regulators.

VD Enable

The output of d-axis current PI controller can be disconnected using this switch. When disabled, the d-axis voltage command is connected to the host register interface and the user can directly update/modify the value. This is particularly useful when the user needs an open loop V/Hz control

Parameter	Description
ID scale	d-axis current feedback scaler
IQ scale	q-axis current feedback scaler
V offset	Phase V current feedback offset adjustment
W offset	Phase W current feedback offset adjustment
Kp for IqId	Current regulator Proportional gain
Ki for IqId	Current regulator Integral gain
Vqlim	q-axis current PI amplifier output limit (for both positive/negative limit)
Vdlim	d-axis current PI amplifier output limit (for both positive/negative limit)
PWM Cval	PWM carrier frequency period
Deadtime	Deadtime
PWM mode	Asymmetrical or Symmetrical PWM
2/3 phase modulation select	2 phase or 3 phase modulation
FOC enable	Torque control enable
PWM enable	PWM enable
Slip gain	Slip gain for induction machine
Encoder Angle Scale Factor	Encoder angle scaler
Max Encoder count	Encoder line count
Z pulse initialization value	Z-pulse initialization count value
Speed scaler	Speed scaler
Encoder Type	Encoder type (Hall A/B/C multiplexed or independent)
Z pulse polarity	Logic sense polarity of z-pulse
Kp SPD	Velocity control PI amplifier Proportional gain
Ki SPD	Velocity control PI amplifier Integral gain
IqLim+	Speed regulator output positive limit
IqLim-	Speed regulator output negative limit

Table 1. Parameter Configuration

1.2 Constraints

The following are constraints for use of IRMCK201 with a custom hardware system.

Motor Current Sensing

As a default current sensing device, the system directly interfaces with two IR2175 high voltage ICs to facilitate current feedback from motor phase. IR2175 requires a series shunt resistor on the motor phase and is normally good at up to 3.7 kW applications with 320V DC bus operation.

DC Bus Voltage Feedback Interface

The IRMCK201 interfaces to the ADS7818 (BurrBrown) serial A/D converter (12 bit) for DC bus feedback.

Analog Reference Input Interface

A ±10V analog reference input can be interfaced with ADS7818 (BurrBrown) 12-bit serial A/D converter with analog multiplexer. The IRMCK201 provides the interface logic to these analog components.

1.3 Design Guidelines

Data range

Data range of Iq (Torque producing current in synchronously rotating frame) and Id (Field component current in synchronously rotating frame).

1 PN (Per Normal, 1PN equivalent to rated condition) = ±4095

Overload range = 3PN

Maximum data range = 4 PN = ±16383 including overload range plus additional room for overshoot

General nomenclature

1. Variables either ending or starting with "n" are negative logic indicating low true signals.
2. Unless specified with the associated data range, all variables are Boolean variables.

Motors

Motor can be either a permanent magnet synchronous motor or an induction motor with encoder feedback.

Hall A/B/C and Z-pulse Initialization

When permanent magnet motor is used, the Hall A/B/C data and z-pulse initialization data must be entered to the associated registers in the host register interface. These data are initial position data to make alignment to the initial rotor magnet position.

2 Concepts

This section provides background information and theory of operation for the major IRMCK201 design elements, including control structures, encoder and IR2175 interfaces and the Space Vector PWM module. The concepts are generally common to both permanent magnet motors and induction machines.

Figure 1 shows the detailed control structure of the IRMCK201. Functions are divided into five logic control units:

- Closed loop current control
- Closed loop velocity control
- Sequencing
- Fault handling and DC bus dynamic braking control
- Communication and external interfaces

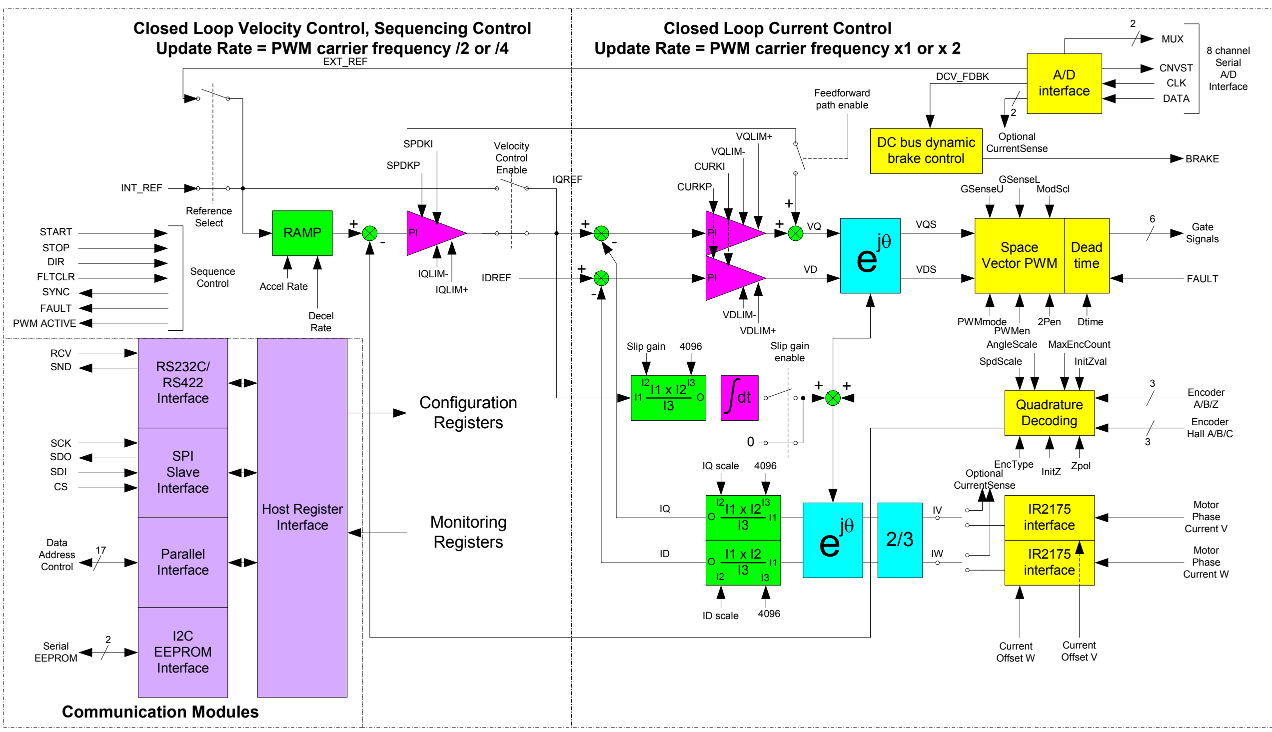


Figure 1. Detailed Control Structure

2.1 Closed Loop Current Control

The closed loop current control structure is based on the synchronously rotating frame field orientation. It employs two Proportional plus Integral (PI) amplifiers to independently regulate torque-producing current, namely Iq, and field flux component current, Id. These currents are oriented in synchronously rotating frame, and contain no AC component of excitation frequency.

The current feedback is obtained through the IR2175 high voltage current sensing IC, and directly transferred to the IRMCK201 IC. Then it is transformed to Id and Iq via a Vector rotator to decouple AC excitation component. Iq is compared against the command torque current while Id is usually compared to zero current reference for PM motor and to magnetizing current reference for induction motor. Two PI controllers produce the required voltage commands, namely Vd and Vq, which are transformed back to stationary frame via Vector rotator and the resulting voltages, Vds and Vqs, contain the AC excitation frequency component. These voltages are passed to the Space Vector PWM modulator and converted into six IGBT gate signals.

The user can configure the closed loop current control for induction machine based field orientation control in addition to permanent magnet motor. The configuration switch, "Slip Gain enable", is provided to support induction machine vector control.

The user can also enable a feedforward control on the Iq regulator, which enhances dynamics of the PI controller by decoupling the CEMF (Counter Electro Mechanical Force) component.

The entire closed loop current control is updated and computed once or twice every PWM carrier frequency period depending on the PWM mode. If symmetrical PWM is selected, then the update rate becomes once every PWM carrier frequency period. If asymmetrical PWM is selected, then the update becomes twice every PWM carrier frequency period.

2.2 IR2175 Current Sensing

Two channels of current feedback interface logic are provided in the IRMCK201 IC. 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.

The offset measurement is automatically added after the 10-bit current measurement has been calculated. The offset value must be measured and compensated.

The 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 by synchronizing signal to PWM, which also drives the internal computation start signal. A block diagram of the current feedback measurement block is shown in Figure 2.

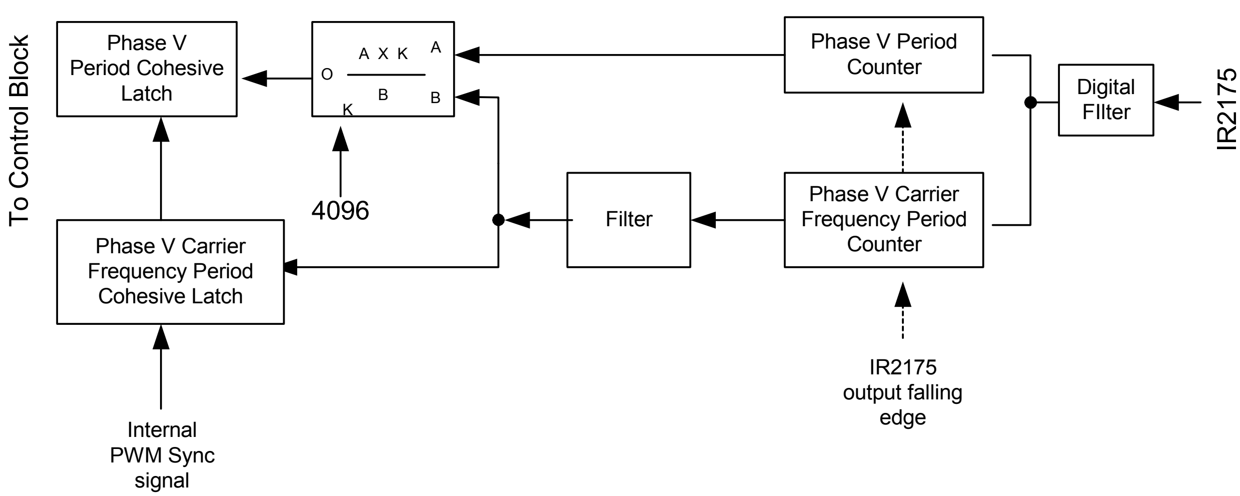


Figure 2. Current Feedback Measurement Block

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 3 depicts a simple time chart of counting.

The duty counter, shown as "TA" in Figure 3, captures/latches the value at the rising edge of IR2175 and reset. Then the counter waits for the next falling edge to start counting up. The carrier frequency counter ("TB") captures/latches the value at the falling 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 falling edge of the IR2175 signal.

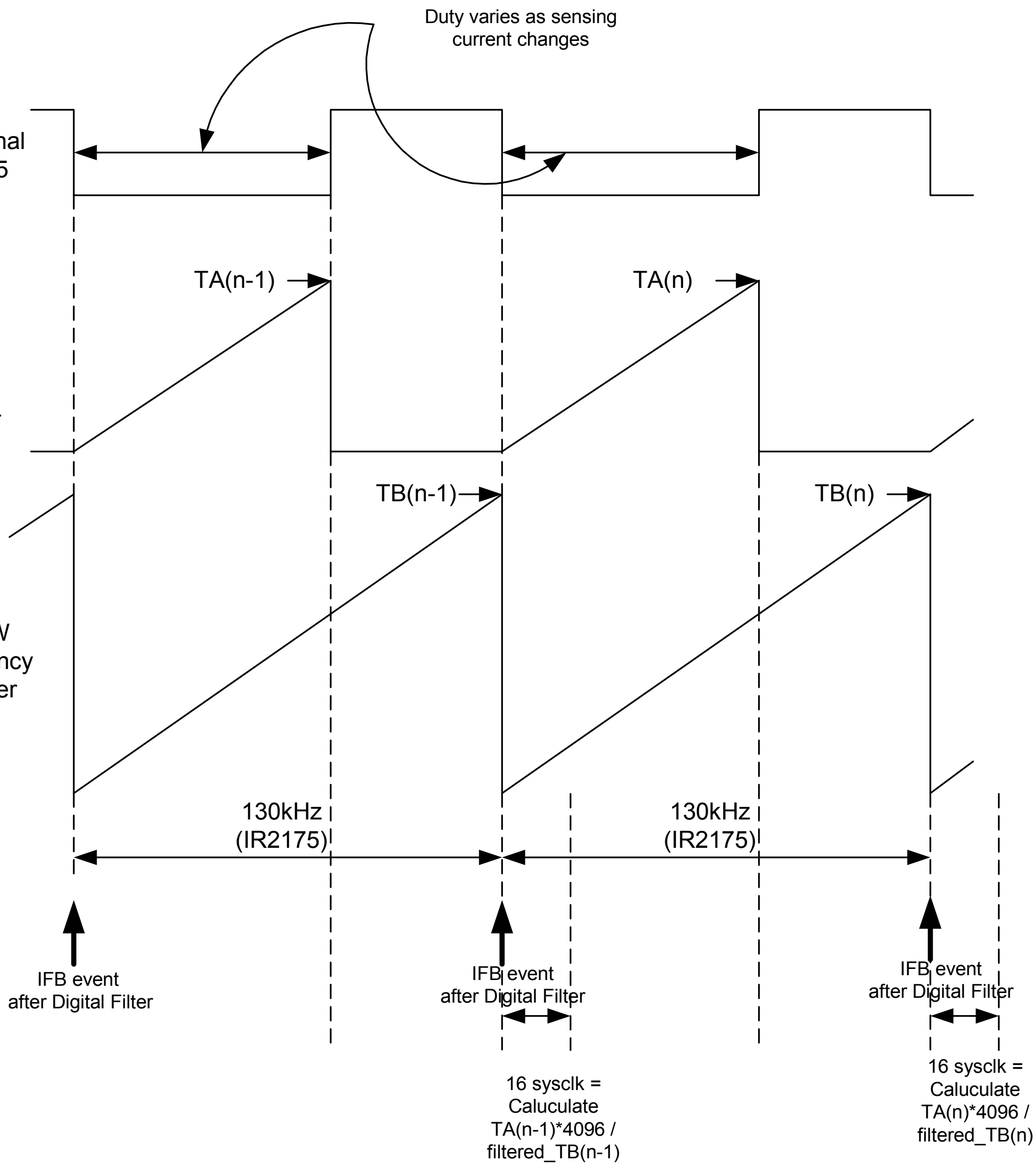


Figure 3. Current Feedback Calculation Timing

2.3 IR2175 Current Feedback Scaling

Maximum input voltage of IR2175	± 260 mV
Shunt resistor value for current sensing	10 mΩ (default)
Maximum Hardware Current (MHC)	± 26 Apeak
Guideline:	
Usable Control Range of current	± 20.8 Apeak to 23.4 Apeak
Overshoot current range	± 2.6 Apeak to ±5.2 Apeak

2.3.1 Selecting Current Sensing Shunt Resistor

Prior to adjusting the current feedback scaling gain, proper shunt resistors need to be selected. The basic guideline of choosing the right value is that the maximum hardware current (MHC) should be greater than the motor's peak overload current plus overshoot current amount. Overshoot current amount should be in the range of 10 to 20% of the maximum hardware current. For the IRMCS2011 hardware system, 10mΩ is the minimum value to be selected due to IGBT current limitation. And as for default factory setting, 10mΩ shunt resistors are installed. If larger shunt resistor values are required, please contact the manufacturer, CADDOCK, with the CD2520FC series surface mount power resistor for the different value.

Trade-offs need to be considered, particularly when the MHC is not close to the motor overload rating; this mismatch may cause coarse resolution of current feedback, say, "Amps per Bit" or "Torque per Bit".

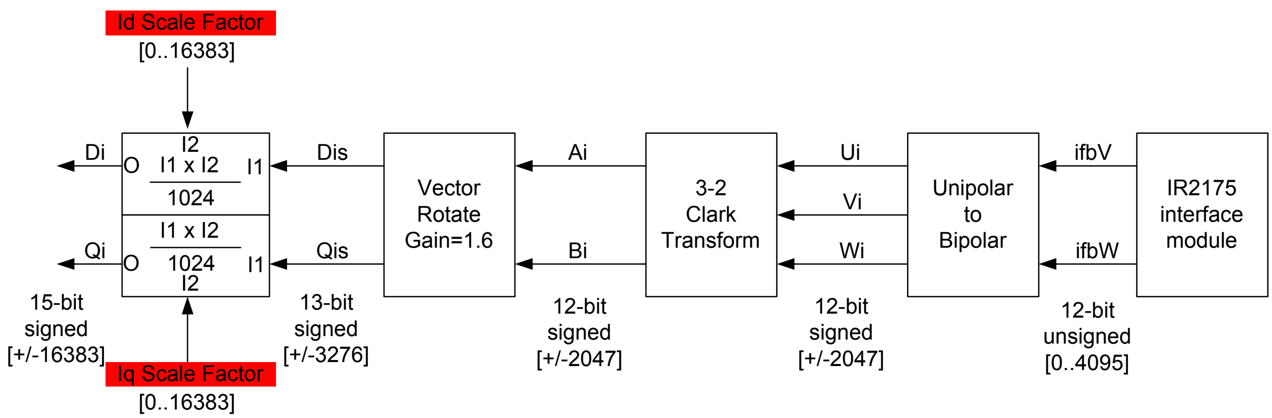


Figure 4. Current Feedback Scaling / Mapping

2.3.2 Adjusting Current Feedback Scaling for Permanent Magnet Motor

Current feedback scaling needs to be adjusted by two parameters, namely **Id scale factor** and **Iq scale factor**, through the host register interface. For a permanent magnet motor application, these two scaling factors are identical. The IRMCK201 is based on the mapping of rated current (continuous current) to ±4095 digital values regardless of the motor current rating. The current control therefore regulates ±4095 as 100% rated torque for torque producing current, "Qi". If the motor has 300% overload capability, then the control regulates a maximum of ±12,287.

The user must know the detailed current feedback scaling path. Figure 4 illustrates the current feedback scaling/mapping implemented in IRMCK201. At the IR2175 interface module, 0 to 4095 is mapped to the maximum hardware current. Then it becomes ±3276 for the maximum hardware current mapping right after the vector rotator due to gain of 1.647. This number then needs to be scaled by user input parameters **Id scale factor** and **Iq scale factor**, which perform scale factor times input divided by 1024. For a permanent magnet motor application with Id = 0, the following equation can be used for determining these scaling gains.

$$Id\,/\,iq\,scale\;factor = \frac{MHC \times 4095 \times 1024}{RC \times 2047 \times \sqrt{2} \times 1.647 \times 0.82}$$

Where MHC is Maximum hardware current in dc(=260mV / Shunt register value in ohm),
RC is Motor Rated Current in rms (continuous current),
0.82 is the maximum modulation index of IR2175 output.

Example:

Motor rated current (RC)	±2.7 Arms / ±3.8 Apeak
Motor overload current	±7.1 Arms / ±10 Apeak

Shunt resistor selection = 20mΩ, since a 10mΩ resistor will have too large unused current range relative to this motor current rating.

Maximum Hardware Current (MHC)	±13 Apeak
Usable Control Range of current	±10 Apeak
Overshoot current range	±3 A

Id/Iq scale factor = 13*4095*1024 / (2.7*2047*1.414*1.647*0.82) = 5165

2.3.3 Adjusting Current Feedback Scaling for Induction Motor

Id scale factor and **Iq scale factor** need to be adjusted for an induction motor application. These two scaling factors are derived by different equations and therefore are usually different values. The IRMCK201 is based on the mapping of rated current (continuous current) to be ±4095 digital values regardless of the motor current rating. The current control therefore regulates ±4095 as 100% rated torque for torque producing current, "Qi". If the motor has 300% overload capability, then the control regulates a maximum of ±12,287. The field component current, "Di" is also regulated 100% field flux with 4095 digital value regardless of the motor physical current or amount of magnetizing current.

In order to find Id scale factor and Iq scale factor, both torque producing current and magnetizing current need to be determined at a rated condition.

Motor nameplate data typically shows the rated current and either no load current or magnetizing current. Torque producing current can be derived from the following equation (all units are in rms):

$$Torque\ Producing\,Current[TPC] = \sqrt{(rated\ current[RC])^2 - (no\,load\ current[NLC])^2}$$

Then, iq scale factor can be derived from the equation:

$$Iq\,scale\ factor = \frac{MHC \times 4095 \times 1024}{TPC \times 2047 \times \sqrt{2} \times 1.647 \times 0.82}$$

Similarly, id scale factor can also be derived from the equation:

$$Id\ scale\ factor = \frac{MHC \times 4095 \times 1024}{NLC \times 2047 \times \sqrt{2} \times 1.647 \times 0.82}$$

Example:

Motor rated current (RC)	±6.0 Arms / ±8.46 Apeak
Motor overload current:	±9.0 Arms / ±12.7 Apeak
No load current (NLC):	±3.8 Arms / ±5.37 Apeak
Maximum Hardware Current:	±13 Apeak

TPC = sqrt($6^2 - 3.8^2$) = 4.64 Arms
Iq scale factor = 13*4095*1024 / (4.64*2047*1.414*1.647*0.82) = 3005
Id scale factor = 13*4095*1024 / (3.8*2047*1.414*1.647*0.82) = 3669

2.3.4 Adjusting Slip Gain for Induction Motor

The slip gain needs to be adjusted for induction machine application. Figure 5 shows the slip gain block diagram. In order to determine the slip gain, the user must have the rated RPM information on the motor nameplate data, and the IRMCK201 closed loop current control update rate information.

