Design Guide: TIDM-02017

Two- and Three-Wheeler Traction Inverter Reference Design

Description

This 5-kW, 48-V, traction inverter reference design aims to provide a foundation for engineers to develop high-performance, high-efficiency traction inverter designs for two-wheeler and three-wheeler applications and quickly get the design to market. The features and modularity of the hardware design enable easy evaluation of the C2000™ device family.

Resources

TIDM-02017	Design Folder
TMS320F28P65x-Q1, DRV3255-Q1	Product Folder
LM25184-Q1, LMR33620-Q1	Product Folder
LMR38010-Q1, TPS7A21-Q1	Product Folder
TCAN1044A-Q1	Product Folder
C2000WARE-MOTORCONTROL-SDK	Tool Folder

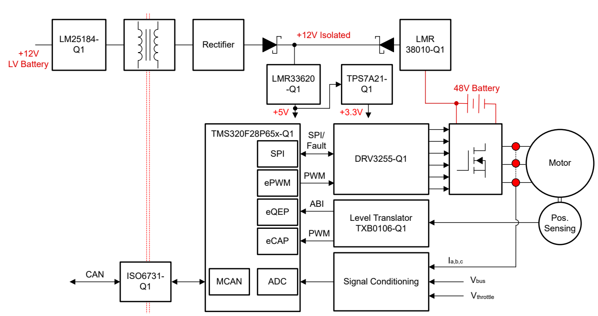
Ask our TI E2E™ support experts

Features

- Hardware and software available with this reference design helps accelerate time to market
- C2000™ real-time microcontrollers demonstrate independent execution of FreeRTOS® and high-performance motor control algorithms
- Implements field weakening control to achieve the full speed and torque range of the motor
- On-chip comparator (CMPSS)-based overcurrent and overvoltage protection provides fast fault response
- DRV3255 integrated gate driver with high drive strength, advanced monitoring, and protection features enables efficient and robust system operation

Applications

- 2-wheeler and 3-wheeler traction drive



1 System Description

1.1 Terminology

FOC	Field Oriented Control
CLA	Control law accelerator
RTOS	Real-time operating system
LDO	Low-dropout regulator
PMSM	Permanent magnet synchronous machine
FWC	Field weakening control
MTPA	Maximum torque per ampere
MOSFET	Metal-oxide-semiconductor field-effect transistor
PWM	Pulse Width Modulation
MCU	Microcontroller Unit
PCB	Printed Circuit Board
RPM	Revolutions Per Minute

1.2 Key System Specifications

The key system specifications are summarized in Table 1-1.

Table 1-1. Key System Specifications

PARAMETER	SPECIFICATIONS (UNITS)	NOTES
P_{OUT}	5 kW	Rated output power
V_{DSmax}	60 V	Maximum Drain-Source Voltage
V_{DC}	48 V	DC bus voltage recommended
I_{DC}	105 A	DC Bus current
f_{SWmax}	25 kHz	Based on the gate driver bias power
I_L	150 A	AC output RMS current
C_{DC}	33.6 µF	DC link capacitor
Dimensions	120 cm × 147.5 cm × 51 cm	Including heat sink height, not including control card height.

- For information on the isolated gate driver, see the DRV3255-Q1 data sheet.
- For information on the microcontroller, refer to the TMSF28P65x-Q1 data sheet.
- For information on the bias supply, refer to UCC14240-Q1 data sheet.
- For information on the power MOSFETs, see the IAUT300N10S5N015 data sheet.
- For information on in phase current sensors, refer to ACS772ECB-250B-PFF-T.

2 System Overview

2.1 Block Diagram

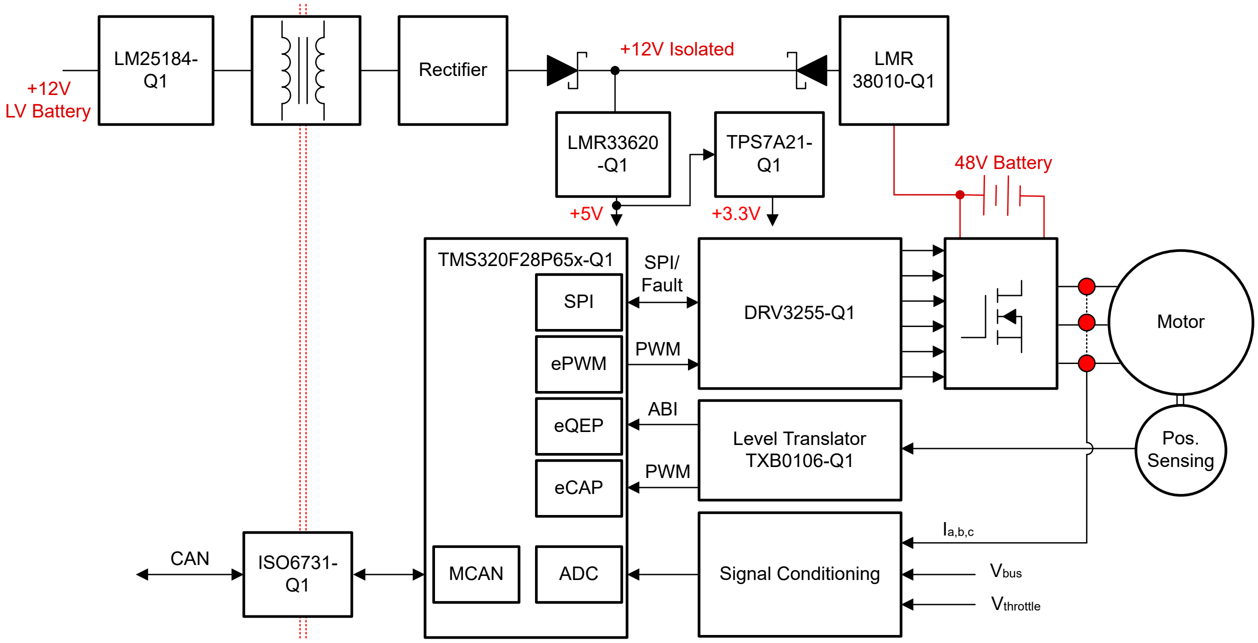


Figure 2-1. TIDM-02017 Two-Wheeler Traction Inverter Block Diagram

2.2 Design Considerations

The primary goal of a two-wheeler traction inverter system is to drive the traction motor which is typically a permanent magnet synchronous motor (PMSM) type or a brushless DC type of motor. Several important design considerations determine the overall functionality and performance of the traction system. At a high level, these include the choice of microcontroller, power stage design, gate driver, motor position sensing, and auxiliary power supply structure.

At the heart of the system is the microcontroller that executes the motor control algorithms and additional high-level functions including communications and protection logic. The TIDM-02017 features the C2000™ real-time control MCU. To address a usual requirement in mid- to high-end two-wheelers, TIDM-02017 software demonstrates the ability of the C2000™ MCU to run an RTOS such as a freeRTOS port on the main C28x CPU and independently running the field-oriented motor control algorithm on the CLA.

For the power stage, each switch of the six-switch inverter consists of four power MOSFETs connected in parallel to increase the ampacity of the inverter. The layout of the surface mount MOSFETs minimizes power loop inductance and allows bottom-side cooling of the power board. To provide efficient operation of the inverter with reasonably short turn-on and turn-off times of the switches, the DRV3255 integrated gate driver with 3.5-A source, 4.5-A sink capability is used. The DRV3255 is a highly-integrated driver with adjustable drive strength capability and a range of diagnostic, monitoring, and protection features for robust system operation. The 90-V DC bus transient capability also makes the design robust against large transient conditions during motor operation.

The inverter measures 279 mm × 291 mm × 115 mm for a total volume of 9.3 L and a power density of up to 32.25 kW/L.

2.3 Highlighted Products

2.3.1 TMS320F28P65x-Q1

The TMS320F28P65x (F28P65x) is a member of the C2000™ real-time microcontroller family of scalable, ultra-low latency devices designed for efficiency in power electronics, including but not limited to: high power density, high switching frequencies, and supporting the use of IGBT, GaN, and SiC technologies.

The real-time control subsystem is based on TI's 32-bit C28x DSP core, which provides 200 MIPS of signal processing performance in each core for floating- or fixed-point code running from either on-chip flash or SRAM. This is equivalent to the 400-MHz processing power on a Cortex®-M7 based device (C28x DSP core gives two times more performance than the Cortex®-M7 core).The C28x CPU is further boosted by the Trigonometric Math Unit (TMU) and Cyclical Redundancy Check (VCRC) extended instruction sets, speeding up common algorithms which are key to real-time control systems. Extended instruction sets enable IEEE double-precision 64-bit floating-point math. Finally, the Control Law Accelerator (CLA) enables an additional 200 MIPS per core of independent processing ability. This is equivalent to the 280-MHz processing power on a Arm® Cortex®-M7 based device (CLA CPU gives 40% more performance than the Cortex®-M7 core).

The F28P65x supports up to 1.28MB of flash memory and up to 248KB of on-chip SRAM is also available to supplement the flash memory.

High-performance analog blocks are tightly integrated with the processing and control units to provide excellent real-time signal chain performance. The Analog-to-Digital Converter (ADC) was enhanced with up to 40 analog channels, 22 of which have general-purpose input/output (GPIO) capability. Implementation of oversampling is greatly simplified with hardware improvement. For safety-critical ADC conversions, a hardware redundancy checker was added and provides the ability to compare ADC conversion results from multiple ADC modules for consistency without additional CPU cycles. Thirty-six frequency-independent PWMs, all with high-resolution capability, enable control of multiple power stages, from three-phase inverters to advanced multilevel power topologies. The PWMs were enhanced with Minimum Dead-Band Logic (MINDL) and Illegal Combo Logic (ICL) features.