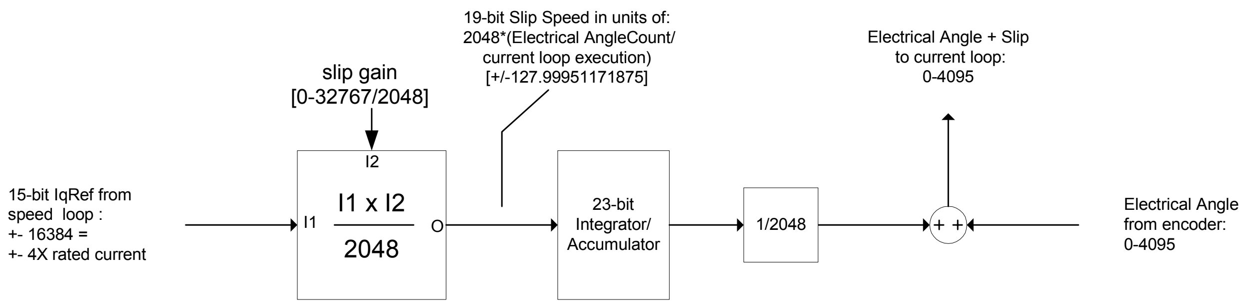


Figure 5. Slip Gain Block Diagram

The following equation should be used to derive the slip gain:

$$Rated\,Hz = \frac{\dfrac{120 \times rated\ frequency}{pole number} - rated\ RPM}{\dfrac{120 \times rated\ frequency}{pole\,number}} \times rated\ \ frequency$$

$$Internal\,slip\,speed = 2048 \times \frac{Rated\ Hz \times 4096}{current\ loop\ update\ rate}$$

$$Slip\ gain = \frac{2048 \times Internal\ slip\ speed}{4096}$$

Example:

Rated RPM:	1725 rpm
Rated frequency:	60 Hz
Pole number:	4
Current loop update rate:	10 kHz

Rated Hz = ((120 * 60 / 4) – 1725) *60 / (120 * 60 / 4) = 2.5Hz
Internal slip gain = 2048 * 2.5 * 4096 / 10000 = 2097
Slip gain = 2097 * 2048 / 4096 = 1048

2.4 Closed Loop Velocity Control

The closed loop velocity control structure can be disabled to form torque mode control instead of velocity mode control. It contains one PI controller and a reference ramp block.

If "SpdLpRate" in the Velocity Control write register group is set to *N*, the closed loop velocity control is updated and executed once every *N* PWM carrier frequency periods. *N* can be from one to seven. (Writing zero disables velocity control) Therefore, if the user changes the PWM carrier frequency, then the velocity control update rate is automatically changed accordingly. The user cannot modify the velocity loop update independently from the PWM carrier frequency.

Input reference can be selected between analog input through A/D interface (ADS7818) or host register update command via communication modules.

2.5 Torque Mode Operation

Torque mode operation provides the user with the option of controlling motor speed from an external source. In this mode, the user supplies the speed ramp and PI controller, and provides IQREF (quadrature reference current) input through the host register interface in order to control torque (motor current). The user's algorithm should calculate and provide a new IQREF value at the start of each PWM period, which is indicated by the SYNC output signal. The EncCntR host read register provides encoder position data to be used in the calculation.

The following host register settings configure the system for torque mode operation:

- Set "SpdLpRate" to 0 in the Velocity Control write register group to disable the speed loop.
- Set "IqRefSel" in the SysConfig write register to 1 to enable use of the "IqRefW" write register.

Once the system is configured for torque mode operation, a quadrature reference current value should be written to the "IqRefW" host write register at the start of each PWM period.

2.6 Encoder Interface

The encoder interface module is included in IRMCK201. The interface assumes the encoder signal to be an incremental and rectangular pulse train waveform with single ended output. All differential signals should be converted to single ended signal. IRMCS2011 has this conversion circuit as shown in Figure 7. The IRMCK201 can accept a maximum 1 MHz encoder frequency. Quadrature signals are used for angle generation in the current control loop but not for speed measurement.

2.6.1 Initialization & Angle Generation

The Sanyo Denki P30B06040DXS00M motor, for example, has a wire saving type 2000 line count pulse encoder. Hall_A, Hall_B, Hall_C information is available only at power up and it gives the approximate 60-degree electrical angle position. This motor has 4 pole pairs (8-pole motor) and therefore it has 15 degrees resolution (360/4/6 = 15) mechanically. If the middle value of that range is taken, the maximum error of the initial position is 30 degrees electrically.

The motor can start running with Hall_A, Hall_B, Hall_C information and within one revolution, a Z-pulse occurs, which gives precise angle information.

The counter EMF profiles can be measured by rotating the shaft manually. Zero angle is defined at negative-going zero crossing of the U phase counter EMF.

When you look at the motor from the front-end shaft side and the shaft is rotating CCW, the output waveforms are shown in Figure 6 in relation to the internal angle counters.

In this figure, a 4-pole motor (2-pole pairs) is assumed, and therefore two complete CEMF cycles are required for one revolution of the mechanical turn. The line-to-neutral voltage and the line-to-line voltage are CEMF waveforms. The line-to-line voltage can be measured at motor three phase voltage terminals. The top trace, Mechanical angle, is the internal up/down counter of the encoder Quadrature A/B pulses. The counter value needs to be initialized at power-up by sensing the Hall A/B/C signal position and reading the corresponding counter data. The counter value is initialized again upon receiving the first z-pulse with the predefined counter data. This two-stage initialization is required since the first initialization only provides a coarse angle resolution (60 electrical angle degree resolution) due to Hall A/B/C signals.

Hall A/B/C data need to be entered through the Motor Configuration dialog box in the ServoDesigner tool. The data are mechanical angle based counter value as shown in Figure 6's top trace. Initialization needs to be done so that the counter value becomes zero at the negative-going zero crossing of the phase U. It is also important that the mechanical angle counter rolls over to zero when it reaches MAX_COUNT, which is equal to the encoder line count times four (i.e. for 2000 line count encoder = 8000). It is also important to provide the data in the middle of Hall A/B/C state changes. A total of six counter values need to be entered for all three-bit combinations of Hall A/B/C. The Z-pulse data also needs to be provided in a similar fashion.

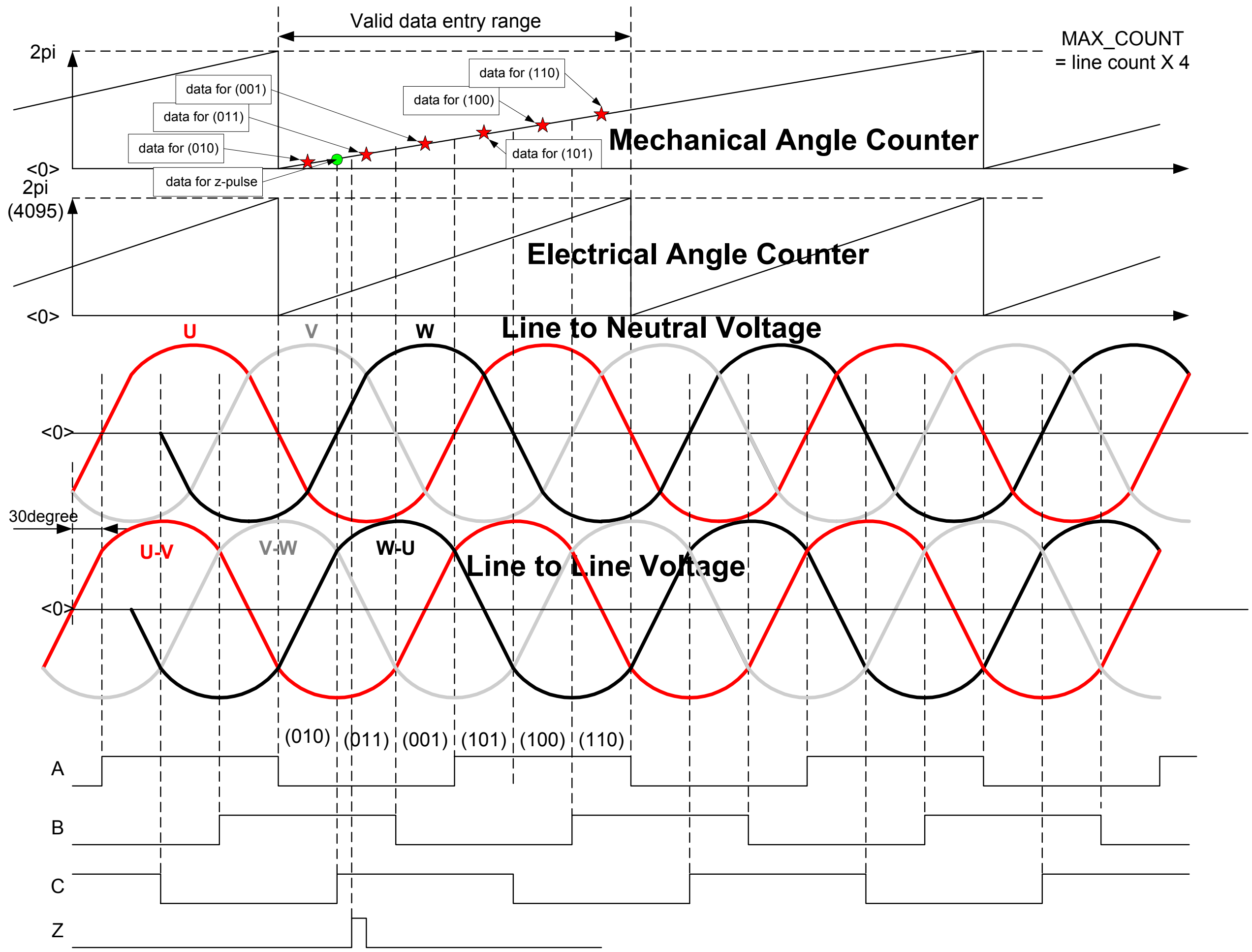


Figure 6. Example of Counter-EMF Profile and Hall Signals

2.6.2 Initializing and Monitoring Encoder Counter Values

The encoder counter is initialized and monitored through the host register interface. Two counters are provided. The main mechanical angle counter is a 16-bit field that wraps to 0 when a configured maximum count (MAX_COUNT) is reached. This is the counter that must be initialized at power-up as described above. An auxiliary 32-bit counter that does not wrap is provided for optional use with position control applications.

The maximum value for the 16-bit counter is configured using the "MaxEncCnt" field of the QuadratureDecode write register group. The maximum count is dependent on the encoder and should be set to represent a 360-degree physical angle, as described above. The 16-bit count value can be initialized by writing to the "EncCntW" field of the QuadratureDecode write register group. The current counter value can be read at any time from the "EncCntR" field of the QuadratureDecodeStatus read register group and represents the current encoder angle.

There is no maximum count setting for the 32-bit counter. The 32-bit count value can be initialized by writing to the "EncCnt32bW" field of the 32bitQuadDecode write register group. The current 32-bit counter value can be read at any time from the "EncCnt32bR" field of the 32bitQuadDecodeStatus read register group and can be used to determine the number of encoder revolutions, which in turn can be translated to the distance traveled by a robotic arm.

The two counters are independent. That is, initializing the 16-bit counter does not affect the value of the 32-bit counter and initializing the 32-bit counter does not affect the value of the 16-bit counter.

2.6.3 *Encoder Application Example*

An encoder application is shown in Figure 7. The left side of the schematic shows the encoder input signals A+/-, B+/-, Z+/- connected to passive components then connects to a differential receiver (DS3486M). The buffered outputs are connected as follows:

1. ENA, connected to IRMCK201 pin 93
2. ENB, connected to RIMCK201 pin 92
3. ENZ, connected to IRMCK201 pin 91
4. HALLA, connected to IRMCK201 pin 98
5. HALLB, connected to IRMCK201 pin 96
6. HALLC, connected to IRMCK201 pin 95

In reduced signal applications, the IRMCK201 has an internal latch that can capture the HALLA, HALLB and HALLC signals when mapped to the ENA, ENB and ENC inputs. The latch closes when the RESETN signal (pin 35) goes high. Refer to the IRMCK201 datasheet for specific timing information.

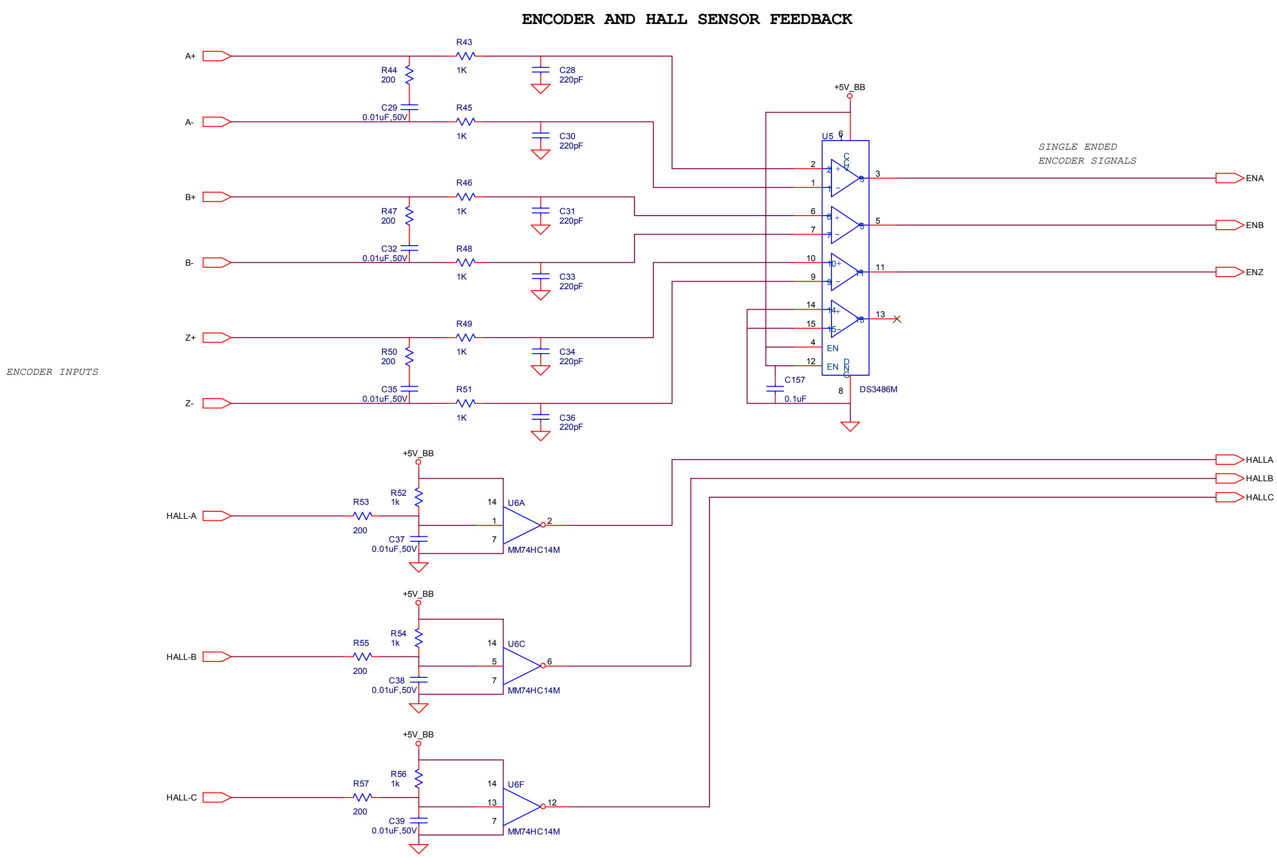


Figure 7. Encoder Input Example

Note that HALLA, HALLB, and HALLC signals are inverted.

2.7 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.

2.7.1 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 8). 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 8. The gain characteristic of the SVPWM module is given in Figure 9. The vertical axis of Figure 9 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

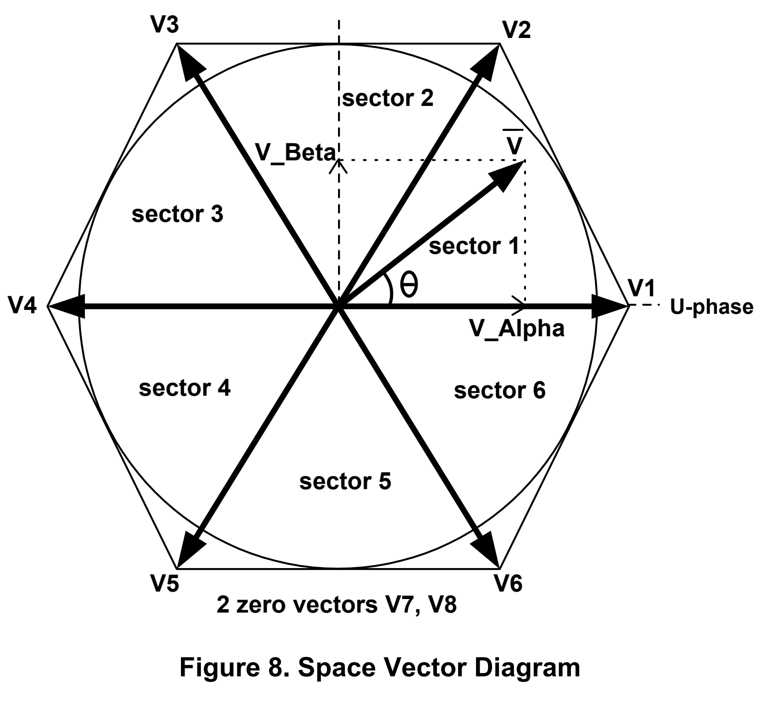


Figure 8. Space Vector Diagram

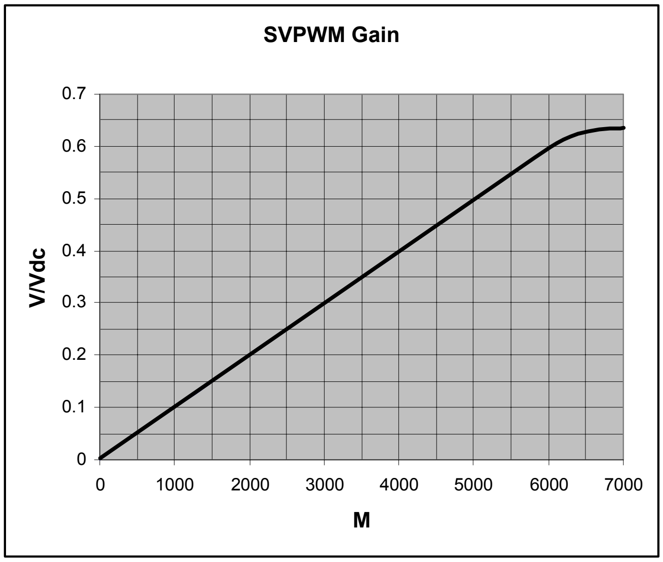


Figure 9. Transfer Characteristics

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 10 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 9) of the PWM modulator reduces and becomes non-linear in the over modulation region.

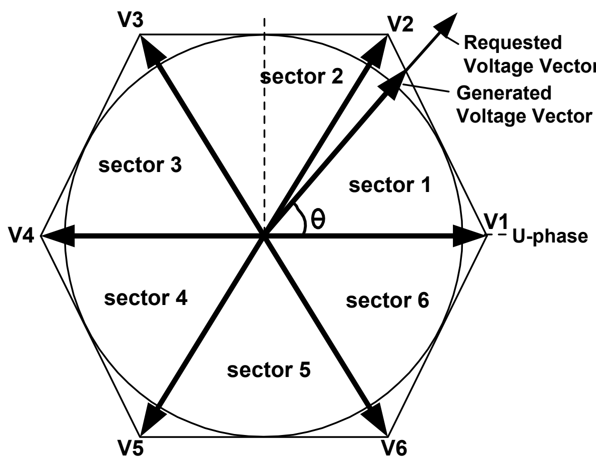


Figure 10. Voltage Vector Rescaling

2.7.2 PWM Operation

Referring to Figure 11, 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 internal 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 SYNC, a new set of Space Vector times and vectors are readily available for actual PWM generation (PhaseU, PhaseV, PhaseW) by sub module PwmGeneration.
Figure 11 (3-phase modulation) illustrates the PWM waveforms for a voltage vector locates in sector I of the Space Vector plane (Figure 8). The gating pattern outputs (PWMUH … PWMWL) include deadtime insertion (describe in later section).