2.3.2 DRV3255-Q1

The DRV3255-Q1 device is a highly-integrated three-phase gate driver for 48-V automotive motor drive applications. The device is specifically designed to support high-power motor drive applications by providing 3.5-A peak source and 4.5-A peak sink gate drive currents, and 90-V MOSFET transient overvoltage support. A highly-efficient bootstrap architecture is used to minimize power losses and self-heating of the gate drivers. A charge pump allows for the gate drivers to support 100% PWM duty cycle control.

A wide range of diagnostics, monitoring, and protection features supports a robust motor drive system design. A highly-configurable Active Short Circuit (ASC) function which enables selected external MOSFETs is integrated to achieve the fast response to system faults and to eliminate the needs of external components.

2.3.3 LM25184-Q1

The LM25184-Q1 is a primary-side regulated (PSR) flyback converter with high efficiency over a wide input voltage range of 4.5 V to 42 V. The isolated output voltage is sampled from the primary-side flyback voltage. The high level of integration results in a simple, reliable, and high-density design with only one component crossing the isolation barrier. Boundary conduction mode (BCM) switching enables a compact magnetic design and better than ±1.5% load and line regulation performance. An integrated 65-V power MOSFET provides output power up to 15 W with enhanced headroom for line transients.

The LM25184-Q1 simplifies the implementation of isolated DC/DC supplies with optional features to optimize performance for the target end-equipment. The output voltage is set by one resistor, while an optional resistor improves output voltage accuracy by negating the thermal coefficient of the flyback diode voltage drop. Additional features include an internally-fixed or externally-programmable soft-start, precision enable input with hysteresis for adjustable line UVLO, hiccup-mode overload protection, and thermal shutdown protection with automatic recovery.

2.3.4 TCAN1044A-Q1

The TCAN1044A-Q1 is a high-speed controller area network (CAN) transceiver that meets the physical layer requirements of the ISO 11898-2:2016 high-speed CAN specification. The transceivers have certified electromagnetic compatibility (EMC) operation making the device an excellent choice for classical CAN and CAN FD networks up to 5 megabits per second (Mbps). Up to 8Mbps operation in simpler networks is possible with these devices. The TCAN1044AV-Q1 includes internal logic level translation through the V_{IO} pin to allow for interfacing the I/Os of the transceiver directly to 1.8-V, 2.5-V, 3.3-V, or 5-V logic levels. The transceiver supports a low-power standby mode and wake over CAN which is compliant to the ISO 11898-2:2016 defined wake-up pattern (WUP).

3 System Design Theory

3.1 Three-Phase PMSM Drive

Permanent Magnet Synchronous motor (PMSM) has a wound stator, a permanent magnet rotor assembly, and internal or external devices to sense rotor position. The sensing devices provide position feedback for adjusting frequency and amplitude of stator voltage reference properly to maintain rotation of the magnet assembly. The combination of an inner permanent magnet rotor and outer windings offers the advantages of low rotor inertia, efficient heat dissipation, and reduction of the motor size.

- Synchronous motor construction: Permanent magnets are rigidly fixed to the rotating axis to create a constant rotor flux. This rotor flux usually has a constant magnitude. When energized, the stator windings create a rotating electromagnetic field. To control the rotating magnetic field, the stator currents must be controlled.
- The actual structure of the rotor varies depending on the power range and rated speed of the machine. Permanent magnets are an excellent choice for synchronous machines ranging up-to a few Kilowatts. For higher power ratings the rotor usually consists of windings in which a DC current circulates. The mechanical structure of the rotor is designed for number of poles desired, and the desired flux gradients desired.
- The interaction between the stator and rotor fluxes produces torque. Since the stator is firmly mounted to the frame, and the rotor is free to rotate, the rotor rotates, producing a useful mechanical output as shown in Figure 3-1.
- The angle between the rotor magnetic field and stator field must be carefully controlled to produce maximum torque and achieve high electromechanical conversion efficiency. For this purpose fine-tuning is needed after closing the speed loop using a sensorless algorithm to draw the minimum amount of current under the same speed and torque conditions.
- The rotating stator field must rotate at the same frequency as the rotor permanent magnetic field; otherwise, the rotor experiences rapidly alternating positive and negative torque. This results in less than excellent torque production, and excessive mechanical vibration, noise, and mechanical stresses on the machine parts. In addition, if the rotor inertia prevents the rotor from being able to respond to these oscillations, the rotor stops rotating at the synchronous frequency, and responds to the average torque as seen by the stationary rotor: Zero. This means that the machine experiences a phenomenon known as *pull-out*. This is also the reason why the synchronous machine is not self starting.
- The angle between the rotor field and the stator field must be equal to 90º to obtain the highest mutual torque production. This synchronization requires knowing the rotor position to generate the right stator field.
- The stator magnetic field can be made to have any direction and magnitude by combining the contribution of different stator phases to produce the resulting stator flux.

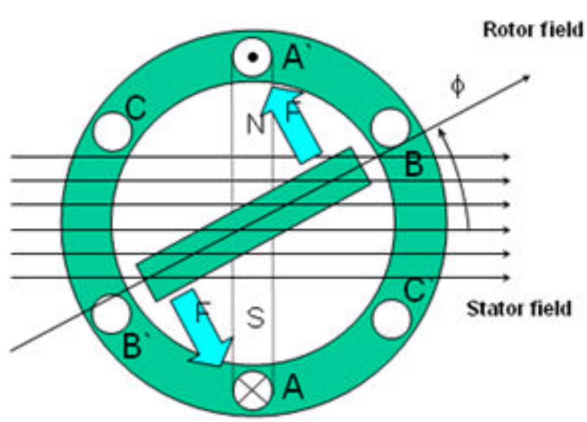


Figure 3-1. Interaction Between the Rotating Stator Flux and the Rotor Flux Produces Torque

3.1.1 Field-Oriented Control of PM Synchronous Motor

To achieve better dynamic performance, a more complex control scheme needs to be applied, to control the PM motor. With the mathematical processing power offered by the microcontrollers, advanced control strategies can be implemented, which use mathematical transformations to decouple the torque generation and the magnetization functions in PM motors. Such de-coupled torque and magnetization control is commonly called rotor flux oriented control, or simply Field-Oriented Control (FOC).

In a direct current (DC) motor, the excitation for the stator and rotor is independently controlled, the produced torque and the flux can be independently tuned as shown in Figure 3-2. The strength of the field excitation (for example, the magnitude of the field excitation current) sets the value of the flux. The current through the rotor windings determines how much torque is produced. The commutator on the rotor plays an interesting part in the torque production. The commutator is in contact with the brushes, and the mechanical construction is designed to switch into the circuit the windings that are mechanically aligned to produce the maximum torque. This arrangement then means that the torque production of the machine is fairly near exceptional all the time. The key point here is that the windings are managed to keep the flux produced by the rotor windings orthogonal to the stator field.

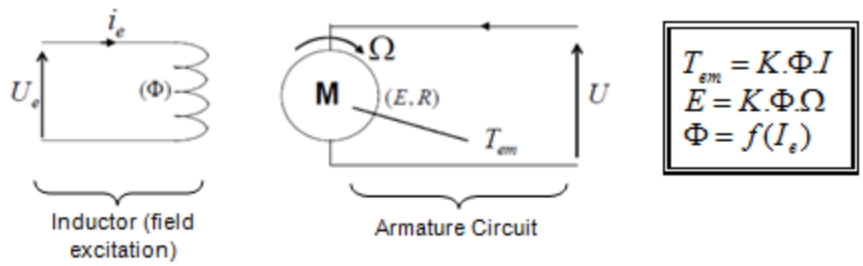


Figure 3-2. Flux and Torque are Independently Controlled in DC Motor Model

The goal of the FOC (also called vector control) on synchronous and asynchronous machines is to be able to separately control the torque-producing and magnetizing flux components. FOC control allows decoupling of the torque and of the magnetizing flux components of stator current. With decoupled control of the magnetization, the torque producing component of the stator flux can now be thought of as independent torque control. To decouple the torque and flux, it is necessary to engage several mathematical transforms, and this is where the microcontrollers add the most value. The processing capability provided by the microcontrollers enables these mathematical transformations to be carried out very quickly. This, in turn, implies that the entire algorithm controlling the motor can be executed at a fast rate, enabling higher dynamic performance. In addition to the decoupling, a dynamic model of the motor is now used for the computation of many quantities such as rotor flux angle and rotor speed. This means that the effect is accounted for, and the overall quality of control is better.

According to the electromagnetic laws, the torque produced in the synchronous machine is equal to the vector cross product of the two existing magnetic fields as in Equation 1 .

$$\tau_{em} = \overrightarrow{B}_{stator} \times \overrightarrow{B}_{rotor} \quad (1)$$

This expression shows that the torque is maximum if stator and rotor magnetic fields are *orthogonal* meaning to maintain the load at 90 degrees. If this condition can be provided all the time and if the flux can be oriented correctly, the torque ripple is reduced and a better dynamic response is provided. However, the constraint is to know the rotor position: this can be achieved with a position sensor such as incremental encoder. For low-cost applications where the rotor is not accessible, different rotor position observer strategies are applied to get rid of position sensor.

In brief, the goal is to maintain the rotor and stator flux in quadrature: the goal is to align the stator flux with the q axis of the rotor flux, for example, orthogonal to the rotor flux. To do this, the stator current component in quadrature with the rotor flux is controlled to generate the commanded torque, and the direct component is set to zero. The direct component of the stator current can be used in some cases for field weakening, which has the effect of opposing the rotor flux, and reducing the back-emf, which allows for operation at higher speeds.

The FOC consists of controlling the stator currents represented by a vector. This control is based on projections which transform a three-phase time and speed dependent system into a two coordinate (d and q coordinates) time invariant system. These projections lead to a structure similar to that of a DC machine control. FOC machines need two constants as input references: the torque component (aligned with the q coordinate) and the flux component (aligned with d coordinate). As FOC is simply based on projections, the control structure handles instantaneous electrical quantities. This makes the control accurate in every working operation (steady state and transient) and independent of the limited bandwidth mathematical model. The FOC thus solves the classic scheme problems, in the following ways:

- The ease of reaching constant reference (torque component and flux component of the stator current)
- The ease of applying direct torque control because in the (d, q) reference frame the expression of the torque is defined in Equation 2.

$$\tau_{em} \propto \psi_R \times i_{sq} \quad (2)$$

By maintaining the amplitude of the rotor flux (ψ_R) at a fixed value, a linear relationship between torque and torque component (i_{Sq}) is obtained. Therefore, the torque can be controlled by controlling the torque component of the stator current vector.

3.1.1.1 Space Vector Definition and Projection

The 3-phase voltages, currents, and fluxes of AC motors can be analyzed in terms of complex space vectors. With regard to the currents, the space vector can be defined as follows. Assuming that i_a, i_b, i_c are the instantaneous currents in the stator phases, then the complex stator current vector is defined in Equation 3.

$$\bar{i}_s = i_a + \alpha i_b + \alpha^2 i_c \quad (3)$$

where

- $\alpha = e^{j\frac{2}{3}\pi}$ and $\alpha^2 = e^{j\frac{4}{3}\pi}$ represent the spatial operators

Figure 3-3 shows the stator current complex space vector.

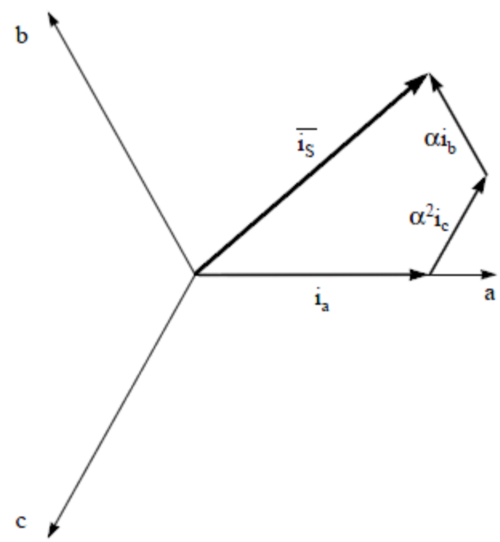


Figure 3-3. Stator Current Space Vector and Component in (a, b, c) Frame

where

- a, b, and c are the three-phase system axes

This current space vector depicts the three-phase sinusoidal system which still needs to be transformed into a two time invariant co-ordinate system. This transformation can be split into two steps:

- $(a, b) \Rightarrow (\alpha, \beta)$ (Clarke transformation) which outputs a 2-coordinate time-variant system
- $(\alpha, \beta) \Rightarrow (d, q)$ (Park transformation) which outputs a 2-coordinate time-invariant system

3.1.1.1.1 $(\mathrm{a},\ \mathrm{b}) \Rightarrow (\alpha, \beta)$ Clarke Transformation

The space vector can be reported in another reference frame with only two orthogonal axis called (α, β). Assuming that the axis a and the axis α*lpha* are in the same direction yields the vector diagram shown in Figure 3-4.

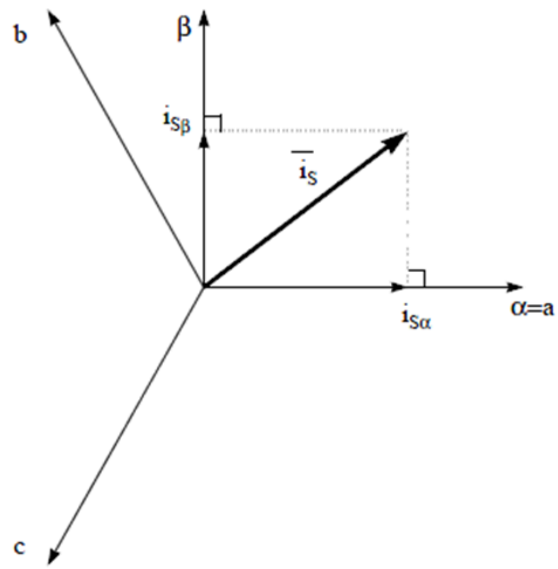


Figure 3-4. Stator Current Space Vector in the Stationary Reference Frame

The projection that modifies the 3-phase system into the (α, β) 2-dimension orthogonal system is presented in Equation 4.

$$\mathrm{i}_{s\alpha} = \mathrm{i}_\mathrm{a}$$
$$\mathrm{i}_{s\beta} = \frac{1}{\sqrt{3}}\mathrm{i}_\mathrm{a} + \frac{2}{\sqrt{3}}\mathrm{i}_\mathrm{b} \quad (4)$$

The two phase (α, β) currents are still dependent on time and speed.

3.1.1.1.2 $(\alpha, \beta) \Rightarrow (\mathrm{d},\ \mathrm{q})$ Park Transformation

This is the most important transformation in the FOC. In fact, this projection modifies a 2-phase orthogonal system (α, β) in the (d, q) rotating reference frame. Considering the d axis aligned with the rotor flux, Figure 3-5 shows the relationship for the current vector from the two reference frame.

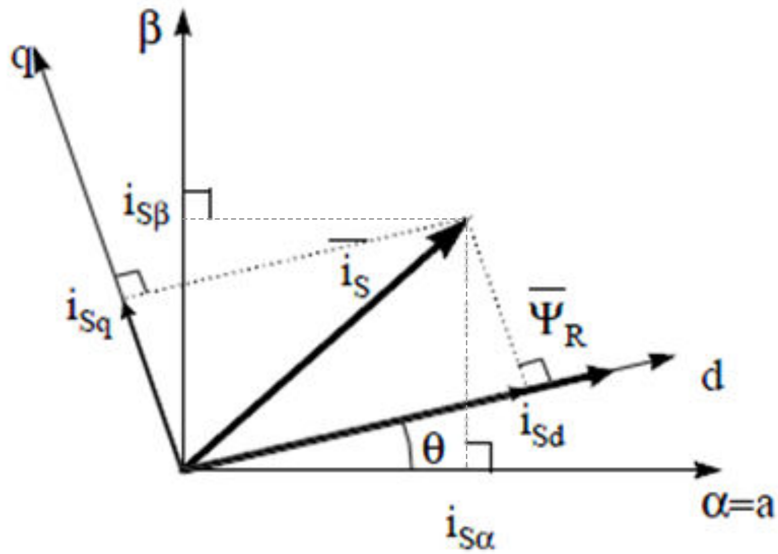


Figure 3-5. Stator Current Space Vector in the d,q Rotating Reference Frame

The flux and torque components of the current vector are determined by Equation 5.

$$i_{sd} = i_{s\alpha}\cos(\theta) + i_{s\beta}\sin(\theta)$$
$$i_{sq} = -i_{s\alpha}\sin(\theta) + i_{s\beta}\cos(\theta) \quad (5)$$

where

- θ is the rotor flux position

These components depend on the current vector (α, β) components and on the rotor flux position; if the right rotor flux position is known then, by this projection, the d,q component becomes a constant. Two phase currents now turn into dc quantity (time-invariant). At this point the torque control becomes easier where constant i_{sd} (flux component) and i_{sq} (torque component) current components controlled independently.

3.1.1.2 Basic Scheme of FOC for AC Motor

Figure 3-6 summarizes the basic scheme of torque control with FOC.