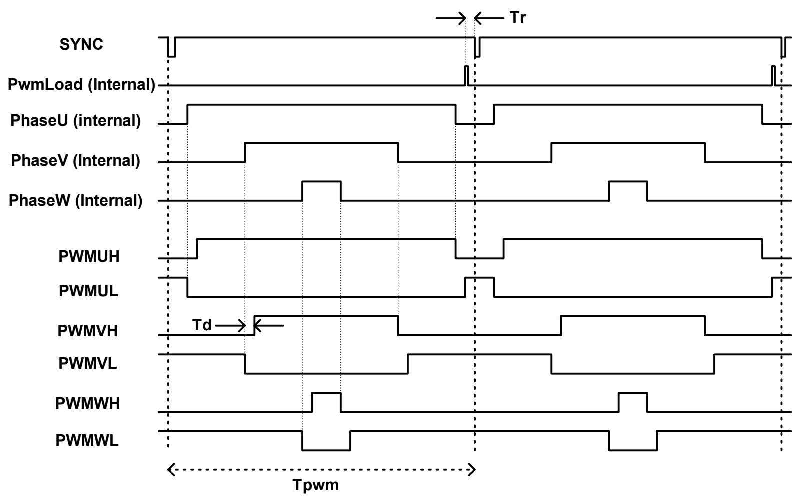


Figure 11. 3-phase Space Vector PWM

2.7.3 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.

2.7.4 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.

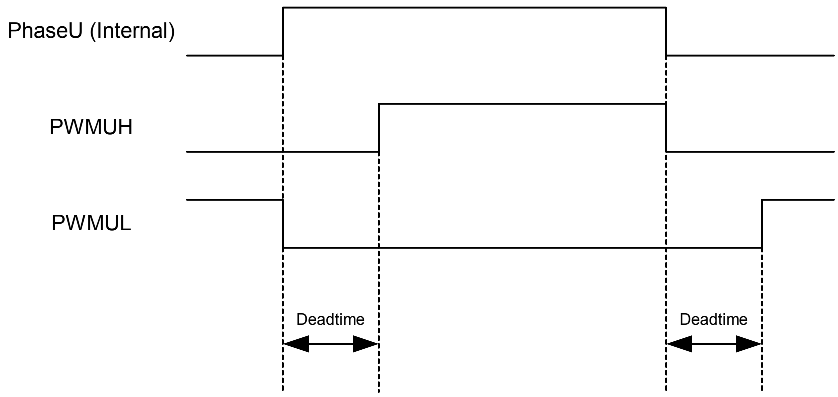


Figure 12. Deadtime Insertion

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.

2.7.5 PWM Mode Select (PwmMode)

The SYNC signal and the PWM internal Reload signal are completely simultaneous and synchronous events. The internal Reload signal resets the PWM counters inside the PwmGeneration module and it is an indicator reference of a new PWM cycle. The rate of signal generation of the SyncPulse and Reload signals is controlled via the configurable parameter PwmMode.

Figure 12 shows the timing of SYNC and Reload signals for the two different PwmMode settings. Asymmetrical mode operation (described in Section 2.7.6) can be obtained by setting PwmMode = 0.

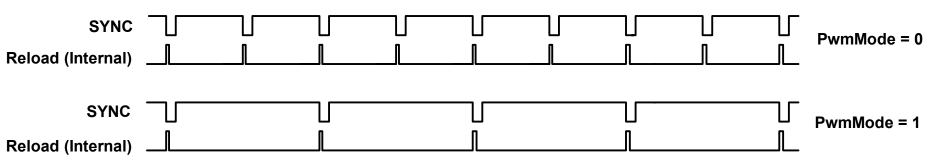


Figure 13. nSYNC and Reload Timing

2.7.6 Symmetrical and Asymmetrical Mode Operation

There are two modes (configured by PwmMode: 0 – Asymmetrical, 1 to 3 - Symmetrical) of operation available for PWM waveform generation, namely the Center Aligned Symmetrical PWM (Figure 11) and the Center Aligned Asymmetrical PWM (Figure 14). The volt-sec can be changed every half a PWM cycle (Tpwm) since PwmLoad occurs every half a PWM cycle (compare Figure 11 and Figure 14). With Symmetrical PWM mode, the inverter voltage can be changed at the rate of the inverter switching frequency. With Asymmetrical PWM mode operation (PwmMode = 0), the inverter voltage can be changed at 2 times the rate of the switching frequency. This will provide an increase in voltage control bandwidth, however, at the expense of increased current harmonics.

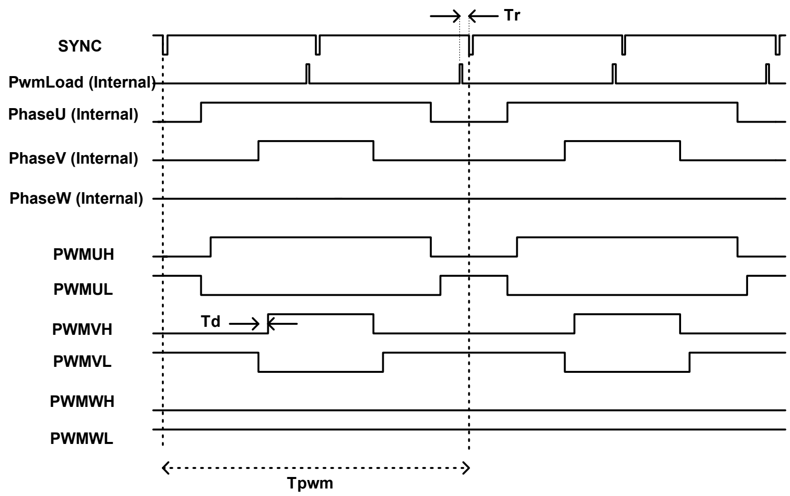


Figure 14. Asymmetrical PWM Mode

2.8 Communication

This section describes the external interfaces supported by the IRMCK201. These include the microprocessor-based interfaces used to access the Host Registers; the discrete I/O interface used for standalone operation; and the analog I/O interface provided for diagnostic purposes.
The IRMCK201 contains a rich set of externally addressable "Host" registers documented in the Reference section of this guide. There are three physical interfaces that can access the Host Registers; RS-232, SPI and Host Parallel.

2.8.1 RS-232 Serial Interface

The RS-232 interface implements a byte serial physical layer in addition to an error checking protocol layer.

The slowest of the three, the Serial Interface is used for inter-board communications typically using cables as the connection medium. The IRMCK201 has implemented 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 signal bit errors. Please refer to the RS-232 protocol documentation in the Reference Section for the specific protocol definition. The RS-232 Serial Interface supports two baud rates based on the signal level on pin 53 of the IRMCK201:

Input at a logic low or 0 = 57,600 baud
Input at a logic high or 1 = 1,031,250 baud

The coding of the bit-serial data is US ASCII, 8 data bits, 1 stop bit and no parity.

The following table describes the physical layer signals of the RS-232 interface.

Signal Name	Direction	Description
TX	Output	A bit-serial signal originated by the IRMCK201 in response to a microprocessor-generated request.
RX	Input	Bit-serial data sent to the IRMCK201 by the microprocessor to interrogate one of the Host Registers.

Table 2. External RS-232 Signal Description

2.8.2 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 the Reference Section for the protocols required to access the Host Registers, and the SPI timing section of the datasheet for the physical layer specifications.

The following table describes the physical layer signals of the SPI interface.

Signal Name	Direction	Description
SCLK	Input	Serial clock generated by the SPI master logic.
MISO	Output	Serial data: Master Input and Slave Output.
MOSI	Input	Serial data: Master Output and Slave Input.
CS	Input	Chip Select signal. Used to qualify the SCLK, MISO and MOSI signals.

Table 3. External SPI I/F Signal Description

2.8.3 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 bases bus protocols. Refer to the Host Parallel documentation in the Reference Section for the protocols required to access the Host Registers, and the Host Parallel timing section of the datasheet for the physical layer specifications.

The following table describes the physical layer signals of the Host Parallel interface.

Signal Name	Direction	Description
HPOEN	Input	When logic low, or 0, indicates the beginning of a parallel data transfer cycle.
HPWEN	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 IRMCK201
HPD[7:0]	Input/Output	An 8-bit wide data bus.
HPA	Input	Address attribute signal. When high, or a logic 1, indicates that the data on the HPD[7:0] bus is a address to be loaded into the IRMCK201 address register.

Table 4. External Host Parallel I/F Signal Description

2.8.4 Synchronization of PWM Cycle to an External Microprocessor

A dedicated SYNC signal is provided on the IRMCK201 (pin 63) that allows synchronization of the internal IRMCK201 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 IRMCK201 while the IRMCK201 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 IRMCK201 is starting a new PWM cycle. Refer to the ADC System Level Timing section of the IRMCK201 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 programmed by varying the "PwmMode" field of the Write Register 0xE. Refer to section 2.7.6 for detailed information.

The SYNC pulse width is suitable for connecting opto-isolation circuitry between the IRMCK201 and the microprocessor.

2.9 External Interfaces

2.9.1 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.

The signals are described in Table 5.

Signal Name	Direction	Description
START	Input	A 100 nsec pulse on this signal enables PWM and FOC, equivalent to setting the "PWMEN" and "FOCEN" bits of the System Control register.
STOP	Input	A 100 nsec pulse on this signal disables PWM and FOC, equivalent to clearing the "PWMEN" and "FOCEN" bits of the System Control register.
IFBCAL	Input	Hold this signal high for 100 msec to initiate a calibration cycle (enables PWM and disables FOC).
FLTCLR	Input	A 1 µsec pulse on this signal clears a drive fault condition. Equivalent to setting the FAULT CLEAR bit of the Fault Control register.
FAULT	Output	This signal indicates the presence of a drive fault condition. The level is high when any of the bits in the Fault Status register are set.
SYNC	Output	This signal is held low for 3 µsec on each PWM period. (The falling edge indicates the start of the PWM period.)
PWMACTIVE	Output	High while PWM output is enabled, equivalent to the value of the "PWMEN" bit in the System Status register.

Table 5. External Interface Signal Description

NOTE: When the "EXT_CTL" bit in the System Configuration register is set to "0", the STOP signal is functional, but all other external interface signals are inactive.

Use the following procedure to configure the discrete I/O interface:

- Set the "IfbOffsEnb" bit in the System Control register to "1" if you want to use the IFBCAL signal to automatically calculate current feedback offset values.
- Write a "1" to the "ExtCtrl" bit in the System Configuration register to enable the external interface pins.
- Write a "1" to the "RmpRefSel" bit in the System Configuration register to select the reference A/D converter as the speed reference source.

2.9.2 Analog I/O Interface

IRMCK201 with associated A/D, multiplexer, and sample/hold circuit provides an analog input to control reference speed. The user must enable the external I/O mode to select speed or torque command source input.

Analog Input

Figure 15 shows the typical hardware circuit for analog input interface. In this configuration, the circuit has 4-channel multiplexed analog input with sample/hold and high voltage attenuation amplifier. These four channels are:

1. Speed or Torque analog command input, +/- 10V mapped to 0-5V input to the ADS7818 12-bit A/D input.
2. DC bus voltage input, 0-500V mapped to 0-5V input
3. Optional Hall effect current sensor input (Phase V) with simultaneous sample/hold, 0-5V input
4. Optional Hall effect current sensor input (Phase W) with simultaneous sample/hold, 0-5V input

Interfacing signals (5V logic) to IRMCK201 are:

1. ADCLK, connected to IRMCK201 pin 61, ADS7818 A/D converter clock input, 8.33MHz clock,
2. ADOUT, connected to IRMCK201 pin 62, ADS7818 A/D converter data output
3. ADCONVST, connected to IRMCK201 pin 60, ADS7818 A/D converter conversion start input
4. ADMUX0, connected to 4052 multiplexer pin 10, 4-to-1 multiplexer address 0
5. ADMUX1, connected to 4052 multiplexer pin 9, 4-to-1 multiplexer address 1
6. RESAMPLE, connected to 4066 analog switch pin 12 and pin13, Sample/hold control signal

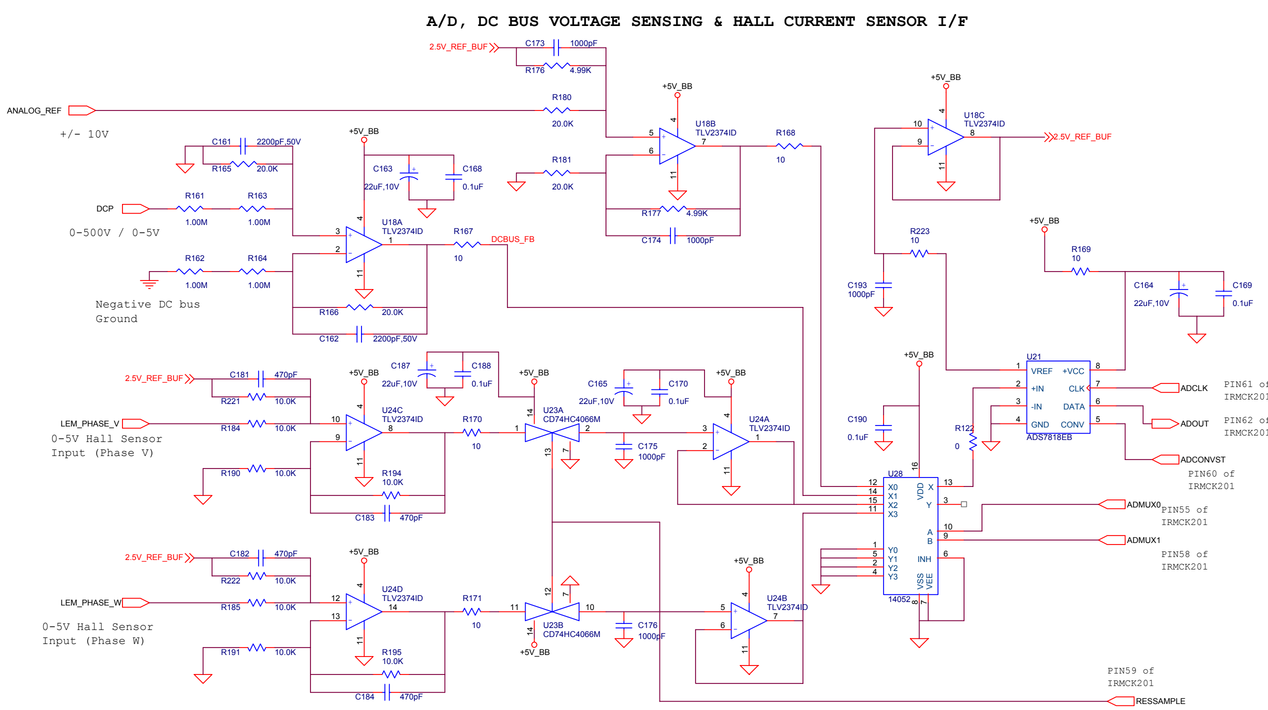


Figure 15. Analog Input Example Circuit

In this circuit, the IRMCK201 automatically scans through A/D conversion of all four channels at the beginning of each PWM cycle (SYNC output, pin 63). Upon each SYNC pulse output, the 4066 sample/hold circuit is placed in hold mode (RESSAMPLE=low) for a short period, then A/D conversion starts at the channel 2 (ADMUX0=0, ADMUX1=1), which is Phase V Hall effect sensor input. The second conversion follows at the channel 3 (ADMUX0=1, ADMUX1=1), which is Phase W Hall effect sensor input. The third conversion takes place at the channel 0 (ADMUX0=ADMUX1=0), which is analog reference input (+/-10V input). The last conversion at the channel 1 (ADMUX0=1, ADMUX1=0) starts for the DC bus voltage input.

Analog Output

The diagnostic purpose D/A interface provides four diagnostic data items through 8-bit PWM and is intended for use with external RC filters for oscilloscope display. For each of the four signals, the user can select the data source from the items shown in Table 6. Each signal is encoded as a pulse-width modulated 8-bit value output at a frequency of 128 KHz. 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 Qi and Di, 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).

Value	Data Source
1	DC bus voltage
2	V phase current
3	W phase current
4	(not valid)
5	Speed PI reference
6	Speed PI feedback
7	Speed PI error
8	IQ ref
9	Q axis voltage (Qv)
10	D axis voltage (Dv)
11	Angle in quadrature counts
12	Q axis current (Qi)
13	D axis current (Di)
14	A axis (stationary frame) voltage Av
15	B axis (stationary frame) voltage Bv

Table 6. Data Sources for Analog Output

2.10 Sequencing Control

The sequencing control is contained in the IRMCK201 system to facilitate basic I/O sequencing. It can be directed either by local discrete I/O pins or host register interface. STOP is always activated by either the host interface register or the local STOP input pin.

Internally, IRMCK201 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 or host register interface or internal drive conditions such as overcurrent, overvoltage, etc. The state diagram is shown in Figure 16.

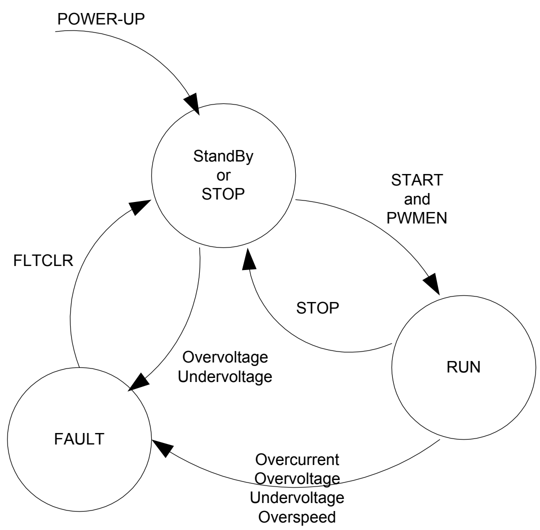


Figure 16. State Diagram and Sequencing

2.11 Fault Handling and DC Bus Dynamic Braking

The IRMCK201 has built-in drive fault and protection features. Table 7 summarizes the types of drive fault conditions.

Fault	IRMCK201 LED pin (pin20, pin21) Indication	Status Indication on Host Register Interface	Description
Overcurrent / Overtemp	REDLED = high GREENLED = low/high toggle	Fault Status Read Register, field GATEKILL = 1	Overcurrent or overtemperature occurred. The IGBT gate driver (IR213x) disables gate drive outputs, momentarily latches a fault condition and asserts GATEKILL to the IRMCK201. This activates the fault latch inside the IRMCK201.
Overvoltage	REDLED = high GREENLED = low/high toggle	Fault Status Read Register, field OVER VOLT = 1	Overvoltage of the DC bus occurred. Only the fault latch inside the IRMCK201 is activated.
Overspeed	REDLED = high GREENLED = low/high toggle	Fault Status Read Register, field OVER SPEED = 1	The speed of the motor exceeded the maximum speed. Only the fault latch inside the IRMCK201 is activated.
Overrun	REDLED = high GREENLED = low/high toggle	Fault Status Read Register, field PTIME FAULT = 1	The computation of algorithm exceeded the selected PWM carrier frequency period. Only the fault latch inside the IRMCK201 is activated.
Low voltage	REDLED = high GREENLED = low/high toggle	Fault Status Read Register, field LOW VOLT = 1	The bus voltage dropped below 120V. Only the fault latch inside the IRMCK201 is activated.
DC bus power loss	REDLED = low GREENLED = low	N/A	DC bus power is not energized.

Table 7. Drive Fault Conditions

When any drive fault occurs, the PWM output is disabled and the gate signals off the IRMCK201 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 Fault Clear bit in the Fault Control host register or the FLTCLR discrete I/O external interface pin.

2.11.1 Gatekill Structure and Overcurrent/Overtemperature Fault

For example, IRMCS2011 design platform for IRMCK201 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 IRMCS2011. The signal is fed to an opto-coupler device to trigger the signal to IRMCK201 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 signal off the IRMCK201 until the user initiates a FAULT CLEAR action.

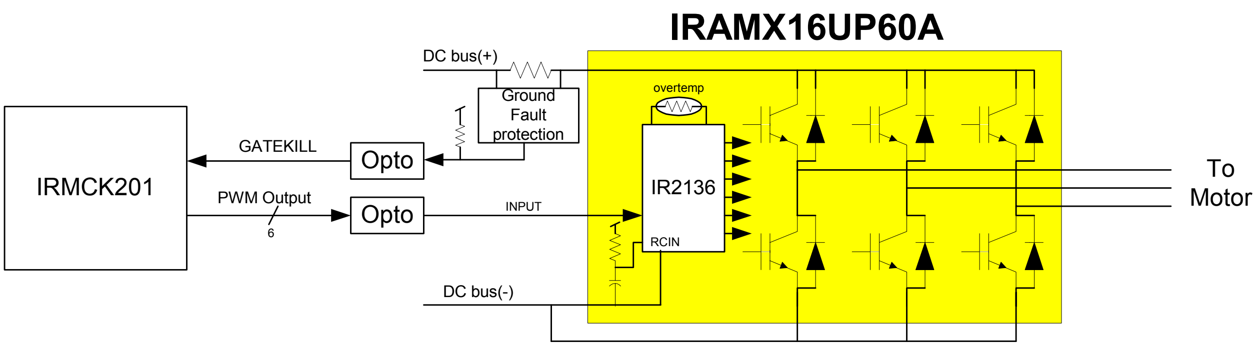


Figure 17. Protection Circuit Block Diagram

Figure 17 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 also 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 IRMCS2011 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 IRMCK201 disables all PWM gate signals and latches GATEKILL. It also disables PWM output, forcing the "PwmEnbW" and "FocEnbW" bits of the System Control register to "0".