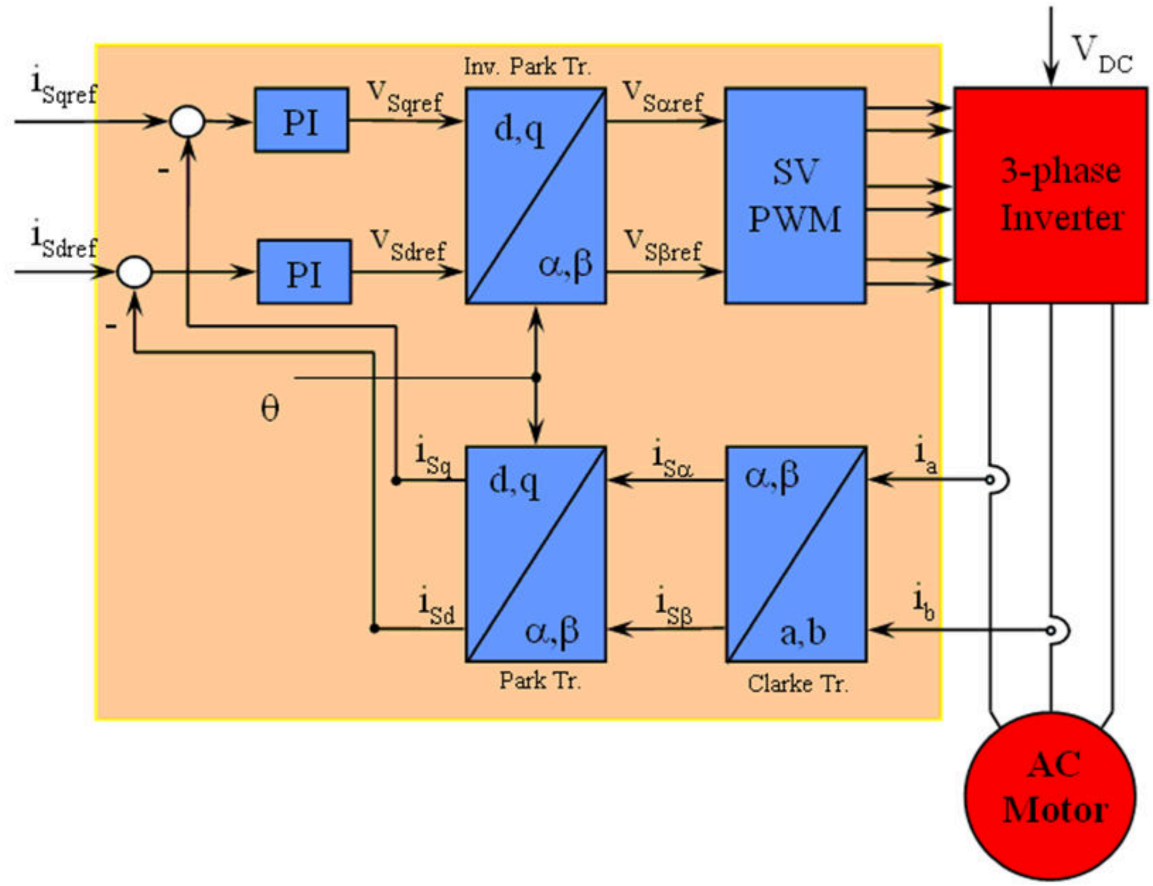


Figure 3-6. Basic Scheme of FOC for AC Motor

Two motor phase currents are measured. These measurements feed the Clarke transformation module. The outputs of this projection are designated $i_{s\alpha}$ and $i_{s\beta}$. These two components of the current are the inputs of the Park transformation that gives the current in the d,q rotating reference frame. The i_{sd} and i_{sq} components are compared to the references i_{sdref} (the flux reference component) and i_{sqref} (the torque reference component). At this point, this control structure shows an interesting advantage: the structure can be used to control either synchronous or induction machines by simply changing the flux reference and obtaining rotor flux position. As in synchronous permanent magnet a motor, the rotor flux is fixed determined by the magnets; there is no need to create one. Hence, when controlling a PMSM, set i_{sdref} to zero. As an AC induction motor needs a rotor flux creation to operate, the flux reference must not be zero. This conveniently solves one of the major drawbacks of the *classic* control structures: the portability from asynchronous to synchronous drives. The torque command i_{sqref} can be the output of the speed regulator when a speed FOC is used. The outputs of the current regulators are V_{sdref} and V_{sqref}; these outputs are applied to the inverse Park transformation. The outputs of this projection are $V_{s\alpha ref}$ and $V_{s\beta ref}$ which are the components of the stator vector voltage in the (α, β) stationary orthogonal reference frame. These are the inputs of the Space Vector PWM. The outputs of this block are the signals that drive the inverter. Note that both Park and inverse Park transformations need the rotor flux position. Obtaining this rotor flux position depends on the AC machine type (synchronous or asynchronous machine).

3.1.1.3 Rotor Flux Position

Knowledge of the rotor flux position is the core of the FOC. In fact if there is an error in this variable the rotor flux is not aligned with the d-axis and i_{sd} and i_{sq} are incorrect flux and torque components of the stator current. Figure 3-7 shows the (a, b, c), (α, β) and (*d, q*) reference frames, and the correct position of the rotor flux, the stator current and stator voltage space vector that rotates with d, q reference at synchronous speed.

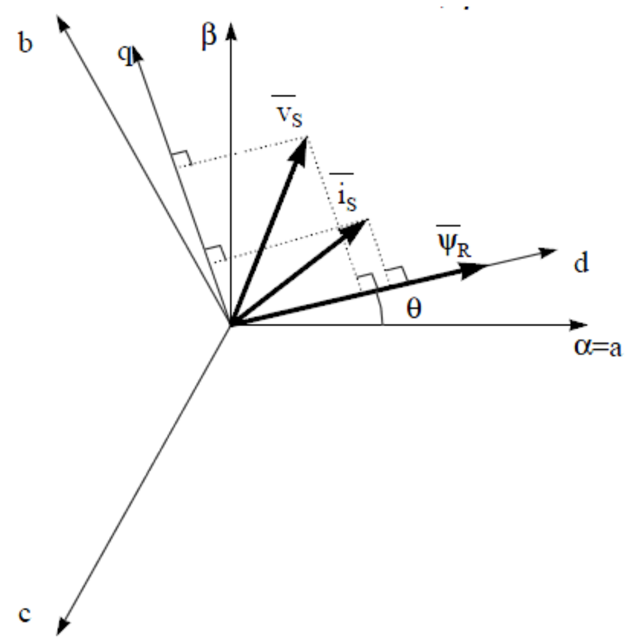


Figure 3-7. Current, Voltage and Rotor Flux Space Vectors in the (d, q) Rotating Reference Frame

The measure of the rotor flux position is different when considering the synchronous or asynchronous motor:

- In the synchronous machine the rotor speed is equal to the rotor flux speed. Then θ (rotor flux position) is directly measured by the position sensor or by integration of rotor speed.
- In the asynchronous machine, the rotor speed is not equal to the rotor flux speed (there is a slip speed), then a particular method is needed to calculate θ. The basic method is the use of the current model which needs two equations of the motor model in *d, q* reference frame.

Theoretically, the FOC for the PMSM drive allows the motor torque to be controlled independently with the flux like DC motor operation. In other words, the torque and flux are decoupled from each other. The rotor position is required for variable transformation from stationary reference frame to synchronously rotating reference frame. As a result of this transformation (so called Park transformation), q-axis current is controlling torque while d-axis current is forced to zero.

3.2 Field Weakening (FW) Control

Permanent magnet synchronous motor (PMSM) is widely used in home appliance applications due to the high power density, high efficiency, and wide speed range. The PMSM includes two major types: the surface-mounted PMSM (SPM), and the interior PMSM (IPM). SPM motors are easier to control due to the linear relationship between the torque and q-axis current. The aim of the field weakening control is to optimize to reach the highest power and efficiency of a PMSM drive. Field weakening control can enable a motor operation over the base speed, expanding the operating limits to reach speeds higher than the rated speed and allow exceptional control across the entire speed and voltage range.

The voltage equations of the mathematical model of an IPMSM can be described in d-q coordinates as shown in Equation 6 and Equation 7.

$$v_d = L_d \frac{di_d}{dt} + R_s i_d - p\omega_m L_q i_q \tag{6}$$

$$v_q = L_q \frac{di_q}{dt} + R_s i_q + p\omega_m L_d i_d + p\omega_m \psi_m \tag{7}$$

Figure 3-8 shows the dynamic equivalent circuit of an IPM synchronous motor.

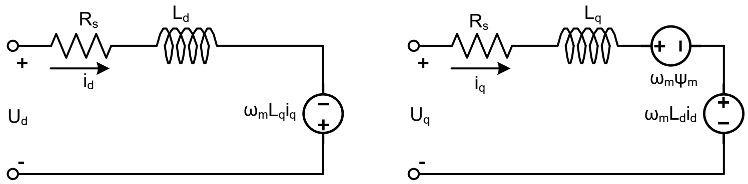


Figure 3-8. Equivalent Circuit of an IPM Synchronous Motor

The total electromagnetic torque generated by the IPMSM can be expressed as Equation 8 that the produced torque is composed of two distinct terms. The first term corresponds to the mutual reaction torque occurring between torque current i_q and the permanent magnet ψ_m , while the second term corresponds to the reluctance torque due to the differences in d-axis and q-axis inductance.

$$T_e = \frac{3}{2}p\left[\psi_m i_q + \left(L_d - L_q\right)i_d i_q\right] \tag{8}$$

In most applications, IPMSM drives have speed and torque constraints, mainly due to inverter or motor rating currents and available DC link voltage limitations respectively. These constraints can be expressed with the mathematical equations Equation 9 and Equation 10.

$$I_a = \sqrt{i_d^2 + i_q^2} \leq I_{max} \tag{9}$$

$$V_a = \sqrt{v_d^2 + v_q^2} \leq V_{max} \tag{10}$$

where

- V_{max} and I_{max} are the maximum allowable voltage and current of the inverter or motor

In a two-level three-phase Voltage Source Inverter (VSI) fed machine, the maximum achievable phase voltage is limited by the DC link voltage and the PWM strategy. The maximum voltage is limited to the value as shown in Equation 11 if Space Vector Modulation (SVPWM) is adopted.

$$\sqrt{v_d^2 + v_q^2} \leq v_{max} = \frac{v_{dc}}{\sqrt{3}} \tag{11}$$

Usually the stator resistance R_s is negligible at high speed operation and the derivative of the currents is zero in steady state, thus Equation 12 is obtained as shown.

$$\sqrt{L_d^2\left(i_d + \frac{\psi_{pm}}{L_d}\right)^2 + L_q^2 i_q^2} \leq \frac{V_{max}}{\omega_m} \tag{12}$$

The current limitation of Equation 9 produces a circle of radius I_{max} in the d-q plane, and the voltage limitation of Equation 11 produces an ellipse whose radius V_{max} decreases as speed increases. The resultant d-q plane current vector must be controlled to obey the current and voltage constraints simultaneously. According to these constraints, three operation regions for the IPMSM can be distinguished as shown in Figure 3-9.

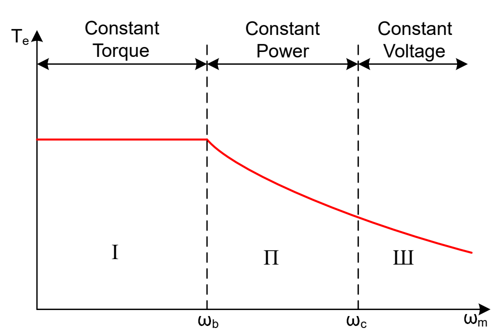


Figure 3-9. IPMSM Control Operation Regions

1. Constant Torque Region: MTPA can be implemented in this operation region to provide maximum torque generation.
2. Constant Power Region: Field-weakening control must be employed and the torque capacity is reduced as the current constraint is reached.
3. Constant Voltage Region: In this operation region, deep field-weakening control keeps a constant stator voltage to maximize the torque generation.

In the constant torque region, according to Equation 8, the total torque of an IPMSM includes the electromagnetic torque from the magnet flux linkage and the reluctance torque from the saliency between L_d and L_q . The electromagnetic torque is proportional to the q-axis current i_q , and the reluctance torque is proportional to the multiplication of the d-axis current i_d , the q-axis current i_q , and the difference between L_d and L_q .

Conventional vector control systems of SPM motors only utilizes electromagnetic torque by setting the commanded i_d to zero for non-field-weakening modes. But while the IPMSM utilizes the reluctance torque of the motor, the designer must also control the d-axis current. The aim of the MTPA control is to calculate the reference currents i_d and i_q to maximize the ratio between produced electromagnetic torque and reluctance torque. The relationship between i_d and i_q , and the vectorial sum of the stator current I_s is shown in the following equations.

$$I_s = \sqrt{i_d^2 + i_q^2} \tag{13}$$

$$I_d = I_s \cos\beta \tag{14}$$

$$I_q = I_s \sin\beta \tag{15}$$

where

- β is the stator current angle in the synchronous (d-q) reference frame

Equation 8 can be expressed as Equation 16 where I_s substituted for i_d and i_q .

Equation 16 shows that motor torque depends on the angle of the stator current vector:

$$T_e = \frac{3}{2} p I_s \sin\beta \left[\psi_m + \left(L_d - L_q\right) I_s \cos\beta \right] \tag{16}$$

This equation shows the maximum efficiency point can be calculated when the motor torque differential is equal to zero. The MTPA point can be found when this differential, $\frac{dT_e}{d\beta}$ is zero as given in Equation 17.

$$\frac{dT_e}{d\beta} = \frac{3}{2} p \left[\psi_m I_s \cos\beta + \left(L_d - L_q\right) I_s^2 \cos 2\beta \right] = 0 \tag{17}$$

Following this equation, the current angle of the MTPA control can be derived as in Equation 18.

$$\beta_{mtpa} = \cos^{-1} \frac{-\psi_m + \sqrt{\psi_m^2 + 8 \times \left(L_d - L_q\right)^2 \times I_s^2}}{4 \times \left(L_d - L_q\right) \times I_s} \tag{18}$$

Thus, the effective d-axis and q-axis reference currents can be expressed by Equation 19 and Equation 20 using the current angle of the MTPA control.

$$I_d = I_s \times \cos\beta_{mtpa} \tag{19}$$

$$I_q = I_s \times \sin\beta_{mtpa} \tag{20}$$

However, as shown in Equation 18, the angle of the MTPA control, β_{mtpa} is related to d-axis and q-axis inductance. This means that the variation of inductance impedes the ability to find the exceptional MTPA point. To improve the efficiency of a motor drive, estimate the d-axis and q-axis inductance online, but the parameters

L_d and L_q are not easily measured online and are influenced by saturation effects. A robust Look-Up Table (LUT) method provides controllability under electrical parameter variations. Usually, to simplify the mathematical model, the coupling effect between d-axis and q-axis inductance can be neglected. Thus, assume that L_d changes with i_d only, and L_q changes with i_q only. Consequently, d- and q-axis inductance can be modeled as a function of the d-q currents respectively, as shown in Equation 21 and Equation 22.

$$L_d = f_1(i_d, i_q) = f_1(i_d) \quad (21)$$

$$L_q = f_2(i_q, i_d) = f_2(i_q) \quad (22)$$

Reduce the ISR calculation burden by simplifying Equation 18. The motor-parameter-based constant, K_{mtpa} is expressed instead as Equation 24, where K_{mtpa} is computed in the background loop using the updated L_d and L_q .

$$K_{mtpa} = \frac{\psi_m}{4 \times (L_q - L_d)} = 0.25 \times \frac{\psi_m}{(L_q - L_d)} \quad (23)$$

$$\beta_{mtpa} = \cos^{-1}\left(K_{mtpa}/I_s - \sqrt{(K_{mtpa}/I_s)^2 + 0.5}\right) \quad (24)$$

A second intermediate variable, G_{mtpa} described in Equation 25, is defined to further simplify the calculation. Using G_{mtpa} , the angle of the MTPA control, β_{mtpa} can be calculated as Equation 26. These two calculations are performed in the ISR to achieve a real current angle β_{mtpa} .

$$G_{mtpa} = K_{mtpa}/I_s \quad (25)$$

$$\beta_{mtpa} = \cos^{-1}\left(G_{mtpa} - \sqrt{G_{mtpa}^2 + 0.5}\right) \quad (26)$$

In all cases, the magnetic flux can be weakened to extend the achievable speed range by acting on the direct axis current i_d. As a consequence of entering this constant power operating region, field weakening control is chosen instead of the MTPA control used in constant power and voltage regions. Since the maximum inverter voltage is limited, PMSM motors cannot operate in such speed regions where the back-electromotive force, almost proportional to the permanent magnet field and motor speed, is higher than the maximum output voltage of the inverter. The direct control of magnet flux is not an option in PM motors. However, the air gap flux can be weakened by the demagnetizing effect due to the d-axis armature reaction by adding a negative i_d. Considering the voltage and current constraints, the armature current and the terminal voltage are limited as Equation 9 and Equation 10. The inverter input voltage (DC-Link voltage) variation limits the maximum output of the motor. Furthermore, the maximum fundamental motor voltage also depends on the PWM method used. In Equation 12, the IPMSM has two factors: one is a permanent magnet value and the other is made by inductance and current of flux.

Figure 3-10 shows the typical control structure is used to implement field weakening. β_{fw} is the output of the field-weakening (FW) PI controller and generates the reference i_d and i_q . Before the voltage magnitude reaches the limit, the input of the PI controller of FW is always positive and therefore the output is always saturated at 0.

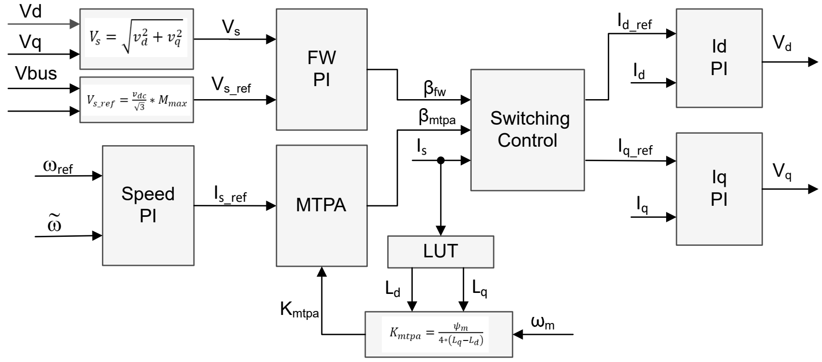


Figure 3-10. Block Diagram of Field-Weakening and Maximum Torque per Ampere Control

The field-weakening control module shown in Figure 3-10 generates the current angle β_{fw} based on input parameters as shown in Figure 3-11.

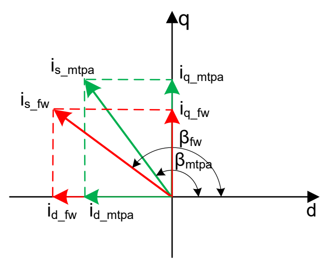


Figure 3-11. Current Phasor Diagram of an IPMSM During FW and MTPA

In a typical application, if using both MTPA and FW control, the switching control module is used to determine angle of application, and then calculate the reference i_d and i_q as shown in Equation 14 and Equation 15. The current angle is chosen as in the following: Equation 27 and Equation 28.

$$\beta = \beta_{\text{fw}} \text{ if } \beta_{\text{fw}} > \beta_{\text{mtpa}} \quad (27)$$

$$\beta = \beta_{\text{mpta}} \text{ if } \beta_{\text{fw}} < \beta_{\text{mtpa}} \quad (28)$$

4 Hardware, Software, Testing Requirements, and Test Results

4.1 Hardware Requirements

4.1.1 Hardware Board Overview

Figure 4-1 shows an overview of a typical two-wheeler traction inverter system.