To reset a fault condition, first write a "1" to the "FltClr" bit of the Fault Control register. This clears the fault in the IRMCK201. 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.

2.11.2 DC Bus Faults and DC Bus Braking

Table 8 describes the actions taken at various DC bus voltage levels based on IRMCS2011. When the Brake IGBT ON level is detected, the Brake IGBT is turned on and the BRAKE signal in the DC Bus Voltage read register is latched until the DC Bus voltage falls below the Brake IGBT OFF level, at which point Brake IGBT is turned off. Overvoltage and low voltage conditions produce corresponding drive faults, as described in Table 7.

DC Bus Action	DC Bus Level	Description
Overvoltage	410V	Overvoltage of the DC bus occurred. IGBT gate driver is disabled and the latched FAULT pending to the IRMCK201.
Brake IGBT ON	380V	Brake IGBT is turned ON above this voltage.
Brake IGBT OFF	360V	Brake IGBT is turned OFF below this voltage.
Nominal bus voltage range	360V-120V	Nominal DC bus voltage range
Low voltage	120V	Lowvoltage fault. PWM is disabled.

Table 8. DC Bus Voltage Level Management

2.12 LED interface

The operating state of the IRMCK201 is indicated by two pins, namely REDLED and GREENLED. There are four indication modes.

Mode 1 indicates normal mode. A red flashing LED appears automatically right after successful power-up reset.

In Mode 2, a green flashing LED appears after PWM carrier frequency period value is written.

A static red LED indicates Mode 3. This is a drive fault condition.

The LED is not lit in Mode 4. This means that either a fatal error has occurred in the IC or the hardware power supply has a problem.

Mode	LED Indication	Description
Mode 1	REDLED = low/high toggle GREENLED = low	IRMCK201 is functioning normally.
Mode 2	REDLED = low GREENLED = low/high toggle	IRMCK201 configuration has been done correctly and SYNC pulse is generated followed by PWM carrier frequency period programming.
Mode 3	REDLED = high GREENLED = low/high toggle	A drive fault condition is pending.
Mode 4	REDLED = low GREENLED = low	IC is not functioning indicating a possible fatal error, or the power supply has failed.

Table 9. LED Modes

3 Techniques

3.1 Drive Parameter Setup

ServoDesigner™ includes an Excel workbook file that partially automates the procedure of calculating the appropriate values for configuration and tuning parameters. In the workbook, you enter motor nameplate data and parameters specific to your application, and Excel formulas calculate the appropriate values for certain motor configuration parameters and write registers. In Excel, you export these values to a text file, and in ServoDesigner, you can import the text file to fill in the motor configuration settings and register values. Then, when you execute the Starting Angle and Configure Motor function in ServoDesigner, the appropriate values are written to the IRMCK201.

The Excel Workbook File

The Excel workbook file is named "IRMCx201-DriveParams.xls". Double click the file to open it in Excel.

At the bottom of the workbook window, you’ll see a number of sheet tabs, which you can use to select the worksheet you want to display. You can use the arrows in the bottom left corner of the window to scroll through the sheet tabs.

The first tab, labeled “Calculation,” selects a worksheet that shows the values used to calculate the exported parameter and register values. The second tab, “IRMCx201_Parameters,” selects a worksheet that shows the results of the calculations. This is the worksheet that must be exported to a text file for use with ServoDesigner. The remaining tabs select motor setup worksheets, which have a format similar to that of the Calculation worksheet, but are pre-initialized with values appropriate for specific motors. Each tab is labeled with the name of the motor and an associated motor ID number in parentheses.

If you want to use one of the supported motors, first note its ID number (as shown on the sheet tab). Then click on the Calculation sheet tab and enter the motor ID number in the blue box at the top of the Calculation worksheet. When you press Enter or click to another field on the worksheet, the values from your selected motor setup worksheet are copied to the Calculation worksheet.

If you want to use an unsupported motor, you can modify one of the existing motor setup worksheets or use the last worksheet, labeled “New Motor.” In addition to entering values specific to your motor (as described below), be sure to change the motor name and description at the top of the worksheet and the motor name on the sheet tab (double-click the label on the tab to change it). When your new motor setup worksheet is complete, click on the Calculation tab and enter the motor ID number in the blue box at the top of the Calculation worksheet to copy the values from your motor setup worksheet. You can modify values directly on the Calculation worksheet, but the values will be lost if you later copy another motor setup worksheet by entering a motor ID in the blue box.

Enter Drive Parameters in Excel

The first stage of configuring drive parameters involves entering the correct settings for the motor you’re using and the specific requirements of your application. If you’re using one of the supported motors, you can skip Steps 1 and 2.

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

Click on the sheet tab that selects the motor setup worksheet for your motor. The first two lines of the motor setup worksheet describe the motor. An example is shown below. Double click on line 1, column B to change the motor name, and double click on line 2, column B to enter a description of the motor. The motor name and description are for your own use; they’re not used in the calculations and not exported to ServoDesigner.

Double click on line 1, column E to specify the motor type: “1” for a permanent magnet motor or “2” for an induction motor.

	A	B	C	D	E	F	G	H	I	J
1	Motor:	P30B06040DXS00M			1	(1 : Permanent Magnet Motor, 2: Induction Motor)				
2		Sanyo Denki 400W								

Step 2. Enter Motor Information.
The motor information section of the motor setup worksheet contains parameter settings that you should be able to find in the datasheet for your motor or on its nameplate.

"=================== Motor Information ==========================="			
Rated Speed	**3000**	rpm	
L_phase	**6.44E-03**	H	(line to line Inductance) / 2
R_phase	**1.4**	ohms/ph	(line to line Resistor) / 2
Rated Amps	**2.7**	Arms	
(NLC)No Load Current	**0**	Arms	
Inertia of Motor	**2.55E-05**	Kg-m2	
(Kt) Torque Constant	**0.533**	N-m/Arms	
(Ke) Voltage Constant	**18.6**	V ln-rms/krpm	
Poles	**8**		
Encoder PPR	**2000**	pulse/revolution	
Wire-Saving Encoder?	**TRUE**	(TRUE / FALSE)	

To enter a value for each parameter, double click in column B on the same line as the parameter name. Descriptions of each parameter are provided below.

Rated Speed	The rated speed of the motor (in RPM).
L_phase	Motor per phase inductance (in Henry).
R_phase	Per phase resistance of the motor (in ohms).
Rated Amps	The rated current of the motor (in Amps rms).
No Load Current	The no load current of the motor (in Amps rms). Needed only for induction motor.
Inertia of Motor	Motor inertia (in Kg-m^2)
Torque Constant (Kt)	Motor torque constant (in Newton-Meter per Amps rms).
Voltage Constant (Ke)	Motor voltage constant (in line-to-neutral rms volts per thousand rpm).
Poles	The number of motor poles.
Encoder PPR	Number of encoder pulses per revolution.
Wire-Saving Encoder	Encoder type. Enter TRUE for a wire-saving encoder; otherwise FALSE.

Step 3. Enter Application Information
The application information section of the motor setup worksheet contains parameter settings that describe the requirements of your specific application.

"=================== Application Information ======================"					
"------------------------- General --------------------------"					
Max RPM			4500	rpm	
(Vdc_Nom) Nominal Vdc			310	Volts	
(OvLoad) Max pu motor current at rated speed			3	pu	
"-------------- Speed Regulator Tuning --------------"					
Speed Regulator BW			200	rad/sec	
Positive Speed Rate limit			1000	rpm/sec	
Negative Speed Rate limit			1000	rpm/sec	
Inertia of Load	(measured)		0	Kg-m2	
SpdLpRate			2	1 SpdLoop per this # of CurLoop	
"-------------------- Current Limits ---------------------"					
Motoring Limit			200	%	
Regen Limit			200	%	
"---------- Inverter Switching Frequency ------------"					
(fc) Pwm carrier freq			10	KHz	
Dead_Time			0.5	usec	
" ------------ Current Regulator Tuning --------------- "					
(Ireg_BW) Current Reg BW			2500	rad/sec	

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.

Max RPM
This is the maximum speed (in rpm) required for your application. When motor speed exceeds this value, the system will generate an Overspeed trip fault. You may need to increase this value to make SpdScl less than 65535

Nominal Vdc
Nominal DC bus voltage (in volts). For use with the IRMCS2011 design 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 pu motor current at rated speed
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 you're not sure how to set this parameter, start with a value of 10 rad/sec and raise it gradually as you tune the system.

Positive Speed Rate limit
This parameter defines the maximum changeable speed allowed for acceleration in second.

Negative Speed Rate limit
This parameter defines the maximum changeable speed allowed for deceleration in second.

Inertia of Load
Load inertia (in Kg-m^2). If load inertia is not specified in your design data, use your best estimate and adjust the value later when you fine-tune drive operation.

SpdLpRate	Number of current loops for each speed loop. For example, a value of 3 would process the speed loop on every third current loop. Range is 1 to 7.
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. Maximum hardware current should be considered.
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. Maximum hardware current should be considered.
Pwm carrier freq	PWM carrier frequency. 10 KHz is the default setting for the IRMCK201 "*.irc" files. The setting of this parameter is a tradeoff between current ripple, inverter loss and EMI noise.
Dead Time	PWM dead time to prevent IGBT shoot through.
Current Reg BW	Current regulator bandwidth (in rad/sec). Normally 1/10 of PWM carrier frequency is the maximum value.

Step 4. Enter Advanced Information (Platform Dependent)
The advanced information section of the motor setup worksheet contains parameter settings that are specific to the hardware platform. **If you are using the IRMCS2011 development platform without modification, you do not need to modify these settings.**

"============ Advance Information (Platform dependent) ==============="				
Note: Below values are fixed for IRMCS201 platform however can be changed for other platform				
(Clk) FPGA clock freq			**33.333**	MHz
Dc bus Scaling (Vdc_Scl)			**8.1875**	cts/volt
I_Torque	(I_Trq_Rated)		**4095**	cts for rated Amps
(Mod_Pk) - U_Alpha U_Beta max linear modulation			**2355**	cts
" --------- **Desired Speed feedback Scaling** ---------"				
(Spd_Scale)			**16384**	cts/(Max RPM)
" ------------- **Current Feedback Scaling** ------------"				
Current Shunt Resistor			**10**	mOhm

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.

IRMCK201 clock freq	The frequency of the system clock, in MHz.
Dc bus Scaling	Scaling from internal count to voltage.
I_Torque	4095 indicates 100% rated amps.
U_Alpha U_Beta max linear modulation	Maximum linear range of Space Vector PWM

SpdScale Maximum speed is mapped to this value internally.

Current Shunt Resistor Shunt resistor value for current feedback.

Step 4. Enter Commutation Information
This section of the motor setup worksheet contains parameter settings used in determining the starting position of the motor.

"================== Commutation Information ===================="		
Angle of Z-pulse (based on UV line to line voltage)	272	degree
Mid Angle when Hall CBA is 001	120	degree
Mid Angle when Hall CBA is 010	240	degree
Mid Angle when Hall CBA is 011	180	degree
Mid Angle when Hall CBA is 100	0	degree
Mid Angle when Hall CBA is 101	60	degree
Mid Angle when Hall CBA is 110	300	degree

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.

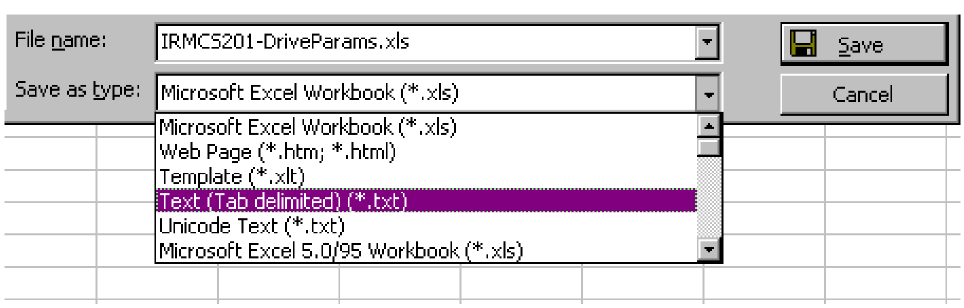
Angle of Z-pulse Enter the angle (in degrees) that z-pulse occurs in the reference of UV line-to-line voltage.

Mid Angle when Hall CBA is *cba*
For these six parameters, enter the angle (in degrees) that corresponds to each combination of Hall sensor values (*cba*).

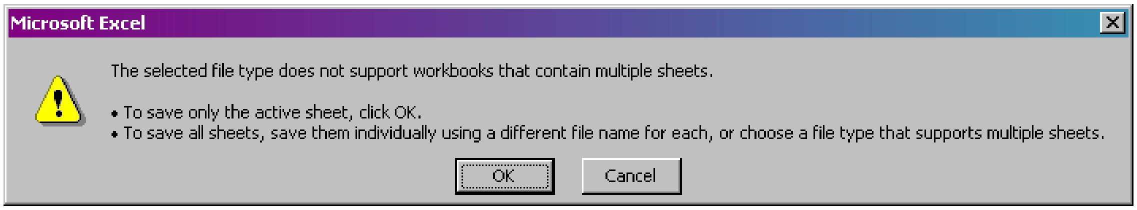
<u>Export Drive Parameters in Excel</u>
In the second stage of configuring drive parameters, the parameter settings you selected in the previous section are used to calculate values for a number of IRMCK201 write registers. The write register values and motor configuration parameters are written to a text file in a specific format defined for use with ServoDesigner.

Step 1. Save Your Settings
When you are satisfied with your settings, select Save from Excel’s File menu to save your workbook file in “.xls” format.

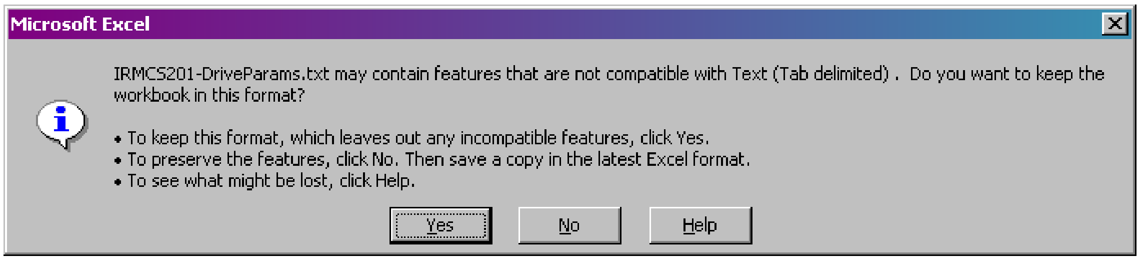
Step 2. Export Drive Parameters
Click on the “IRMCx201_Parameters” tab at the bottom of the workbook window. This worksheet shows the register values that are calculated based on the settings in the Calculation worksheet. From Excel’s File menu, select “Save As…”. In the Save As dialog, select Save as type: “Text (Tab delimited) (*.txt)” as shown below. Then browse to a folder where you want to save the exported drive parameters file, specify a file name, and click Save.



Click OK when you see the following warning message:



Click Yes when you see a warning message like this:



Import Drive Parameters in ServoDesigner

The final stage of drive parameter configuration involves loading the drive parameter settings into your ServoDesigner database and writing the registers to the IRMCK201 IC. 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 you 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, you can view and modify these settings by selecting Motor Configuration from the Preferences menu. When you import the drive parameters text file 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 your 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 IRMCK201 register when you double click the register entry.
- Also in the register definition, the EEPROM Value to Write can be saved to EEPROM and used to initialize the IRMCK201 register on power up.
- In the Function Definitions section, one or more functions may write the register value to the IRMCK201. (A function is set up to perform a sequence of operations automatically.)

When you import the drive parameters text file into ServoDesigner, you have 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. (If you haven't already created a configuration file to use with your project, 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 you 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 mode you choose, ServoDesigner may prompt you 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 you currently have open. If you want to save the new settings in your configuration file, you must select Save from the File menu before exiting ServoDesigner. If you don't do this, the updates will be lost, and you'll have to repeat the Import Drive Parameters function next time you open your configuration file.

Step 4. Write the Settings to the IRMCK201
The Import Drive Parameters function does not write any values to the IRMCK201 IC; it simply updates the register settings in your database. To transfer the register settings to the IRMCK201, you must 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.

3.2 New Motor Adaptation for a Permanent Magnet Motor

This section is based on IRMCS2011 hardware. It provides a step-by-step procedure for determining the correct configuration parameters for a new permanent magnet motor, and describes how to store the parameters in the ServoDesigner tool. If you use the Excel workbook file as described in the "Drive Parameter Setup" section above, a portion of this procedure is automated and you can skip certain steps where noted.

A similar procedure for an induction motor is provided in Section 3.3.

3.2.1 Step 1 – System Setup

In this step, you need to prepare your motor, cables, computer and IRMCS2011.

Step 1-1
Prepare IRMCS2011 version 2.0 or later.

Step 1-2
Make an encoder cable and a power cable for your motor.

- Assemble 15-pin male D-Sub connector, referring to Figure1.
- Eleven pins are used: A+ (pin2), A- (pin3), B+ (pin4), B- (pin5), Z+ (pin6), Z- (pin7), HALL_A (pin10), HALL_B (pin11), HALL_C (pin12), 5V(pin1 or pin9) and GND (pin8 or pin15).
- Make sure that the encoder is a 5V type. If it is not a 5V type, proper modification is required.
- If hall sensors have differential output, connect only positive sides and leave negative sides open.

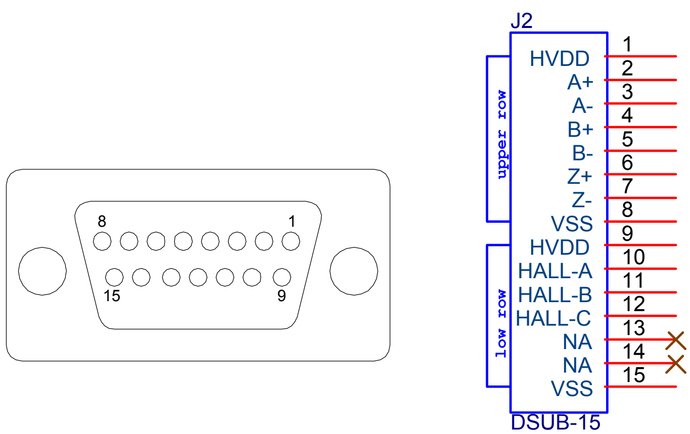


Figure 18. Encoder Interface Connector, J2

Step 1-3
Connect the encoder cable to J2 of IRMCS2011 hardware.

Step 1-4
Connect 3-phase AC lines to R/S/T terminals of IRMCS2011 hardware J1. The order doesn't matter. If single-phase AC lines are used, connect them to any two of R/S/T terminals

Step 1-5
Connect the motor power cable to U/V/W terminals of IRMCS2011 hardware J1. The order isn't important; however, use the order specified in your motor datasheet, if available.

Motor frame ground wire needs to be connected to the E terminal of J1 if available.

Step 1-6
Connect an RS232 serial cable from the computer to IRMCS201 hardware connector J6.

3.2.2 *Step 2 – Motor Sequencing and Encoder A/B Establishment*

In this step, motor U/V/W phase sequence and encoder A/B will be determined. You need to prepare a 2-channel oscilloscope and measure Encoder A and Encoder B signals (Step 2-14).

Step 2-1

Turn on the power and verify that the LED on the IRMCS2011 hardware is flashing red.

- If the LED stays off, check your AC line and cables.
- If the LED is not flashing red, your hardware has a problem. Contact IR customer support.

Step 2-2

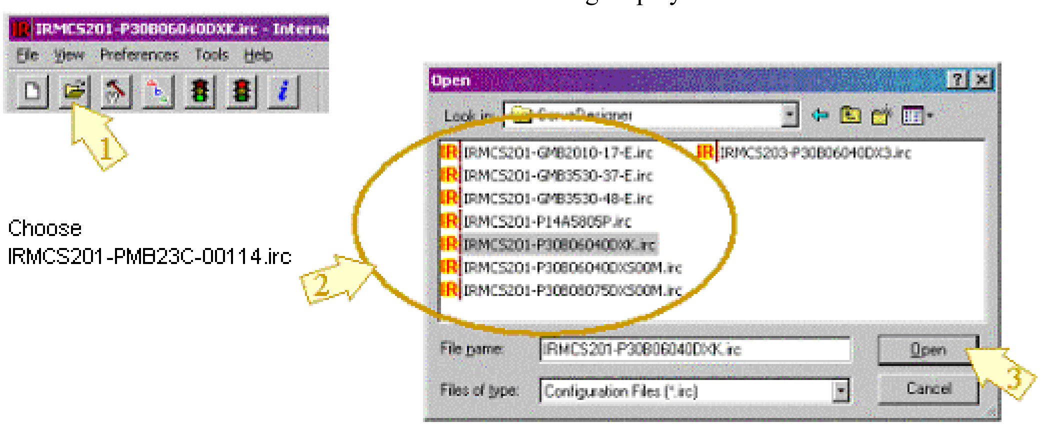
Close all unnecessary programs on your computer due to possible interaction with other applications.