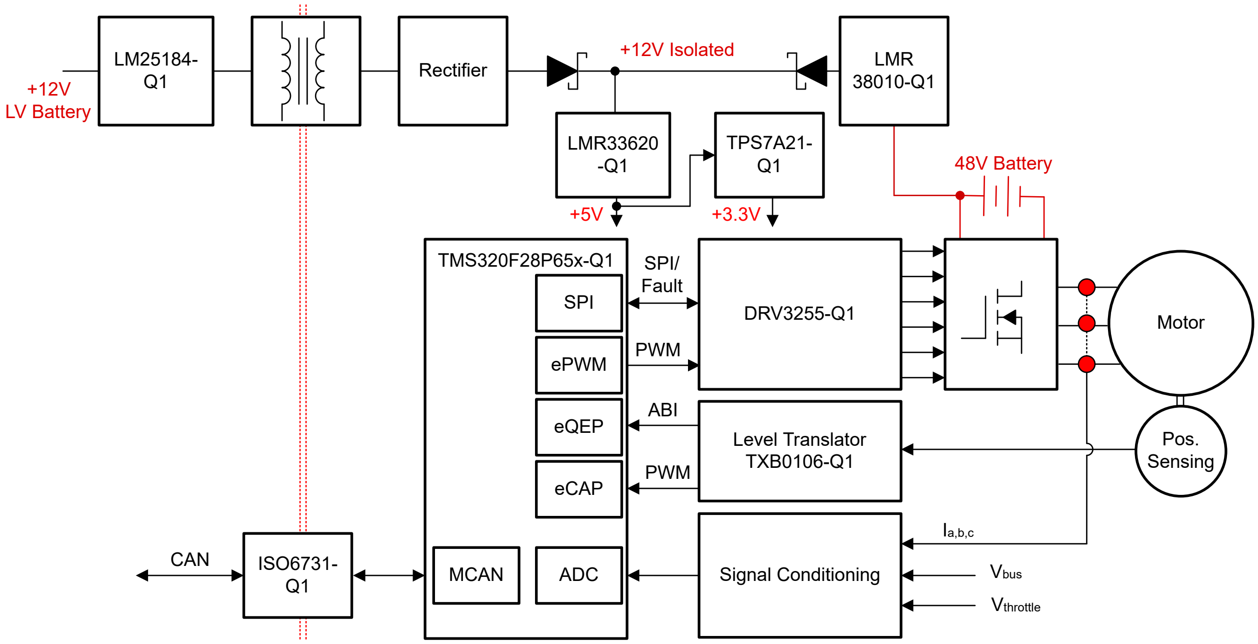


Figure 4-1. TIDM-02017 Hardware Board Block Diagram

The motor control board has functional groups that enable a complete motor drive system. The following is a list of the blocks on the board and their functions, Figure 4-2 shows the top view of the board and different blocks of the TIDM-02017 PCB:

- DC bus input
  - DC bus input connector
  - 33.6-μF film capacitor
- 3-phase inverter
  - Up to 5-kW three-phase inverter supports PMSM or IPM
  - 20-kHz switching frequency
  - In-phase Hall-effect current sensing
- Control
  - TMDSCNCD28P65X MCU controlCARD™
  - 200-MHz, 32-bit CPU with FPU, CLA, and TMU
- Auxiliary power supply
  - An external 12-V power supply is required for the board. Power can be provided by connecting a AC-12-V adapter.

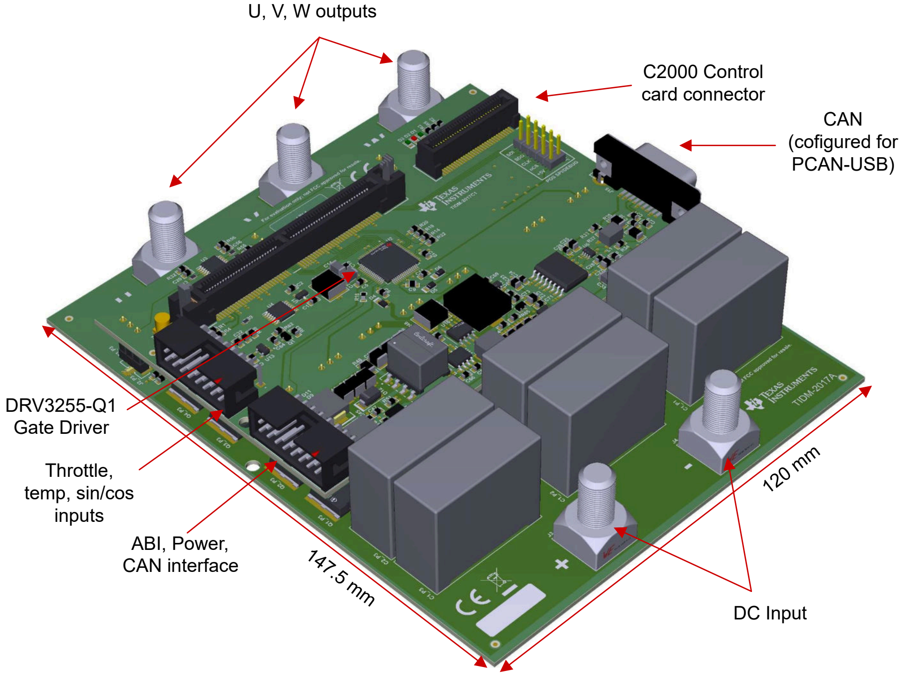


Figure 4-2. Layout of Two-Wheeler Traction Inverter Reference Design Board

TI recommends taking the following precautions when using the board:

- Do not touch any part of the board or components connected to the board when the board is energized.
- Do not touch any part of the board, the kit or the assembly when energized.
- Power components can be hot especially at high loads.

The connectors J16 and J17 are the external interfaces for the control board. The pinouts of each of the connectors are provided in Table 4-1 and Table 4-2, respectively.

Table 4-1. Pinout for Connector J16

PIN	SIGNAL	PIN	SIGNAL
1	CAN_L	2	CAN_H
3	GND_Bat	4	GND_Bat
5	+12V_Bat	6	+12V Bat
7	+5_Pos	8	GND
9	QEP_I	10	GND
11	PWM/GPI	12	GND
13	QEP_B	14	GND
15	QEP_A	16	GND

Table 4-2. Pinout for Connector J17

PIN	SIGNAL	PIN	SIGNAL
1	GND	2	GND
3	+5_Throttle	4	+5_Throttle
5	AMR_SINP	6	AMR_SINN
7	AMR_COSP	8	AMR_COSN
9	Inv_Temp	10	GND
11	Motor_Temp	12	GND
13	Throttle_2	14	GND
15	Throttle_1	16	GND

4.1.2 Test Conditions

The following conditions and equipment apply for the user to test the reference design software:

- For input, the power supply source supports up to 48 V if using the DC power supply. Set the input current limit of input DC power supply to 120 A. However, TI recommends starting with a lower current limit during initial board bring-up.
- For output, a 3-phase PMSM with dynamometer.

4.1.3 Test Equipment Required for Board Validation

- DC power supply
- Digital oscilloscope
- 3-phase permanent magnet synchronous motor
- Dynamometer

4.2 Test Setup

4.2.1 Hardware Setup

Use the following steps to setup the hardware for testing:

1. Connect a USB cable to the TMDSCNCD28P65X controlCARD for JTAG connection.
2. Connect the motor wires to the output terminals marked U, V, and W.
3. Connect the auxiliary 12-V DC power supply through the J16 connector.
4. Apply a DC bus power supply to input terminals marked + and –. Limit the maximum input voltage to 48-V DC.
5. Connect multimeter, oscilloscope probes, and other measurement equipment to probe or analyze various signals and parameters as desired. Only use appropriately-rated equipment and follow proper isolation and safety practices.

4.2.2 Software Setup

Download and install the Code Composer Studio™(CCS) IDE which is needed to build and run the project. CCS version 12.5.0 or newer is required for this project. Find more details about CCS installation and use in the CCS User's Guide.

The software for this reference design is provided as part of the C2000Ware MotorControl SDK v5.01.00.00 or newer. Once the software is downloaded and installed, browse the folder for this design by going to `<SDK install location>\solutions\tidm_02017_2w_traction_f28p65x\`.

4.2.2.1 Code Composer Studio™ Project

To import the reference project in CCS, click *Project → Import CCS Projects*, and browse to `<SDK install location>\solutions\tidm_02017_2w_traction_inverter\f28p65x\ccs` and click *Select Folder*. Select the project called *tidm_02017_2W_traction_inverter_f28p65x* and click *Finish*. The project is now visible in the *Project Explorer* pane in CCS.

The *tidm_02017_2w_traction_inverter* project contains a single build configuration. The default configuration is called *Flash* and uses the *tidm_02017_2w_traction_inverter_config.syscfg* file to configure the device clock, memory, peripherals, and also to generate the linker command file. The build configuration also includes the necessary pre-defined symbols.

For evaluation purposes, new build configurations can be created. To build a new configuration, right-click on the project and select *Build Configurations → Manage* and select new.

In addition to the single-shunt current sensing mode, there are several more optional features in the software that can be enabled and disabled using predefined symbols in the project properties. The options are as follows:

- *MOTOR_FWC* to enable field-weakening control (FWC)
- *SKIP_QEP_CALIB* to skip initial position calibration. If enabled, the PWM output of the position sensor to load initial position value.

To view and edit the predefined symbols, right-click on your project and select *Properties*. Then go to the *Predefined Symbols* section of the *C2000 Compiler* options as shown in <broken link>. By default the features listed above are enabled. To disable the features, edit the symbol and append "*_N*" to the symbol name.

4.2.2.2 Software Structure

Figure 4-3 shows the general structure of the project. The device peripheral configuration is based on C2000Ware DriverLib and is generated using the SysConfig tool. The SysConfig tool also generates the clock settings and the linker command file. This enables easy migration using SysConfig. The design can be ported to a different C2000 MCU by changing the device settings in SysConfig and remapping the pin assignments, as necessary. The motor parameters, protection threshold and other settings can be changed in *settings.h* if migrating the reference design to a different motor or running the system at different conditions.