Step 2-3

Execute 'ServoDesigner.exe' (Version 4.1.0.4 or later). Refer to the ServoDesigner User's Guide for questions about ServoDesigner if this is the first time you have used the tool. **You should have "Product ID and version checking" disabled and "Status polling" enabled!**

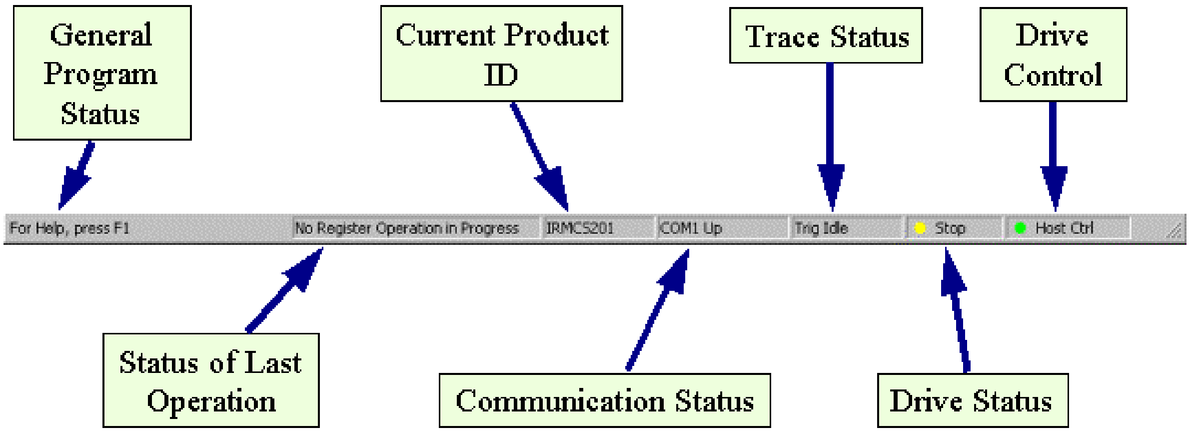
Step 2-4

Open 'IRMCK201-PMB23C-00114.irc' as shown in the following display.



Step 2-5

Maximize ServoDesigner's window and check the status display at the bottom of the window. It should show 'No product ID' (because "Product ID and version checking" is disabled), 'COMx Up', 'Trig Idle', a yellow light with 'Stop' and a green light with 'Host Ctrl'.

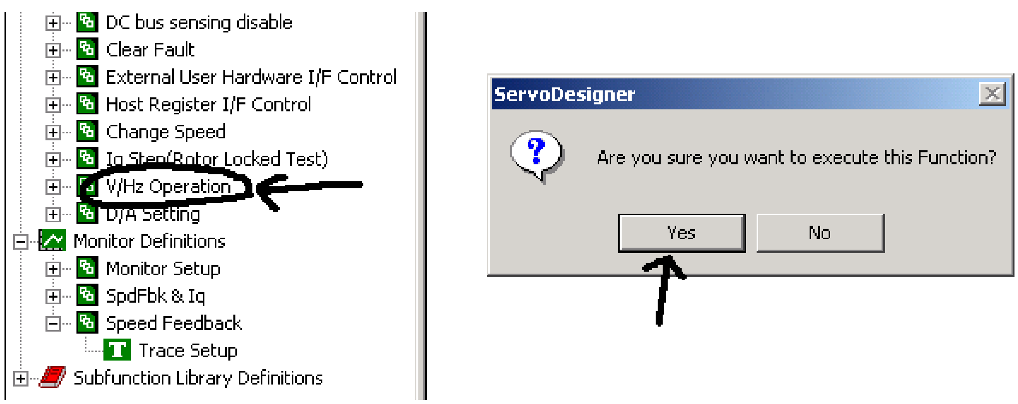


Step 2-6


Click the 'Configure Motor' button. The LED on the IRMCS2011 hardware should change to flashing green. If it doesn't change, check your hardware and serial cable connection.

Step 2-7

Expand 'Function Definitions', double click 'V/Hz Operation' and click on 'Yes'.



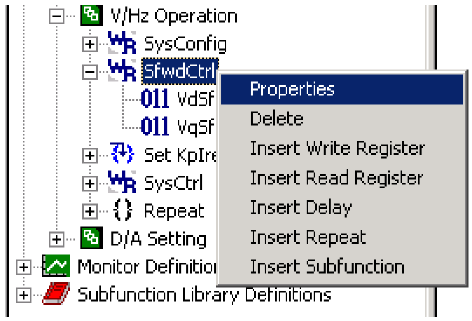
- If the motor shaft doesn't rotate and just vibrates, go to Step 2-8.
- If the motor shaft rotates in one direction, go to Step 2-11.
- If the motor shaft doesn't move at all, check the system setup.

Step 2-8

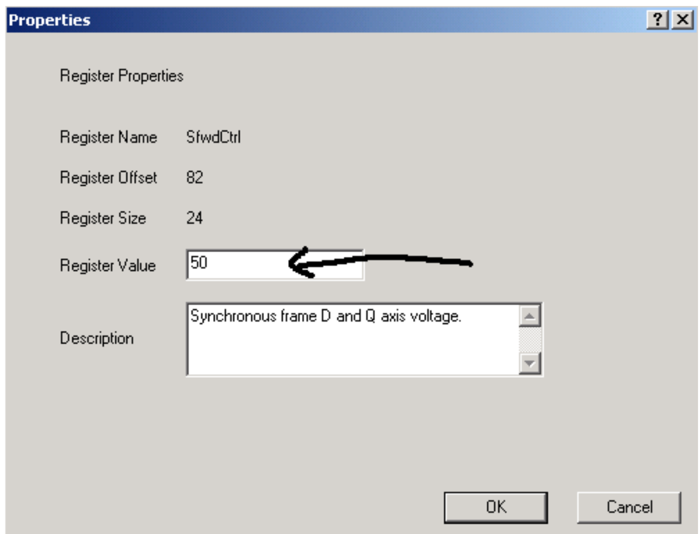

Click the 'Stop motor' button.

Step 2-9

Click '+' and, inside the 'V/Hz Operation' function, right click on 'SfwdCtrl'. Select Properties.



Step 2-10
Increase value in 'V/Hz Operation' by 10, click 'OK' and return to Step 2-7.



Step 2-11
Check the motor rotational direction at the motor front end (facing the motor front).

- If it's clockwise, go to Step 2-12.
- If it's counter clockwise, go to Step 2-14.

Step 2-12

Click 'Stop motor' button and turn off the power. Wait for a couple of minutes until the LED on IRMCS2011 hardware goes off. Then proceed to Step 2-13.

Step 2-13
Swap two wires of power cable by moving 'V' phase wire to 'W' of J1 and 'W' phase wire to 'V'.
Turn on the power and return to Step 2-5.

Step 2-14
Monitor encoder feedback signals by hooking up oscilloscope probes to test point "EncA" and "EncB". Probe ground should be connected to test point "VSS".

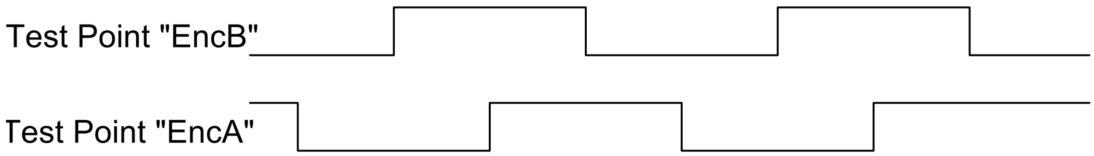


Figure 19. Sample Encoder A and B Signal Timing (B leads A)

Check if EncB leads EncA.

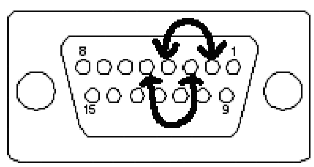
- If B leads A, go to Step 2-16.
- If A leads B, go to Step 2-15.

Step 2-15
Turn off the power and wait for a couple of minutes until the LED on IRMCS2011 hardware goes off.

Swap encoder A+ wire with encoder B+ wire and encoder A- wire with encoder B- wire in the connector going to J2. In other words, change the connection between pin2 and pin4, and between pin3 and pin5.

Close the trace window and minimize ServoDesigner.

Turn on the power and return to Step 2-5.



Step 2-16
Turn off the power and wait for a couple of minutes until the LED on IRMCS2011 hardware goes off.

Put UVW labels on UVW wires respectively and remove UVW wires and E wire, if connected, from terminal block J1 on the IRMCS2011 hardware.

Now you're ready to go to Step3: Hall_ABC and Z_pulse Measurement.

3.2.3 Step 3 – Hall_ABC and Z-pulse Measurement

In this step, you need to prepare a four-channel oscilloscope and measure UV line-to-line counter EMF voltage with respect to Hall_ABC and Z_pulse.

Step 3-1
Set up your oscilloscope in order to measure Hall_ABC and U-V line-to-line counter EMF voltage.

- Set volt/div of all four channels to 10V/div.
- Set time/div to 10 msec/div
- Hook up channel 1 probe to the motor 'U' wire and channel 1 ground to the motor 'V' wire.
- Hook up channel 2 probe to the test point 'HallC' and channel 2 ground to the test point 'VSS'.
- Hook up channel 3 probe to the test point 'HallB' and channel 3 ground to the test point 'VSS'.
- Hook up channel 4 probe to the test point 'HallA' and channel 4 ground to the test point 'VSS'.
- Set trigger source to channel 1, level to 10V, mode to normal, coupling to DC and slope to positive.

Step 3-2
Make sure the encoder cable is plugged in and the motor U/V/W are isolated one from another.

Turn on the power and rotate the motor shaft counter clockwise at the motor front end (facing the motor front) while monitoring channel 1. If it doesn't trigger, turn the shaft faster. Adjust trigger level, delay and time/div so that you can get one entire cycle of sinusoidal waveform as shown in Figure 20.

Once you get a good waveform, save this waveform.

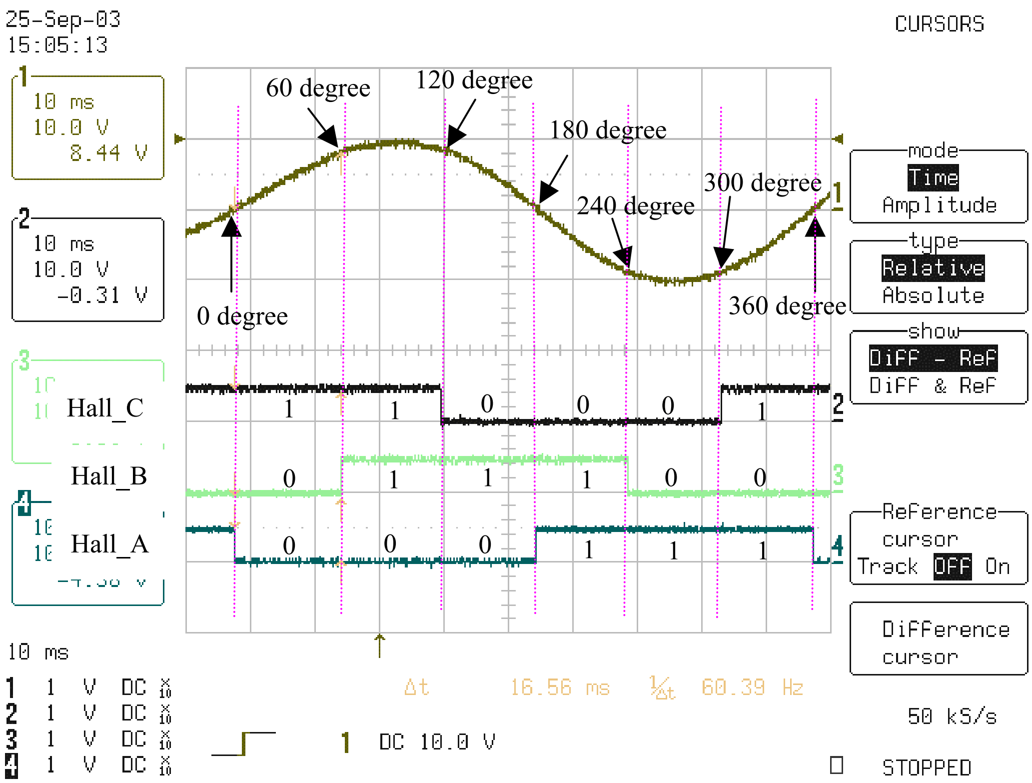


Figure 20. Sample Waveform of U-V counter EMF, Hall_C, Hall_B and Hall_A

Step 3-3
Turn off the power and set up your oscilloscope in order to measure the encoder Z-pulse and U-V line-to-line counter EMF voltage.

- Move channel 2 probe to the test point "EncZ".
- Set volt/div of channel 2 to 5V/div.
- Set trigger source to channel 2, trigger level to 2V and trigger mode to normal.
- Turn off channel 3 and channel 4.

Step 3-4
Turn on the power and rotate the motor shaft counter clockwise at the motor front end (facing the motor front) while monitoring channel 1. Adjust trigger level, delay and time/div so that you can get one entire cycle of sinusoidal waveform as shown in Figure 21.

Once you get a good waveform, save this waveform, too.

Turn off the power and remove scope probes.

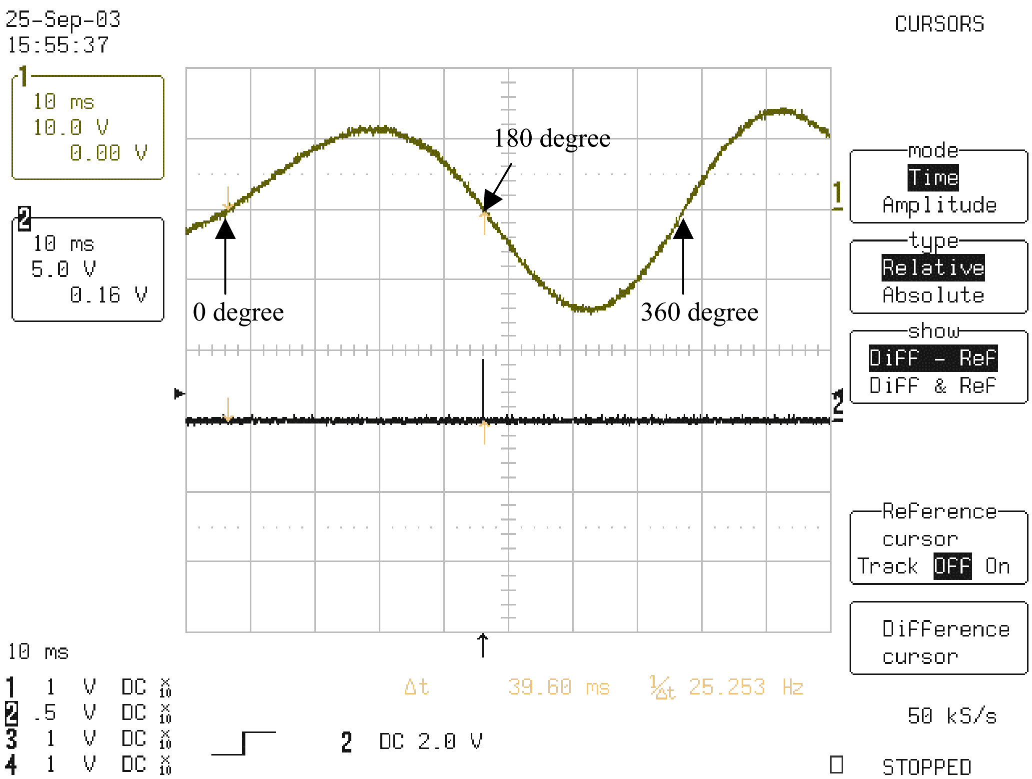


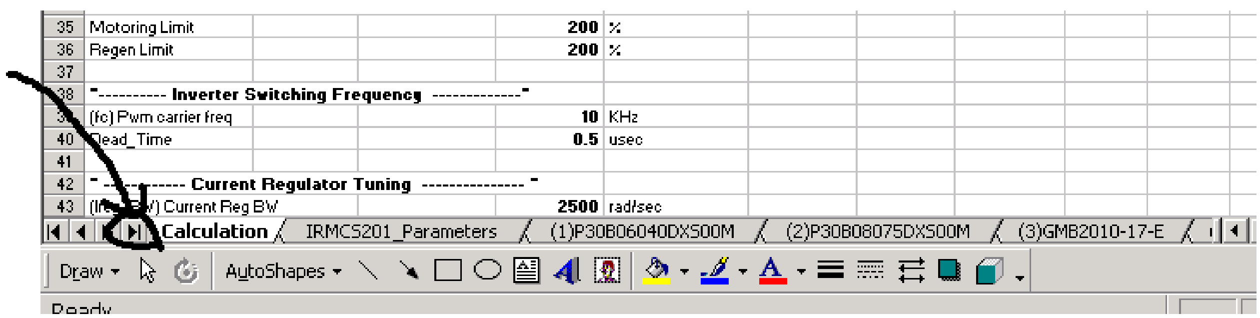
Figure 21. Sample Waveform of U-V counter EMF and Encoder Z_pulse

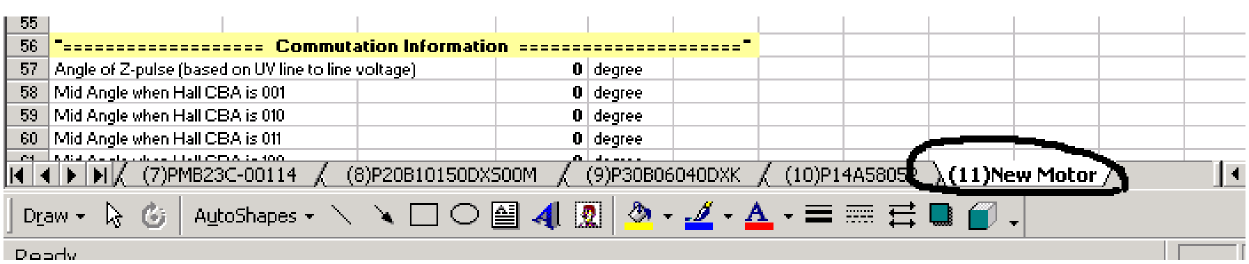
3.2.4 Step 4 – Entering Motor Parameters to Spreadsheet

In this step, you need to enter all the information you have about your motor and generate a text file to import parameters into ServoDesigner.

Step 4-1

Open 'IRMCx201-DriveParams.xls' and select '(11) New Motor' Worksheet.





Step 4-2
Enter all the parameters. Refer to Section 3.1 for more information.

If you're not sure how to set a parameter value, select the motor that is most similar to yours from the provided worksheets and copy from it.

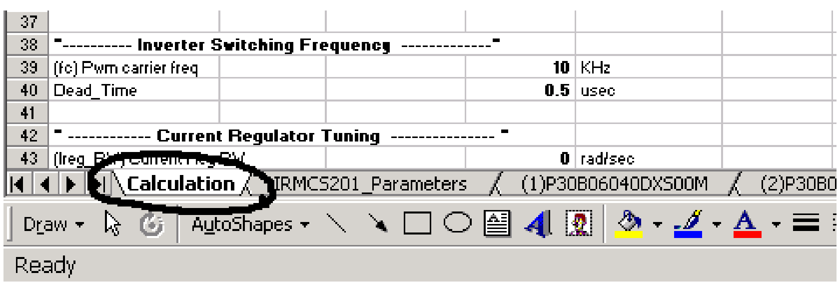
Enter 'Commutation Information' based on the waveforms you saved at steps 3-2 and 3-4. **Zero angle is set as the point when U-V line-to-line counter EMF voltage crossrs zero from negative to positive.** Hall CBA can change at each 30-degree segment. Any transition from Hall sensors occurs only at multiples of 30 degrees. Choose from 0, 30, 60, 90, 120, 150, 180, 210, 240, 270, 300, and 330. Each HallCBA has a 60-degree span. '000' and '111' don't exist.

- In the example shown in Figure 21, it's obvious that Z-pulse occurs at 180 degrees. Type 180 to 'Angle of Z-pulse'.
- In the example shown in Figure 20, when the angle is 0-60 degrees, Hall CBA is 100. Type the mid angle, which is 30 degrees for this example, into the 'Mid Angle when Hall CBA is 100' field. Fill out the rest of the fields.