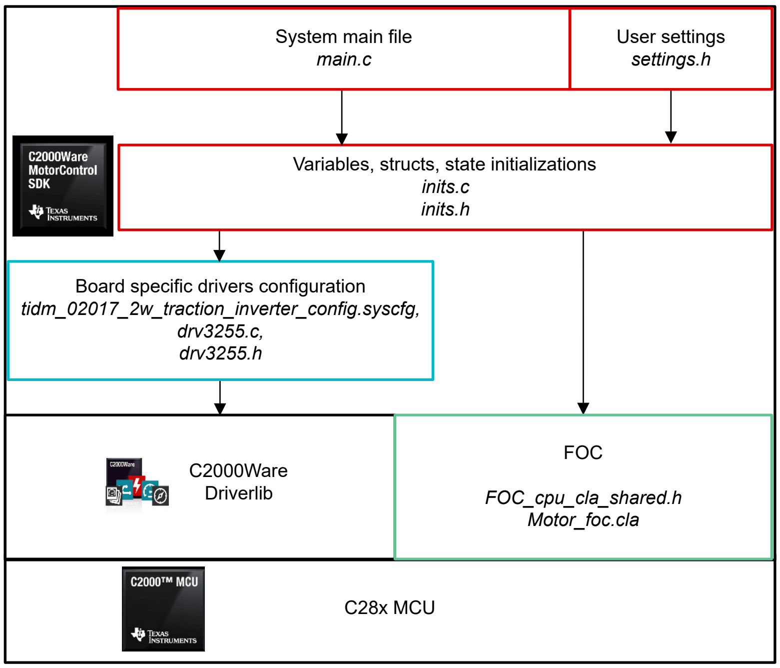


Figure 4-3. Project Structure Overview

Figure 4-4 shows the project software flow diagram. The main C28x CPU initializes the device, global variables, calibrates the ADC offset, and finally initializes the FreeRTOS scheduler. Once the scheduler is initialized, the main CPU does not run any other tasks. All tasks thereafter are run in the FreeRTOS context. In this software, there are two FreeRTOS tasks that toggle two LEDs on the control board. One of the tasks also checks the fault status flags and resets them if *clearFaults* is enabled. Add more tasks as necessary. The motor control algorithm runs exclusively on the CLA, specifically in CLA task 1. Once configured, the CLA task is triggered by the ADC End of Conversion (EOC). Data exchange between CLA and CPU happens using the CPU-CLA or CLA-CPU message RAM or the shared data RAM.

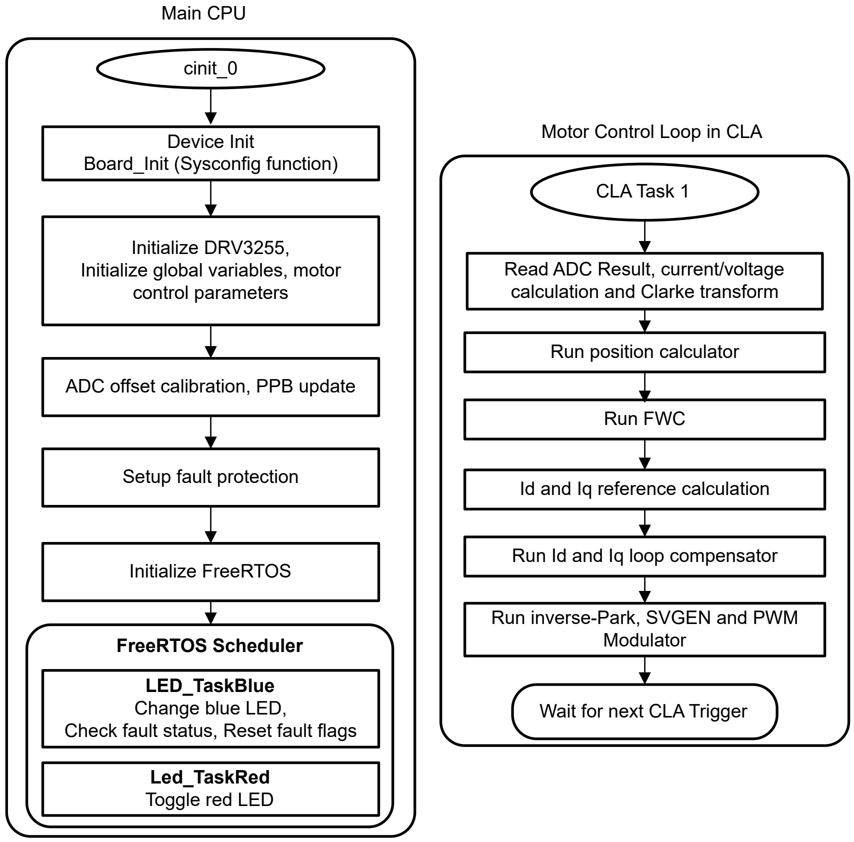


Figure 4-4. Flowcharts of Background Software and Motor Control ISR

4.3 Test Procedure

This section describes the procedure to run the setup using the software provided with the design. The designer must have previously set up the hardware and software as described in Section 4.2.

However, if the motor type and the test parameters need to be changed, the parameters are found in the *settings.h* file. The default settings can be changed by changing the value of the *USER_MOTOR* macro. If you intend to use a motor outside the list of predefined motors, copy the template used.

4.3.1 Project Setup

Import the project into CCS and select the appropriate build configuration. Right-click on the project in the *Project Explorer* and select *Rebuild Project*. Confirm that the *Console* pane shows that the project built without any errors.

On successful completion of the build, with the *tidm_02017_2w_traction_f28p65x* project selected, go to *Run → Debug* or click the *Debug* button on the tool bar. By default the project launches a debug session using the *F28P65.ccxml* file in the project. *F28P65.ccxml* is configured to use the Texas Instruments XDS110 USB Debug Probe onboard the TMDSCNCD28P65X controlCARD.

After clicking *Debug*, CCS automatically connects to the target, load the output file into the device, and change to the *CCS Debug* perspective. Halt the program at the start of main().

If the *Expressions* pane is not already open, click *View → Expressions* in the CCS menu bar. Either add variables manually or import a recommended list of variables associated with this build level by right-clicking within the *Expressions* window, selecting *Import...*, and finding the file `<SDK install location>\solutions\tidm_02017\common\debug\tidm_02017_watch_window_experssions.txt`. Click the *OK* button and the window gets populated with the variable.

Click the *Continuous Refresh* button in the *Expressions* window tool bar to configure CCS to update the data continuously at a rate defined in the CCS debug preferences.

4.3.2 Running the Application

Run the code by going to *Run → Resume* or clicking the *Resume* button in the tool bar. The project can now run and the variables display in the *Expressions* window. Check the following to confirm the application and hardware set up are working:

- The green power LED on the gate-drive boards must be on. This indicates the auxiliary power supply to the board is enabled.
- If the initializations occur correctly and the FreeRTOS scheduler is working, the red and blue LEDs are supposed to blink. If the DRV3255 gate driver is initialized correctly without any faults, the fault bits corresponding to the DRV, *faultStatusFlags.DRVnFLT1* and *faultStatusFlags.DRVFLT* is supposed to be 0.
- Similarly other variables in the *faultStatusFlags* structure show the status of other faults. If no fault flags are set, to run the test motor, the *runMotor* can be set to *runMotor*. Variables need to appear similar to what is shown in Figure 4-5.
- Start with a conservative value of *IqRef* and slowly increase *IqRef* during testing. Compare the actual I_Q feedback against reference by viewing the variables in *motor1* structure.
- If the motor does not operate as expected, try to disable *SKIP_QEP_CALIB* in the pre-defined symbols. This forces the control loop to perform initial position calibration. In this case, when *runMotor* is set, the control algorithm automatically sets *IdRef* for alignment, then set *IqRef* to spin the motor with a generated angle and finally once the index pulse of the ABI position output is detected, reset *IdRef* and *IqRef* to 0. At this point, slowly start increasing *IqRef*.
- Check the calibration offsets of the motor inverter board. The offset values of the motor phase current sensing values must be half per unit value, that is, around 0.5.
- The PWM output for motor drive can also be probed with an oscilloscope.

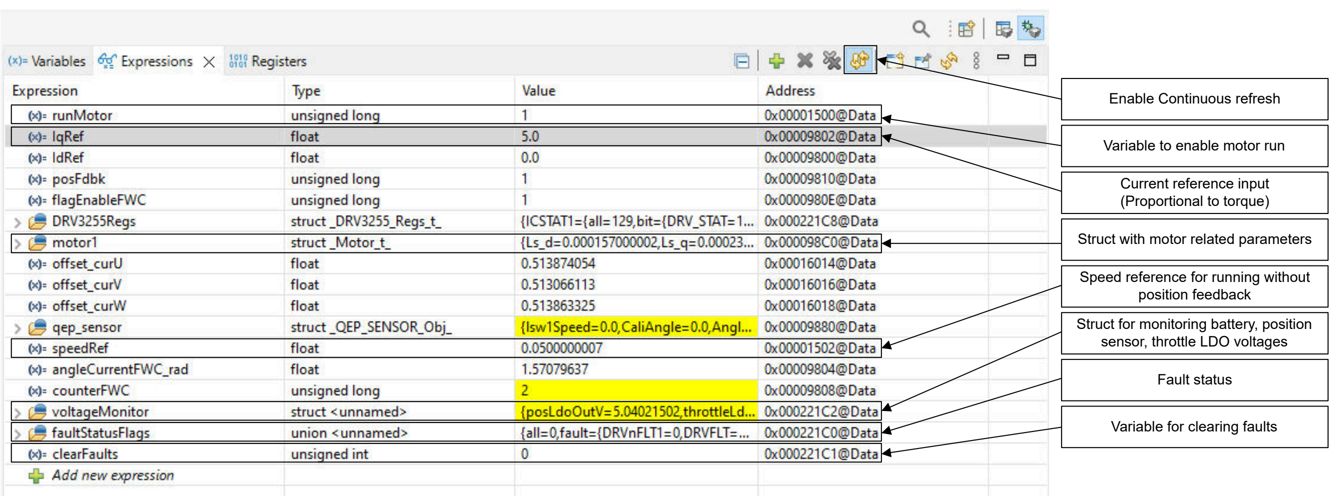


Figure 4-5. Runtime Control and Debug Through Expressions View

Halt the CPU by first clicking the *Suspend* button on the toolbar or by selecting *Target → Suspend*. To run the application from the start again, reset the controller by clicking on the *CPU Reset* tool bar button or clicking *Run → Reset → CPU Reset* and then click on the *Restart* button or *Run → Restart*. Close the CCS debug session by clicking the *Terminate* button or by clicking *Run → Terminate*. This halts the program and disconnects CCS from the controller.