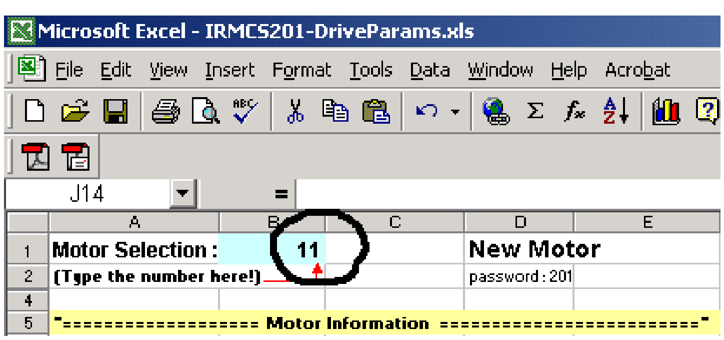
56	"=================== Commutation Information ====================="		
57	Angle of Z-pulse (based on UV line to line voltage)	180	degree
58	Mid Angle when Hall CBA is 001	270	degree
59	Mid Angle when Hall CBA is 010	150	degree
60	Mid Angle when Hall CBA is 011	210	degree
61	Mid Angle when Hall CBA is 100	30	degree
62	Mid Angle when Hall CBA is 101	330	degree
63	Mid Angle when Hall CBA is 110	90	degree

Step 4-3

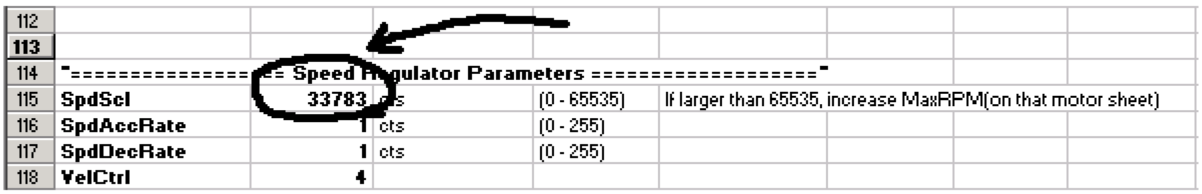
- Go to the 'Calculation' sheet.



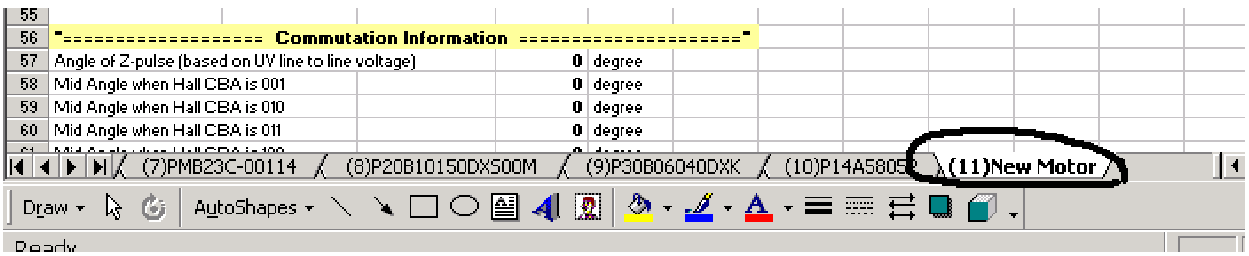
- Enter 11 as the Motor Selection.



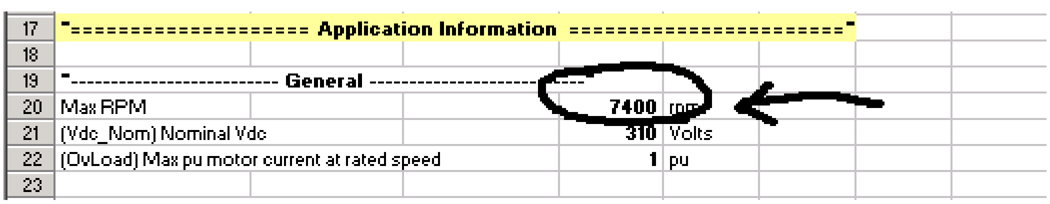
Step 4-4
Check the 'SpdScl' value.
- If it's larger than 65535, go to Step 4-5.
- If it's less than 65535, go to Step 4-6.



Step 4-5
- Go to '(11) New Motor' sheet.

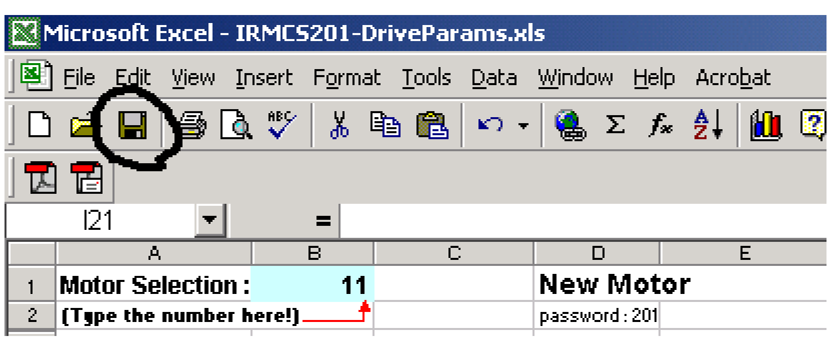


- Increase Max RPM by 1000.

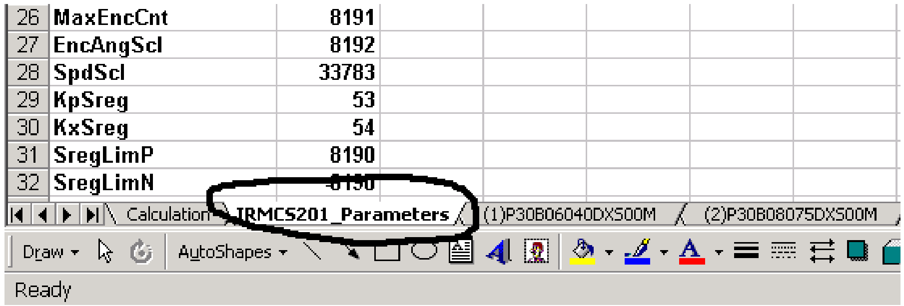


- Return to Step 4-3

Step 4-6
- Save the Excel spreadsheet file.

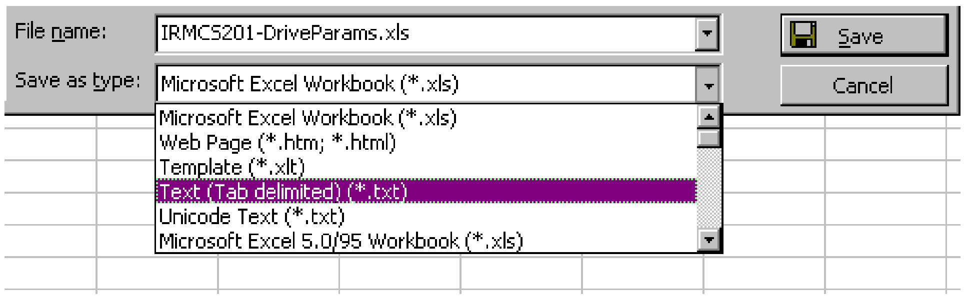


- Go to 'IRMCx201_Parameters' sheet.

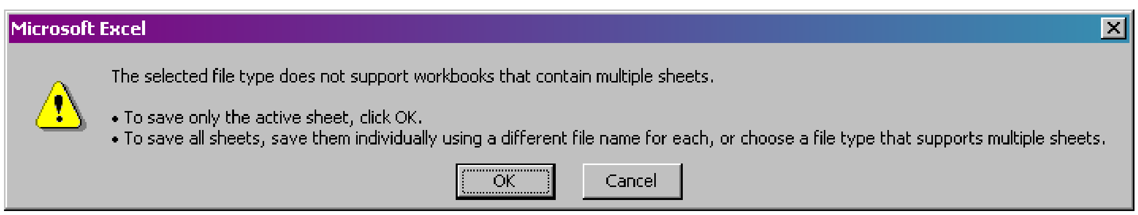


Step 4-7

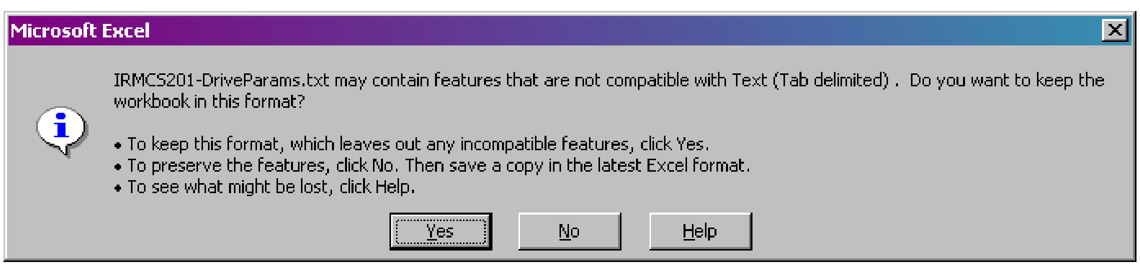
- Export parameters as text format. From Excel's File menu, select "Save As…". In the Save As dialog, select Save as type: "Text (Tab delimited) (*.txt)" as shown below. Then browse to a folder where you want to save the exported drive parameters file, specify a file name, and click Save.



- Click 'OK' when you see the following warning message.



- Click 'Yes' when you see the following warning message.



- Close Excel Window and click 'No' when you see the following warning message.

3.2.5 Step 5 – Import Parameters to ServoDesigner and Run the Motor

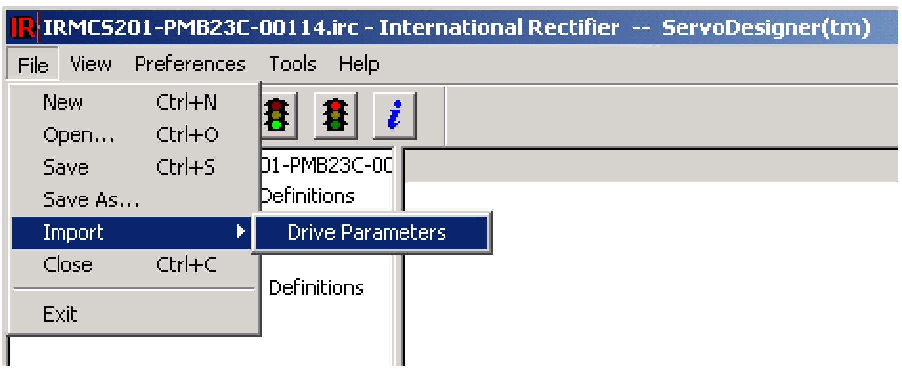
In this step, you import the text file to ServoDesigner and run the motor.

Step 5-1

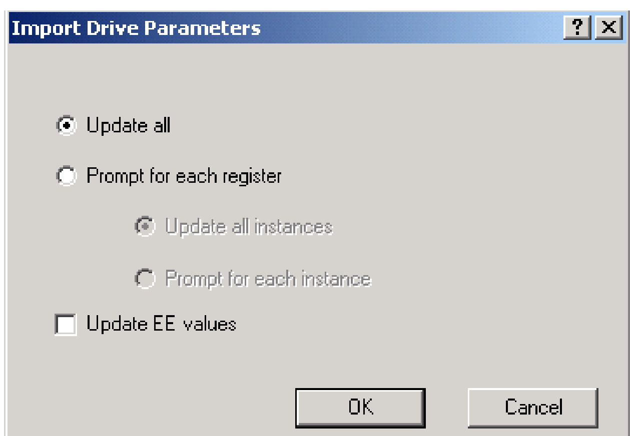
If you're following this procedure from Step 1, ServoDesigner was already executed in Step 2 and 'IRMCK201-PMB23C-00114.irc' is opened. If not, please run ServoDesigner and open 'IRMCK201-PMB23C-00114.irc'. Maximize the ServoDesigner window.

Step 5-2

- From the File menu, select Import, and from the Import sub-menu, select Drive Parameters.



- Browse to locate the text file you exported from Excel and click Open to open it.
- In the Import Drive Parameters dialog, select 'Update all' for the simplest operation. If you choose another mode, ServoDesigner will prompt you 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.



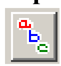
Step 5-3

- Connect your motor U/V/W/E wires back to U/V/W/E terminal of J1 on the IRMCS2011 hardware.
- Make sure the system setup is OK and all probes are removed.
- Turn on the power and check the LED and serial communication as we did in Step 2-1 and 2-5.

Step 5-4


Click the 'Configure Motor' button.

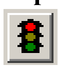
Step 5-5


Click the 'Starting Angle' button.

Step 5-6


Click the 'Start Motor' button.

Step 5-7


Click the 'Stop Motor' button.

Step 5-8

In Step 5-6, did the motor start rotating without clunking?

- If yes, go to Step 5-11.
- If no, go to Step 5-9.
- If the LED changes to static red, go to Step 5-10.

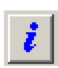
Step 5-9

Check your spreadsheet values.

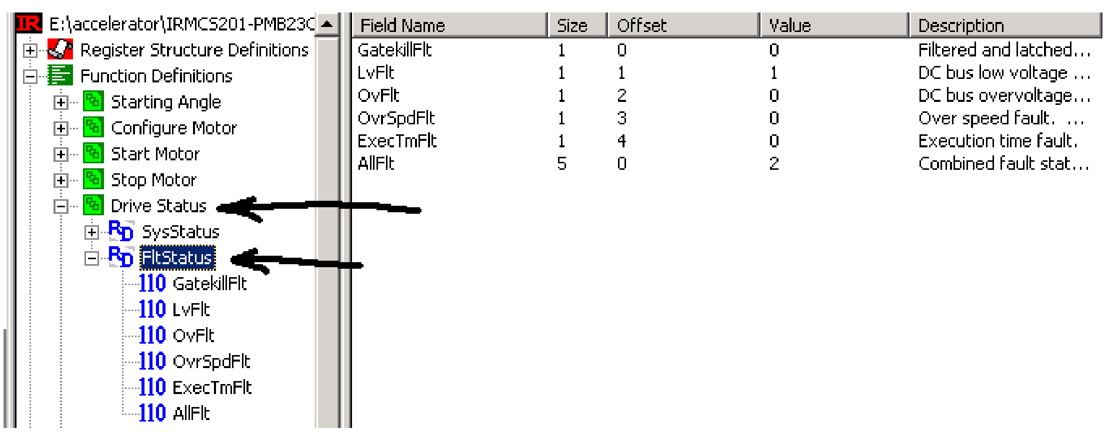
- In particular, verify that the Hall CBA and Z-pulse fields are correct.
- Verify that Encoder PPR and Number of Poles are correct.

Step 5-10

Execute this step if the LED changed to static red at Step 5-8 or 5-12.

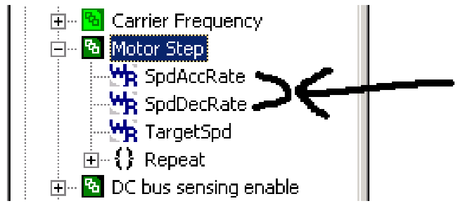

Click 'Drive Status'.

Expand the 'Drive Status' function by clicking '+'. Click on FltStatus. Look at the right hand side window to figure out which fault happened. Refer to the IRMCK201 datasheet for more information about fault conditions.



Step 5-11

Go to the 'Motor Step' function and change 'SpdAccRate' and 'SpdDecRate' to 127 respectively by right clicking and selecting Properties.



Step 5-12
Run the 'Motor Step' function by double clicking it.
Does the motor change speed from 0 rpm to a high speed?
- If yes, go to proceed to Step 5-13.
- If the LED turns to static red, check the drive status as described in Step 5-10. Depending on the fault, take appropriate action.

Step 5-13

Click 'Stop Motor' button.

Step 5-14
Use ServoDesigner's "Save As…" option to save your irc file with a different name.

Now, you're ready to play withwww your motor and the IRMCS2011!

3.3 New Motor Adaptation for an Induction Motor

This section provides a step-by-step procedure for determining the correct configuration parameters for a new induction machine adaptation, and describes how to store the parameters in the ServoDesigner tool. If you use the Excel workbook file as described in the "Drive Parameter Setup" section above, a portion of this procedure is automated and you can skip certain steps where noted.

A similar procedure for a permanent magnet motor is provided in Section 3.2.

An induction motor can be used in place of a permanent magnet motor based on the following criteria.

1) Must be 230V 3 phase machine
2) Must be equal to or less than 1 kW or 6 Arms continuous current rating.
3) Must have an encoder at the back end of the motor, and be incremental A quadrature B type with 5V differential output. The maximum encoder line count is 5000 ppr.

3.3.1 Step 1 - Encoder Connector Assembly

- Assemble 15-pin male D-Sub connector, referring to Figure22 .
- Only six pins are used, because z-pulse is not necessary for an induction machine. The six pins are: A+ (pin2), A- (pin3), B+ (pin4), B- (pin5), 5V(pin1 or pin9) and GND (pin8 or pin15).
- Make sure that the encoder is a 5V type. If it is not a 5V type, proper modification is required.
- Disable z-pulse by connecting Z+ to GND and Z- to 5V.

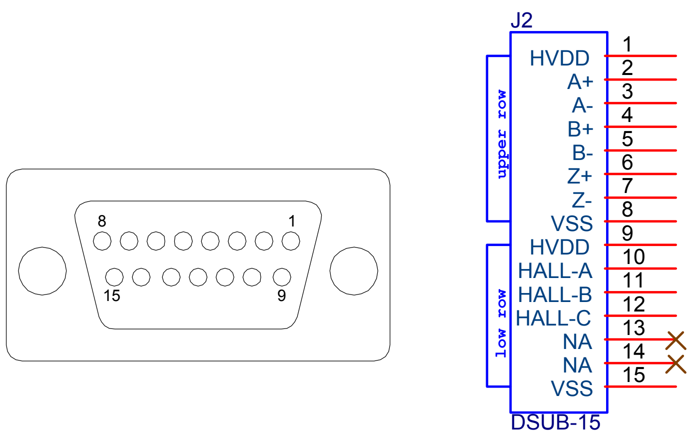


Figure 22. Encoder Interface Connector, J2

3.3.2 Step 2 - Encoder A/B Quadrature Polarity Establishment

- **Do not** connect motor U/V/W leads to Terminal Block J1 yet.
- Plug the encoder connector into J2.
- Rotate the motor shaft by hand in a counter-clockwise direction at the motor front end (facing the motor shaft).
- Using an oscilloscope, monitor the encoder A and B signals at test points 'EncA' and 'EncB' respectively.
- Verify that the B signal leads the A signal as shown in Figure 23.
- If A signal leads B signal, swap A+ with B+ and A- with B- by soldering wires on the connector pins.

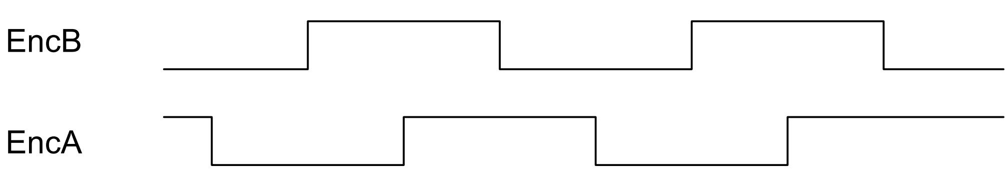


Figure 23. Sample Encoder A and B Signal Timing

3.3.3 Step 3 Motor Phase sequencing

- Unplug the encoder connector first.
- Connect the motor leads to U/V/W pins of terminal block J1.
- Run the ServoDesigner tool. (For detailed instructions, refer to the document "ServoDesigner User's Guide".)
- Open the configuration file "IRMCK201-P14A5805P.irc".
- Select Save As… from the File menu to create a new file using the name of the motor under test (your new motor).
- Execute the Configure Motor function by clicking the toolbar button (hammer and wrench icon) or double clicking the Configuration Motor function in the tree view (as shown in Figure 24).
- Execute the V/Hz Operation function by double clicking the name of the function in the tree view. The function will cause the motor shaft to turn. (The shaft may not turn if load is coupled. This step must be executed in a no load condition.)

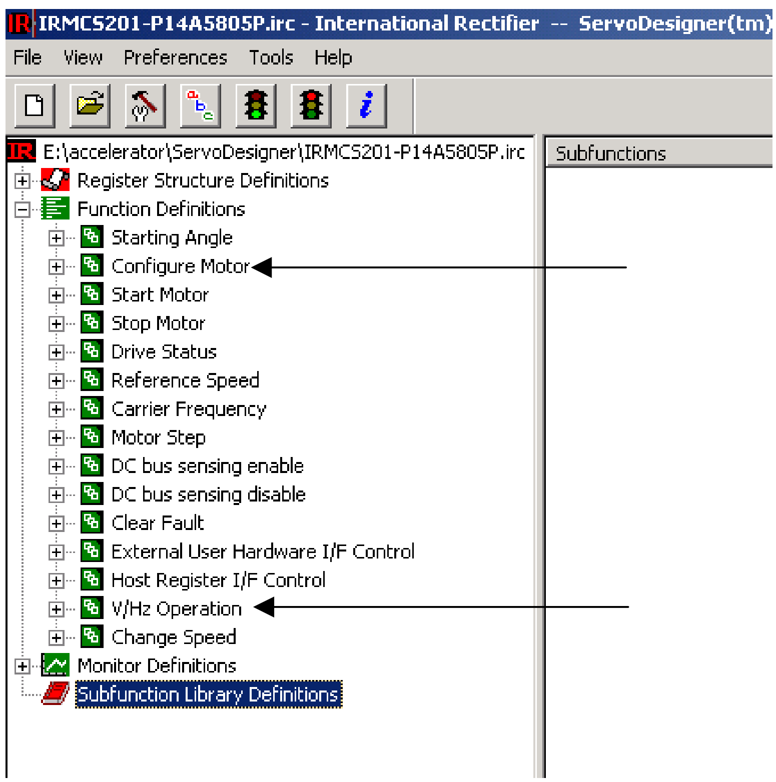


Figure 24. ServoDesigner Functions

- Verify the rotation direction. If it's counter-clockwise, go to Step 4.
- If the direction of rotation is clockwise, swap V with W and repeat Step 3.

3.3.4 Step 4 – Current Feedback Scaling

Id scale factor and **Iq scale factor** need to be separately adjusted for an induction motor application. These two scaling factors are derived by different equations and therefore are usually different values. The IRMCx201 design is based on the mapping of rated current (continuous current) to be ±4095 digital values regardless of the motor current rating. The current control therefore regulates ±4095 as 100% rated torque for torque producing current, "Qi". If the motor has 300% overload capability, then the control regulates a maximum of ±12,287. The field component current, "Di" is also regulated 100% field flux with 4095 digital value regardless of the motor physical current or amount of magnetizing current.

If you use the Excel workbook, the Id and Iq scale factors are calculated for you. The following paragraphs describe how these values are determined.

In order to find Id scale factor and Iq scale factor, both torque producing current and magnetizing current need to be determined at a rated condition.

Motor nameplate data typically shows the rated current and either no load current or magnetizing current. Torque producing current can be derived from the following equation (all units are in rms):

$$Torque\ producing\ Current\,[TPC] = \sqrt{(rated\ current)^2 - (no\ load\ current[NLC])^2}$$

Then, iq scale factor can be derived from the equation:

$$Iq\ scale\ factor = \frac{MHC \times 4095 \times 1024}{TPC \times 2047 \times \sqrt{2} \times 1.647 \times .82}$$

Similarly, id scale factor can also be derived from the equation:

$$Id\ scale\ factor = \frac{MHC \times 4095 \times 1024}{NLC \times 2047 \times \sqrt{2} \times 1.647 \times .82}$$

where .82 is maximum modulation index of IR2175 and 1.647 is gain scale of the vector rotator in the current feedback path.

Example:

Motor rated current (RC)	±6.0 Arms / ±8.46 Apeak
Motor overload current:	±9.0 Arms / ±12.7 Apeak
No load current (NLC):	±3.8 Arms / ±5.37 Apeak
Maximum Hardware Current:	±13 Apeak

TPC = sqrt(6^2 – 3.8^2) = sqrt(21.56) = 4.64 Arms
Iq scale factor = 13 x 4095 x 1024 / (4.64 x 2047 x 1.414 x 1.647 x .82) = 3004
Id scale factor = 13 x 4095 x 1024 / (3.8 x 2047 x 1.414 x 1.647 x .82) = 3669

3.3.5 Step 5 – Slip Gain adjustment, Current controller PI gain adjustment

The slip gain needs to be adjusted for induction machine application. Figure 25 shows the slip gain block diagram.

If you use the Excel workbook, the slip gain is calculated for you. The calculation requires the rated RPM information on the motor nameplate data and the IRMCx201 closed loop current control update rate information.

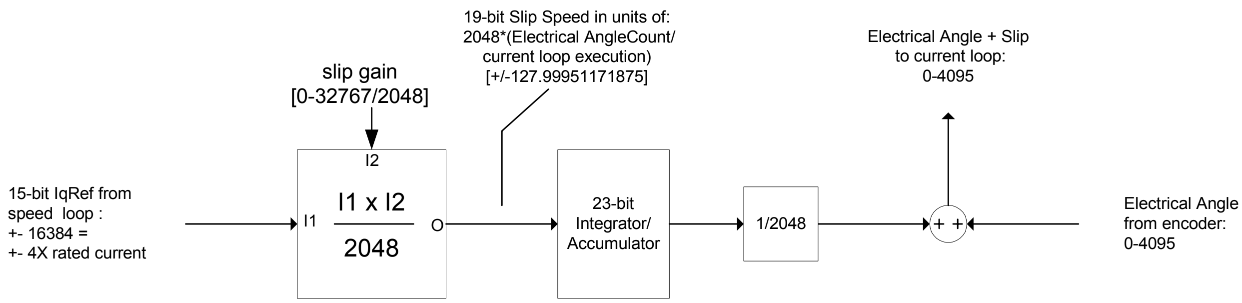


Figure 25. Slip Gain Block Diagram

The following equation is used to derive the slip gain:

$$Rated\ Hz = \frac{\dfrac{120 \times rated\ frequency}{polenumber} - rated\ RPM}{\dfrac{120 \times rated\ frequency}{pole\,number}} \times rated\ frequency$$

$$Internal\ slip\ speed = 2048 \times \frac{Rated\ Hz \times 4096}{current\ loop\ update\ rate}$$

$$Slip\ gain = \frac{2048 \times Internal\ slip\ speed}{4096}$$

Example:

Rated RPM:	1725 rpm
Rated frequency:	60 Hz
Pole number:	4
Current loop update rate:	10 kHz

Rated Hz = ((120 * 60 / 4) – 1725) *60 / (120 * 60 / 4) = 2.5Hz
Internal slip speed = 2048 * 2.5 * 4096 / 10000 = 2097
Slip gain = 2097 * 2048 / 4096 = 1048

3.3.6 Step 6 – Encoder Configuration

If you are using the Excel workbook, you can skip the remaining steps (steps 6 – 8). Follow the instructions in the "Drive Parameter Setup" section instead.

In ServoDesigner, modify the maximum encoder counter value in the Configure Motor function. Expand the Configure Motor function in the tree view and locate the MaxEncCnt register entry. Right click on MaxEncCnt and select Properties. In the Register Value field of the Properties dialog, enter the maximum encoder counter value as follows:

MaxEncCnt = EncoderPPR * 4 – 1

Modify the encoder angle scale factor in the Configure Motor function. Right click on the EncAngScl register entry in the Configure Motor function and select Properties. In the Register Value field of the Properties dialog, enter the encoder angle scale factor value as follows:

EncAngScl = Number of Poles / 2 * 4096 * 4096 / (EncoderPPR * 4)

3.3.7 Step 7 – Speed Feedback Scaling, Speed controller PI gain adjustment

In ServoDesigner, modify the speed scale factor in the Configure Motor function. Expand the Configure Motor function in the tree view and locate the SpdScl register entry. Right click on SpdScl and select Properties. In the Register Value field of the Properties dialog, enter the speed scale factor value as follows:

SpdScl = 60 * 16383 * 33.333/32 *10^6 / (EncoderPPR * MaxRPM)

3.3.8 Step 8 – Save Values in ServoDesigner

In ServoDesigner, select Save from the File menu to save your changes to the Configure Motor function.

3.4 Speed-Controlled Servo Motor Initialization and Operation

This section describes the register-controlled configuration and operation for an example system that uses an IRMCK201 IC with a host microprocessor, IRAM intelligent IGBT module, IR2175 current sense IC, and a Sanyo-Denki P30B06040DXS00M servomotor. The system is illustrated in Figure 26.

The IRMCK201 IC interfaces with a host microprocessor through a set of host registers, which are divided into write (control) and read (status) registers. The host microprocessor configures the IRMCK201 host write registers at power-on, and then uses the appropriate host write and read register fields for motor start, stop, and fault processing. For this example, we have assumed that the user is using the IRMCK201 internal torque loop and speed loop to control motor speed.

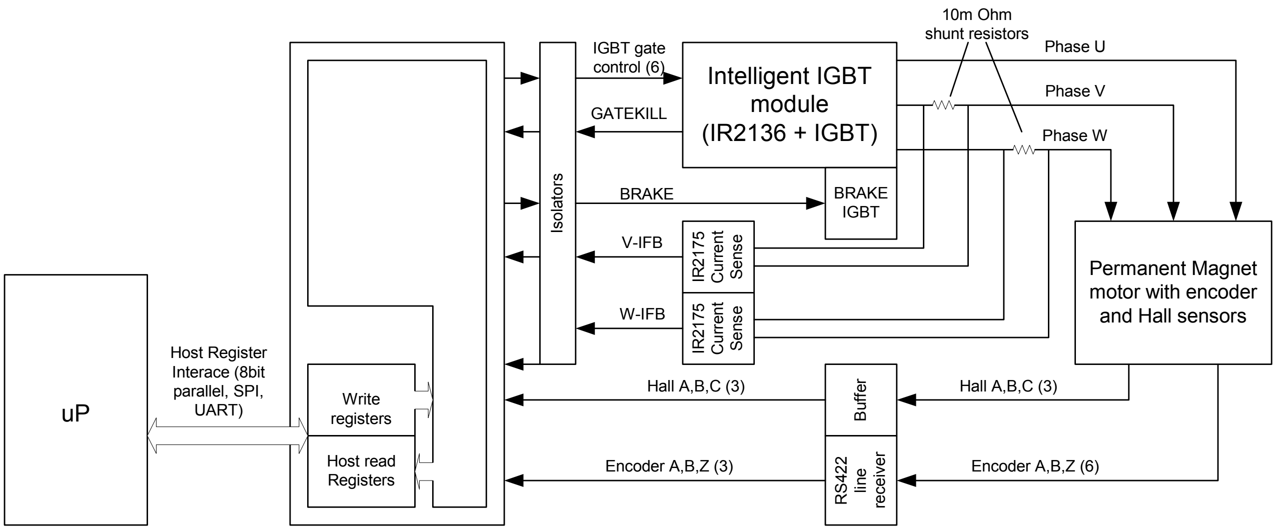


Figure 26. Host Microprocessor Controlled IRMCK201 System

3.4.1 Initialization

Table 10 gives IRMCK201 Motion Control Processor host write register initialization values for this example. These registers must be initialized at power-on before starting the motor. A brief description is provided for each field. Where noted, more detailed information is provided in the paragraphs following the table.

Field Name	Reset Value	Initial Value	Comment
Quadrature Decode Register Group			
EncCntW	0	-	Encoder Count Write Register. See description below for Hall Sensor initialization procedure.
MaxEncCnt	0	0x1F3F 7999	Maximum Encoder Count Value.[1]
ZEncCnt	0	0x158 (344)	Z-pulse Encoder Count Value.[1]
EncAngScl	0	8388 0x20C4	Encoder Angle Scale Factor.[1]
PwrOnRedSig	0	0	Power-on Reduced Signal Enable, only used for EEPROM standalone implementations.
ZpulsePol	0	1	Encoder Z-pulse polarity.

Field Name	Reset Value	Initial Value	Comment
ZpulseEnb	0	1	Enable Z-pulse load from ZencCnt.
CntEnb	0	1	Enable encoder counter.
PWM Configuration Register Group			
GatekillSns	0	0	IR213x Gatekill signal sense, 0 = active low.
GateSnsL	0	0	IR213x low side IGBT gate control sense, 0 = active low.
GateSnsU	0	0	IR213x high side IGBT gate control sense, 0 = active low.
SD	0	1	IR213x shutdown control (active low).
PwmPeriod	0	1666 0x682	Divider for 10kHz PWM.[1]
PwmDeadTm	0	17 0x11	Divider for 17 * .030 = .51 usec.
PwmConfig	0	1	Symmetrical, center-aligned PWM.
Current Feedback Configuration Register Group			
IdScl	0	4925 0x133D	Rotating frame "D" axis feedback current scale factor.[1]
IqScl	0	4925 0x133D	Rotating frame "Q" axis feedback current scale factor.[1]
System Control Register Group			
DcCompEnb	0	1	DC bus compensation enable. Setting this bit causes the current regulator output voltage to be automatically compensated for DC-bus voltage fluctuations.
IfbOffsEnb	0	1	Automatic current feedback offset application enable[1]
FocEnbW	0	0	FOC closed-loop control enable[2]
PwmEnbW	0	0	PWM output enable[2]
Torque Loop Configuration Register Group			
IdRef	0	0	Direct/Magnetizing axis reference current to current-loop PI controller. Used with external user velocity loop.
IqRef	0	0	Quadrature axis reference current to current-loop PI controller. Used with external user velocity loop.
KpIreg	0		Current regulator proportional gain
KxIreg	0		Current regulator integral gain
SlipGn	0	0	Slip-gain for induction motor implementation
Vdlim			Current regulator D-axis output limit
Vqlim			Current regulator Q-axis output limit
Velocity Control Register Group			
KpSreg	0		Speed regulator proportional gain
KxSreg	0		Speed regulator integral gain
SregLimP	0		Speed regulator output positive limit
SregLimN	0		Speed regulator output negative limit
SpdScl	0		Speed calculation scale factor
TargetSpd	0		Target motor speed
SpdAccRate	0		Motor acceleration rate
SpdDecRate	0		Motor deceleration rate
Fault Control Register			
FltClr	0	0	Fault clear register[2]
DcBusMEnb	0	1	Dc bus voltage monitor enable
SVPWM Scaler Register			
ModScl	0	5020 0x139C	SVPWM modulation scale factor. Set this value to PwmPer*sqrt(3)*4096/2355

Field Name	Reset Value	Initial Value	Comment
Diagnostic PWM Control Register Group			
PwmData0Sel	0	0	Diagnostic DAC Pwm #0 Data Select[2]
PwmData1Sel	0	0	Diagnostic DAC Pwm #1 Data Select[2]
PwmData2Sel	0	0	Diagnostic DAC Pwm #2 Data Select[2]
PwmData3Sel	0	0	Diagnostic DAC Pwm #3 Data Select[2]
System Configuration Register Group			
RmpRefSel	0	0	Speed ramp reference select.
HostVdEnb	0	0	Direct host D-axis voltage control enable
HostAngEnb	0	0	Direct host DQ voltage axis angle control enable
IqRefSel	0	0	Current regulator reference input select
SpdRefSel	0	0	Speed regulator reference input select
ExtCtrl	0	0	External user interface control enable (no-host standalone mode)
Direct Host Voltage Control Register Group			
VdSfwd	0	0	Host-controlled D-axis voltage.
VqSfwd	0	0	Host-controlled Q-axis voltage.
ElecAngW	0	0	Host-controlled Electrical Angle.
32-bit Quadrature Decoder Register Group			
EncCnt32bW	0	0	32-bit Encoder count register, for position control
EEPROM Control Register Group			
EeWrite	0	0	
EeRead	0	0	
EeRst	0	0	
EeAddr/RegMapCode	0	0	
EeDataW	0	0	
Hall Sensor Encoder Initialization Group			
HallCBA001	0	0	
HallCBA010	0	0	
HallCBA011	0	0	
HallCBA100	0	0	
HallCBA101	0	0	
HallCBA110	0	0	

Table 10. IRMCK201 Write Register Initialization Values

Notes:
1. See description below for more information
2. These fields are also changed dynamically during motor operation. See description below for more details.

Encoder Maximum Count (MaxEncCnt)

When the encoder count reaches this value, the next count pulse resets the count register to 0. The encoder count tracks rotor angle, which is used for current loop d-q axis transformations. This register must be set to the count that corresponds to a 360-degree physical angle, which in this case is (2000 lines * 4) – 1 = 7999.

Encoder Z-pulse Count (ZEncCnt)

The encoder Z-pulse count specifies a count that is automatically loaded into the encoder count register each time the Z-pulse asserts. This action aligns the encoder count with the rotor angle so that the encoder can be used as an angle sensor. The Z-pulse value should be set such that the encoder count is 0 when the rotor angle is 0 relative to the stator U-phase MMF. For the 8-pole motor in this example, the Z-pulse occurs at an electrical angle of 62 degrees (mechanical angle of 15.5 degrees), so the ZEncCnt field should be set to 8000*15.5/360 = 344. Figure 27 shows the relationship between encoder count, Z-pulse count, physical angle, and electrical angle for this example. The Z-pulse count value can be determined either experimentally or from the motor manufacturer's data.

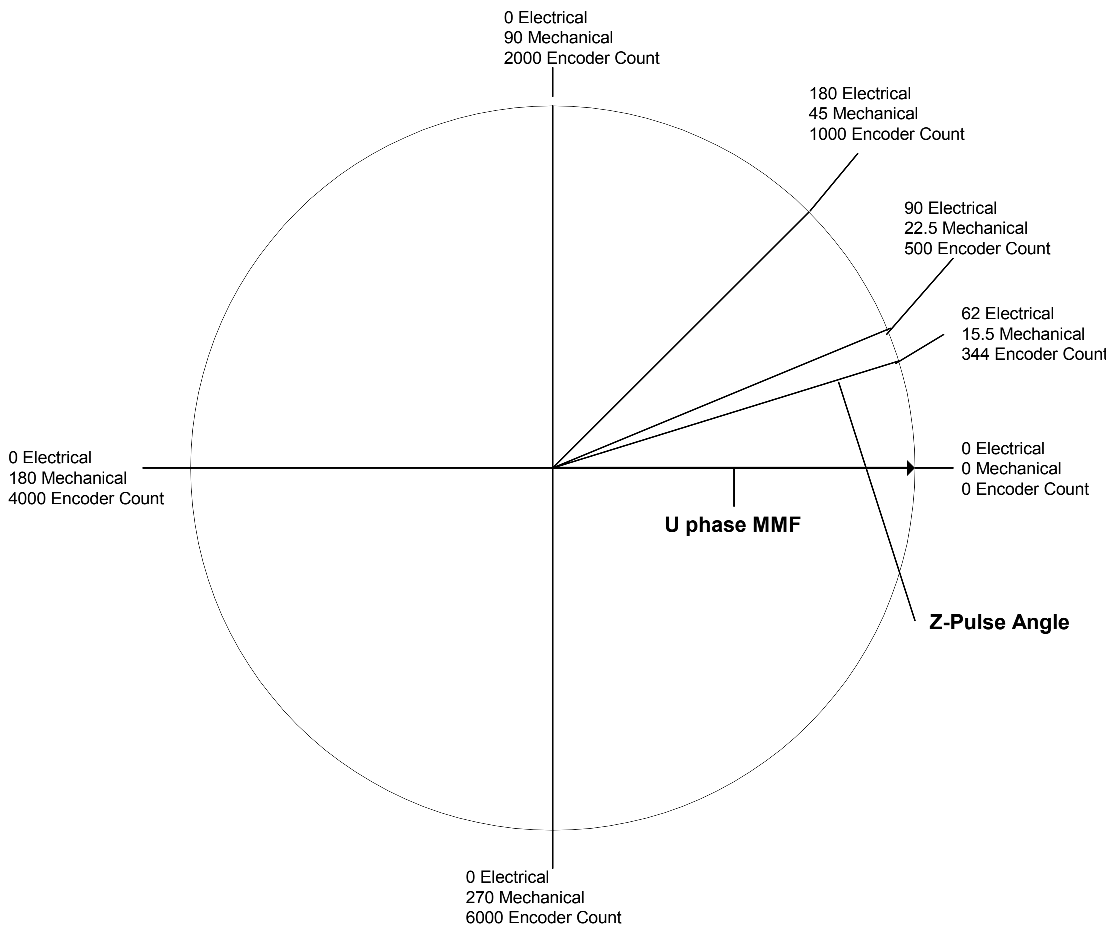


Figure 27. Motor Angle Relationships

Encoder Count Initialization via Hall Sensors (EncCntW)

At system power-up, the encoder count will not give the correct rotor angle representation until the first Z-pulse occurs. The host uses the Hall ABC sensors to approximate initial rotor position (to within 60 degrees) so that it can rotate the motor before the first Z-pulse. Figure 28 shows the Hall ABC values and the corresponding initial electrical angle and encoder count. During system initialization the host microprocessor must read the Hall ABC values from the Quadrature Decode Status Register Group Hall A,B,C register fields (PwrOnHall A,B,C fields for a wire-saving encoder) and use a lookup table to load the appropriate encoder count. The Hall ABC angle data can be determined either experimentally or from the motor manufacturer's data.