It is not necessary to terminate the debug session each time the code is changed. Instead go to *Run → Load → Load Program...* (or *Reload Program...* if using the same file). If CSS detects the executable was rebuilt, CCS automatically inquires if reloading the executable is desired.

4.4 Test Results

In this section, the results of tests performed on the TIDM-02017 inverter system are presented. To test the traction inverter, the inverter was connected to a 48-V, 5.5-kW interior PMSM motor. The motor was mounted on a hysteresis dynamometer which generates mechanical load on the motor shaft. The control of the inverter, as mentioned previously, runs in torque mode. Therefore, the inverter generates the commanded torque and the speed of the motor is determined by the applied external load. This mode of testing mimics real-world operating scenarios for traction motor drives. For this test, the dynamometer operates in constant speed mode. In other words, the dynamometer generates the necessary load to regulate the speed of the motor to the set speed reference. Table 4-3 shows the results of the motor-dynamometer testing of the inverter for different reference speeds and torque (current) references. The motor shaft power measured by the dynamometer and input power measured by the DC source are also given. The test conditions and obtained power results are shown in Table 4-3. The test waveforms for the phase current and the DC bus voltage are shown in Figure 4-6, Figure 4-7, Figure 4-8, and Figure 4-9.

Table 4-3. Test Results of Traction Inverter Testing with Motor-Dynamometer Setup

REFERENCE SPEED (RPM)	CURRENT REFERENCE (A)	DYNAMOMETER OUTPUT POWER (W)	DYNAMOMETER MEASURED TORQUE (N.M)	INPUT POWER (W)
2000	15	347	1.66	480
2000	55	1590	7.4	1800
2500	55	1310	5	1550
2500	65	1614	6.1	1870
2500	95	2453	9.36	2820
3000	95	1974	6.27	2400
3000	110	2556	8.13	3044
3500	135	3120	8.78	3811
3500	160	3873	10.7	4761

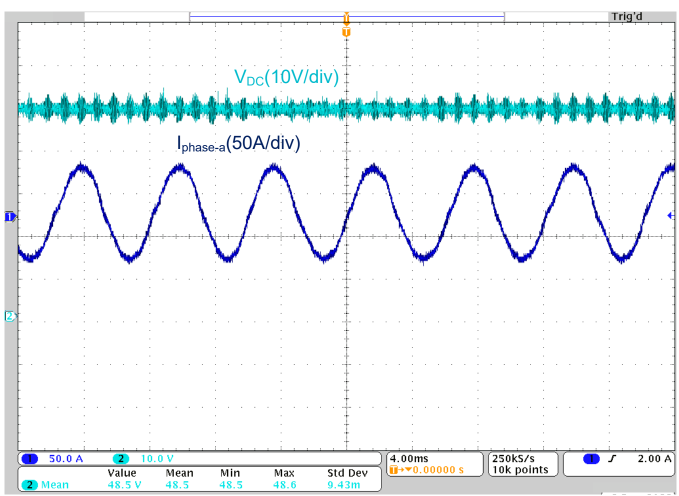


Figure 4-6. DC Bus Voltage and Phase Current Waveforms at Motor Speed = 2000 rpm, Current Reference = 55 A

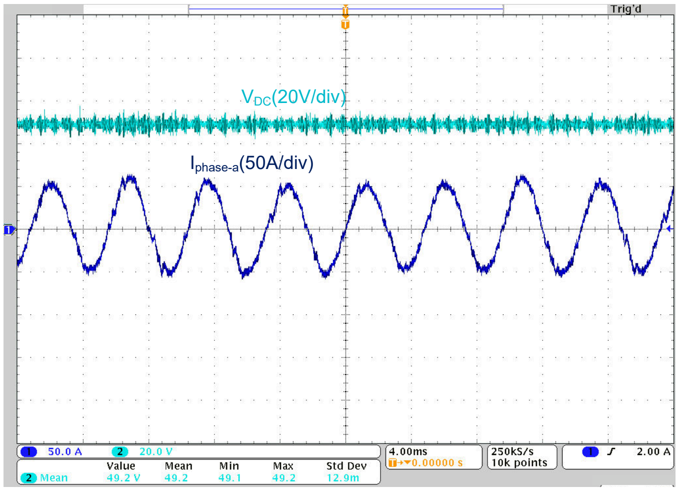


Figure 4-7. DC Bus Voltage and Phase Current Waveforms at Motor Speed = 2500 rpm, Current Reference = 55 A

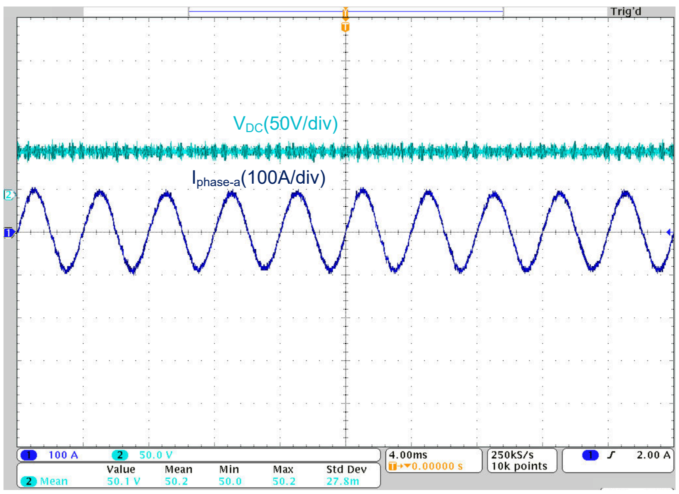


Figure 4-8. DC Bus Voltage and Phase Current Waveforms at Motor Speed = 3000 rpm, Current Reference = 95 A

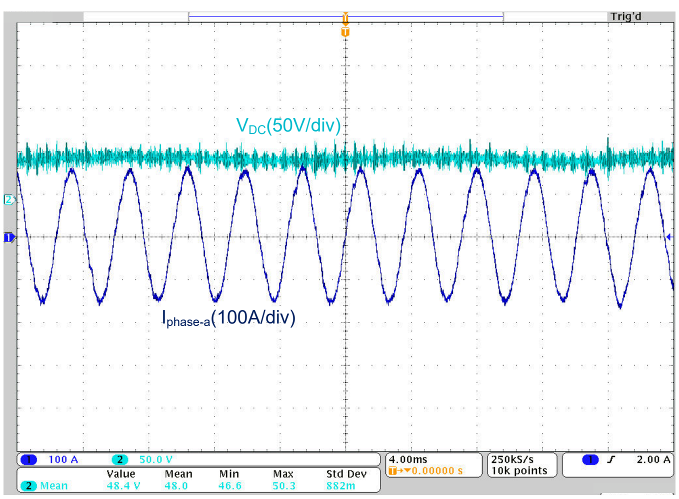


Figure 4-9. DC Bus Voltage and Phase Current Waveforms at Motor Speed = 3500 rpm, Current Reference = 160 A

5 Design and Documentation Support

5.1 Design Files

5.1.1 Schematics

To download the schematics, see the design files at TIDM-02017.

5.1.2 BOM

To download the bill of materials (BOM), see the design files at TIDM-02017.

5.1.3 PCB Layout Recommendations

5.1.3.1 Layout Prints

To download the layout prints, see the design files at TIDM-02017.

5.2 Tools and Software

Tools

TMDSCNCD28P65x — TMDSCNCD28P65X is a low-cost evaluation and development board for TI C2000™ MCU series of F28P65x devices. The board comes with a HSEC180 (180-pin high-speed edge connector) and, as a controlCARD, is an excellent choice for initial evaluation and prototyping. For evaluation of TMDSCNCD28P65X, a 180-pin docking station TMDSHSECDOCK is required and can be purchased separately or as a bundled kit.

Software

C2000WARE-MOTORCONTROL-SDK — The MotorControl SDK is a set of software, tools, and documentation designed to minimize C2000™ MCU real-time controller-based motor control system development time.

5.3 Documentation Support

1. Texas Instruments, *TMS320F28P65x Real-Time Microcontrollers Data Sheet*
2. Texas Instruments, *TMS320F28P65x Real-Time Microcontrollers Technical Reference Manual*
3. Texas Instruments, *The Essential Guide for Developing With C2000™Real-Time Microcontrollers Application Note*
4. Texas Instruments, *Enhancing the Computational Performance of the C2000™ Microcontroller Family Application Note*

5.4 Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

5.5 Trademarks

C2000™, TI E2E™, controlCARD™, Code Composer Studio™, and E2E™ are trademarks of Texas Instruments.
FreeRTOS® is a registered trademark of Amazon Web Services, Inc.
Arm® and Cortex® are registered trademarks of Arm Limited.
All trademarks are the property of their respective owners.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265