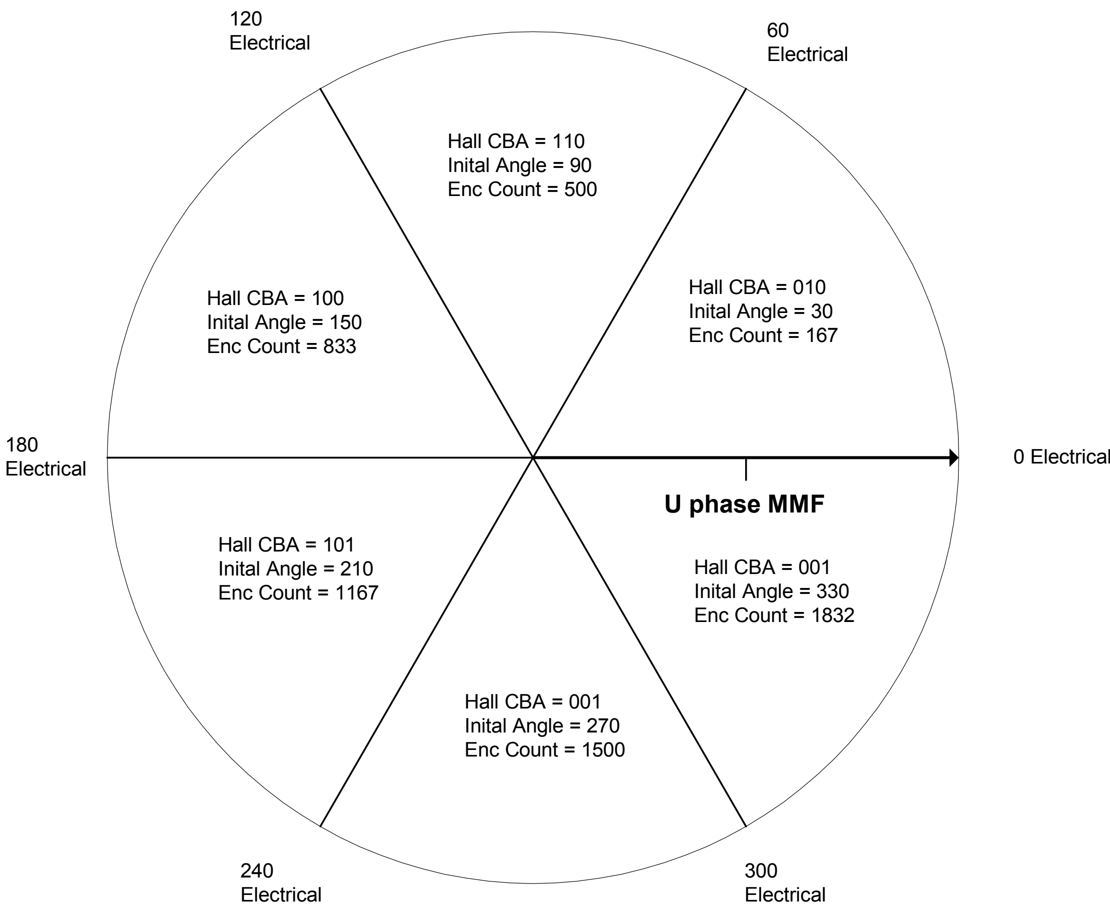


Figure 28. Hall Sensor and Initial Angle Values

Encoder Angle Scale Factor (EncAngScl)

The IRMCK201 FOC algorithm performs transformations to and from the rotating d-q frame using a 12-bit electrical angle with 0-4095 corresponding to angles from 0 – 359.9. The electrical angle is determined from the encoder count and the Encoder Angle Scale factor using the following equation:

ElecAng = ((MtrPoles / 2) * 4096 * (EncCnt) / (MaxEncCnt + 1)) MOD 4096

Where:
ElecAng is the motor electrical angle.
MtrPoles is the number of poles in the motor.
EncCnt in the encoder count.
MaxEncCnt is value in the the maximum encoder count field.

The EncAngScl parameter should be set to (MtrPoles / 2) * 4096 * 4096 / (MaxEncCnt + 1) so that the above operation can be performed in hardware with a single scaled multiplication as follows:

Electrical angle = EncAngScl * EncCnt / 4096

For this example EncAngScl = 4 * 4096 * 4096 / 2000 = 8388

Pwm Initialization (PwmConfig, PwmPeriod, PwmDeadTm)

Before starting the motor, the host must initialize the PWM parameters for the desired mode, frequency and dead time. For this example we use 10 kHz Symetrical center aligned PWM with a .5 us dead time, so that PwmConfig = 1, PwmPer = 33.333Mhz / (2 * 10 kHz) – 1 = 1666, PwmDeadTm = 17.

Current Feedback Scale Factors (IdScl, IqScl)

Figure 29 shows the calculation scaling for the IRMCK201 FOC current feedback path. The current feedback scale factors should be selected to scale the rotating frame d-q current so that it ranges from –4095 to +4095 at the input of the current loop PI controller summing junction. For this example, we are using a 10-milliohm current sense resistor with a rated motor current of 2.7 Amps or 3.82 Amps Peak. Referring to the figure, this means that the 15-bit programmable scale factor should be set to 4095 / (2900 * 3.82 / 13) = 4.81. Since the scale factors have 10 fractional bits, the final values for the IdScl and IqScl fields are 4.81 * 1024 = 4925. Please refer to the IR2175 datasheet for more information on IR2175-based current sensing.

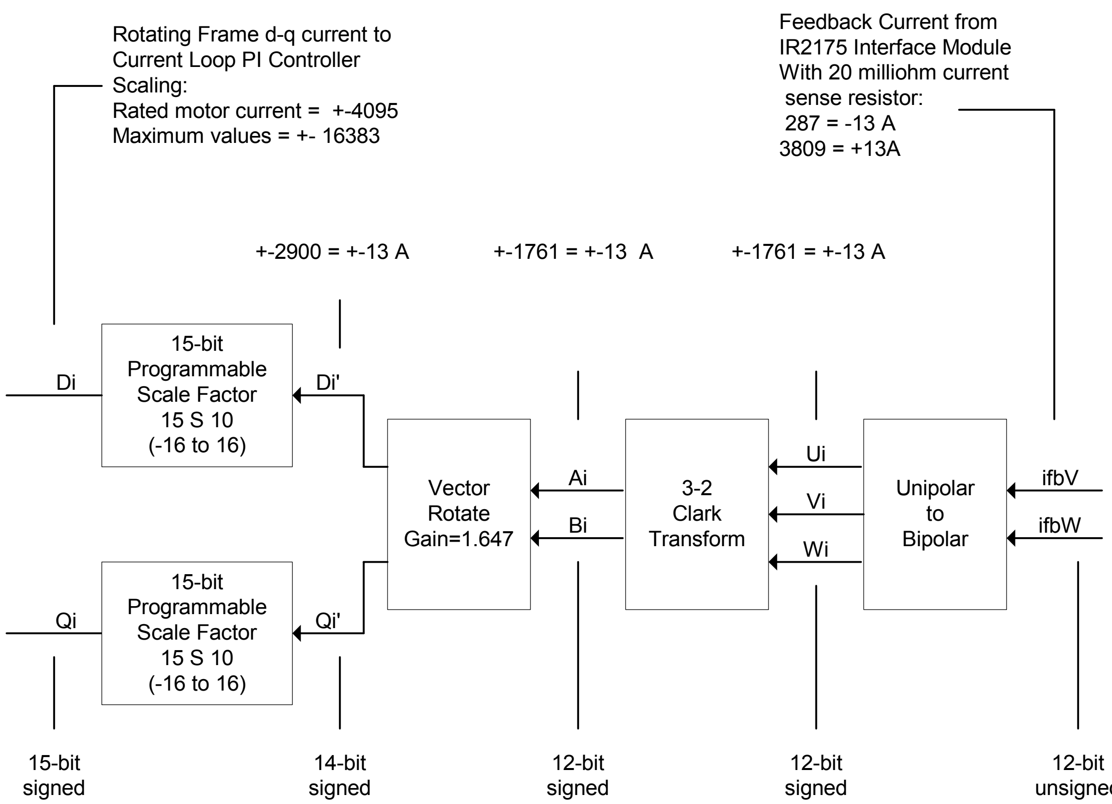


Figure 29. Current Feedback Scaling

3.4.2 Current Offset Calculation

Before running the motor, the IR2175 current measurement offset must be determined so that a correction can be applied to the current feedback before it is used in the FOC algorithm. To determine the IR2175 current measurement offset, we measure the V and W phase currents when the motor is stationary with no applied voltage. In this example, however, we cannot make a valid measurement when the phase voltage is completely off because the IR2175 receives its power from the motor line voltage via bootstrap capacitor. The IRMCK201 works around this by recognizing the condition where PwmEnbW = 1 and FocEnbW = 0, in which case it applies PWM with 0% duty cycle. With 0% PWM, the high side IGBT gate drivers are off, and the low side IGBT gate drivers are on most of the time, but switch off during the dead-time period on each PWM cycle. This short off period drives the IR2175 bootstrap power circuit, which allows a valid current measurement.

When the IRMCK201 detects the current offset calculation condition (PwmEnbW = 1, FocEnbW = 0), it automatically averages 4096 consecutive IR2175 current feedback measurements and stores the resulting current offsets in the IfbVOffs and IfbWOffs read registers. Since the IR2175 performs a current measurement every 1/120Khz = 8.33 usec, each offset calculation takes about 35 msec. In order to apply these offsets to the current feedback measurements, the host sets the IfbOffsEnb bit to "1". The host can also read the IfbVOffs and IfbWOffs registers for diagnostic purposes.

The host can also perform its own current offset calculations by reading the IvFbk and IwFbk registers and writing the desired offsets to the IfbOffsV and IfbOffsW write registers (note that these are different from the IfbVOffs and IfbWOffs read registers). When the IfbOffsEnb bot is set to "0", the IRMCK201 applies these values to the current feedback measurements instead of the automatically calculated values.

Regardless of the method used to determine current offset, the user's system must provide a mechanism for generating the calculation offset condition with the motor stationary and ensure the generation of valid offsets. The IR ServoDesigner tool, for example, accomplishes this each time the "Start Motor" command is executed as follows:

1. Set PwmEnbW = 1, FocEnbW = 0
2. Delay 100ms
3. Set PwmEnbW = 1, FocEnbW = 1 (starts the motor running)

3.4.3 Starting and Stopping the Motor

After initial configuration is complete, the host starts and stops the motor using the PwmEnbW and FocEnbW bits in the System Control Register. As an example, the following "Start Motor" sequence runs the motor at 2000 rpm using the acceleration rate that was set at initial configuration:

1. Set TargetSpeed = 7281 (16383 * 2000 / 4500)
2. Set PwmEnbW = 1, FocEnbW = 0 (start Ifb Offset calculation)
3. Delay 100ms (two IfbOffset calculations)
4. Set PwmEnbW = 1, FocEnbW = 1, IfbOffsEnb = 1, DcCompEnb = 1 (starts the motor running with auto-generated current offset and DC bus compensation)

To stop the motor, the host can use one of the following two methods:

1. Disable PWM and let the motor coast to a stop by setting PwmEnbW = 0 and FocEnbW = 0.
2. Bring the motor to a controlled stop using the deceleration ramp by setting TargetSpd = 0, monitoring the motor speed until it is near 0, and then setting PwmEnbW = 0 and FocEnbW = 0.

3.4.4 Emergency Stop

As a safety feature, the external user interface "Stop" pin can be asserted to disable PWM immediately at any time regardless of the IRMCK201 configuration. See Section 3.5 for more information on the external user interface.

3.4.5 Fault Handling

The IRMCK201 detects the several fault conditions that can occur during system operation. When a fault condition occurs, the IRMCK201 sets the fault status register and leaves PWM disabled until the host clears the fault condition and re-enables PWM and Field-Oriented-Control (FOC). An illustration of typical fault/fault-clear sequence is as follows:

1. When a fault condition occurs, the IRMCK201 clears the PwmEnbW, PwmEnbR, FocEnbW and FocEnbR bits, sets the appropriate bit in the fault status register, and turns on the red LED output pin. At this point the drive is disabled.
2. The host microprocessor, which has presumably been monitoring the FltStatus read register, detects the fault condition and initiates whatever operator alert is appropriate for the particular application. Note that the external LED will also light red if it is connected to the IRMCK201 red LED pin.
3. To clear the fault, the host microprocessor sets the FltClr write bit to "1" to clear the fault and then back to "0" to re-enable fault detection.

3.5 Standalone Operation and Register Initialization via Serial EEPROM

This section describes the register-controlled configuration and operation for an example system that uses the IRMCK201 system in standalone mode, which requires no initialization by a host microprocessor. In standalone mode, the IRMCK201 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 30.

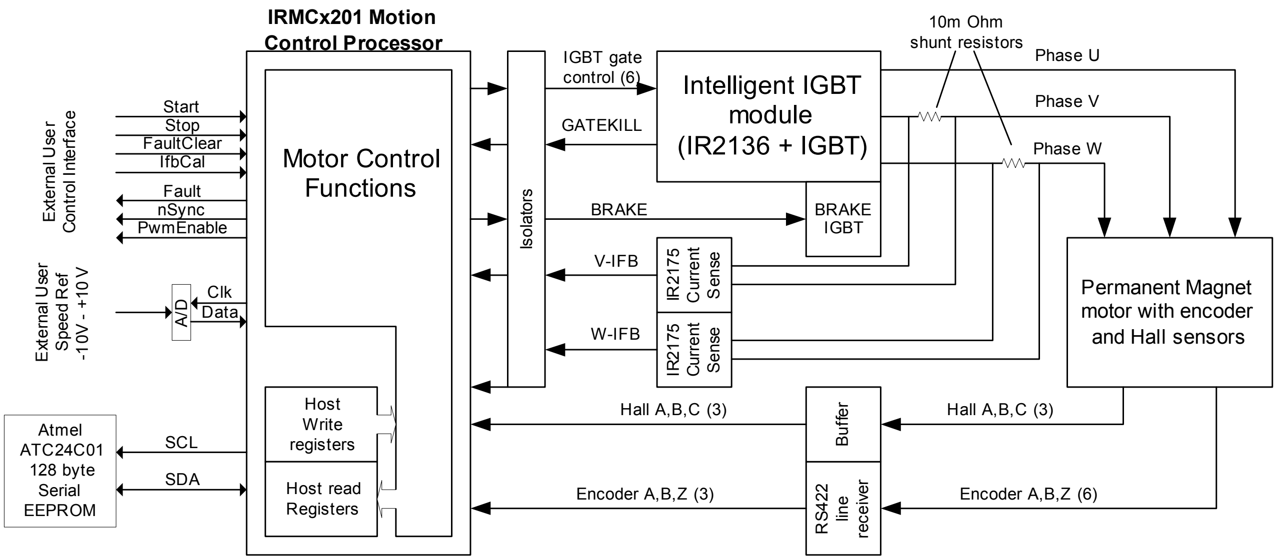


Figure 30. IRMCK201 Standalone System

3.5.1 Register Initialization via EEPROM

Each time the IRMCK201 powers up, it checks for valid EEPROM data by reading a single byte from EEPROM address 0x5D, which represents the IRMCK201 register map version code. If this value matches the IRMCK201 internal register map version, the IRMCK201 EEPROM initialization sequencer performs the following operations:

1. Read 128 sequential bytes from the EEPROM and store them in host write registers 0 - 0x7F.
2. Use the initial hall sensor values as an index to the hall initialization encoder values stored at EEPROM addresses 0x72-0x7D, and read the initial encoder value.
3. Write the initial encoder position to the EnCntW field to set the initial rotor angle.

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.

3.5.2 Current Offset Calculation

When operating the IRMCK201 in standalone, the user initiates current offset calibration by asserting the IfbCal signal for a period of at least 100ms. Before starting the motor, the user should assert IfbCal at least once while the motor is stationary. IfbCal assertion causes the condition PwmEnbW = 1, FocEnbW = 0 as described in the preceding section. The user can perform periodic offset calibration by asserting IfbCal as required by the particular application.

3.5.3 Starting and Stopping the Motor

To start or stop the motor in standalone mode, the user simply drives the "Start" or "Stop" signal to a logic "1". When both signals are asserted, the "Stop" signal has precedence. These signals are intended to be driven as pulsed signals with a minimum pulse width of 100 nsec. Note that driving "Stop" to a DC logic "1" disables the motor unconditionally, so that the "Stop" signal can also be used as an emergency kill switch.

3.5.4 Fault Processing

When the IRMCK201 detects a fault condition, it disables PWM and asserts the "Fault" signal. In standalone mode, the user clears the fault condition using the "FaultClear" signal. Fault processing is otherwise identical to that described Section 3.4.5.

4 Reference

This section provides reference material for the IRMCK201 products. It includes an example encoder input schematic that is referenced in the Techniques Section of this document.

4.1 Example Encoder Input Schematic

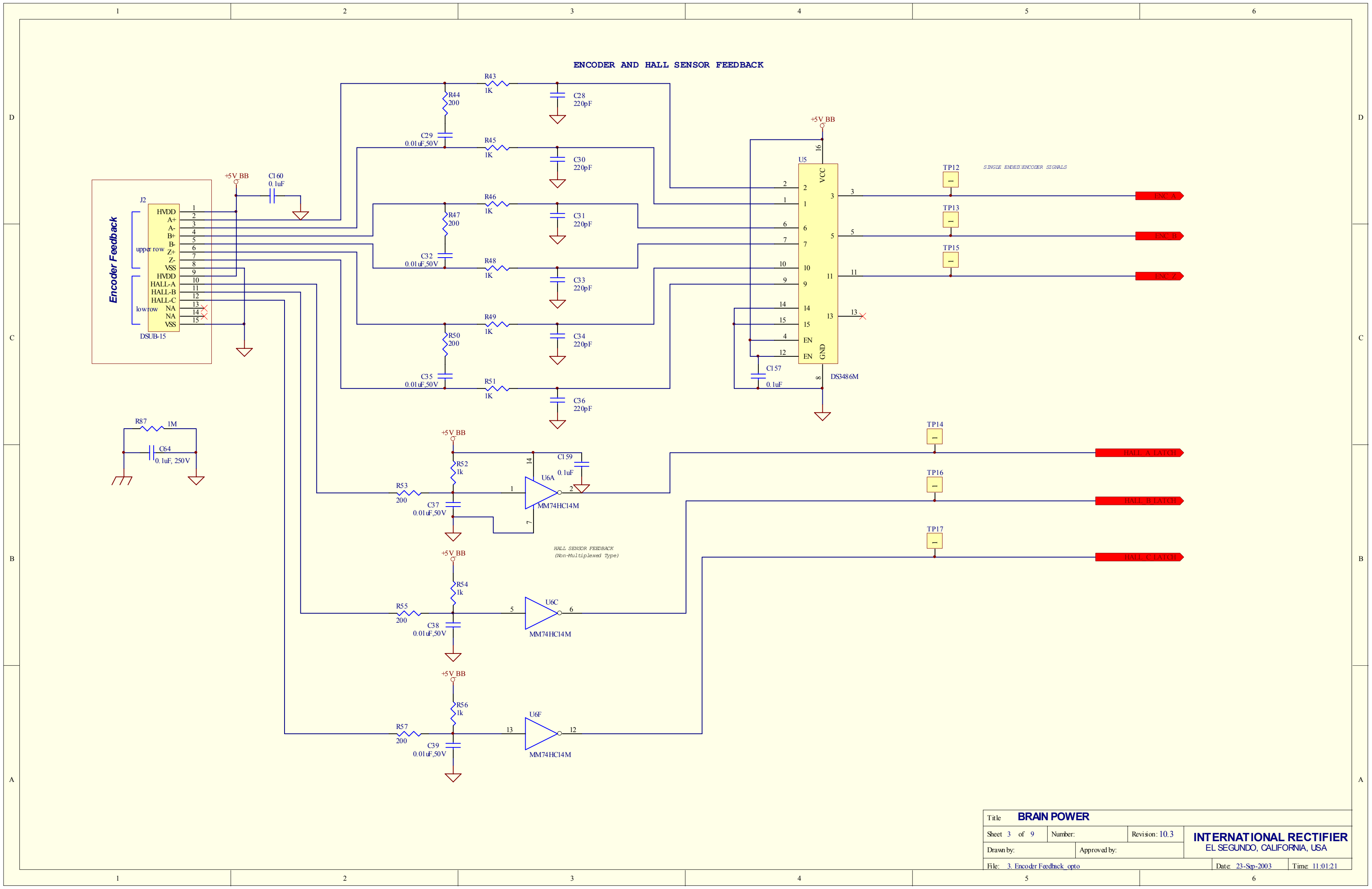


Figure 31. Example Encoder Input Schematic

International Rectifier

WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245, Tel: (310) 322 3331
EUROPEAN HEADQUARTERS: Hurst Green, Oxted, Surrey RH8 9BB, UK Tel: ++ 44 1883 732020
IR CANADA: 7321 Victoria Park Ave., Suite 201, Markham, Ontario L3R 2Z8, Tel: (905) 475 1897
IR GERMANY: Saalburgstrasse 157, 61350 Bad Homburg Tel: ++ 49 6172 96590
IR ITALY: Via Liguria 49, 10071 Borgaro, Torino Tel: ++ 39 11 451 0111
IR FAR EAST: 171 (K&H Bldg.), 30-4 Nishi-ikebukuro 3-Chome, Toshima-ku, Tokyo Japan Tel: 81 3 3983 0086
IR SOUTHEAST ASIA: 315 Outram Road, #10-02 Tan Boon Liat Building, Singapore 0316 Tel: 65 221 8371
http://www.irf.com