TI Designs リファレンス・デザイン
AM437x シングルチップ・モータ制御のデザイン・ガイド

TEXAS INSTRUMENTS

TI Designs リファレンス・デザイン

TI Designs リファレンス・デザインは、システムの迅速な評価とカスタム化に必要な方法、試験結果、設計ファイルなどを提供しています。開発期間の短縮に役立ちます。

設計リソース

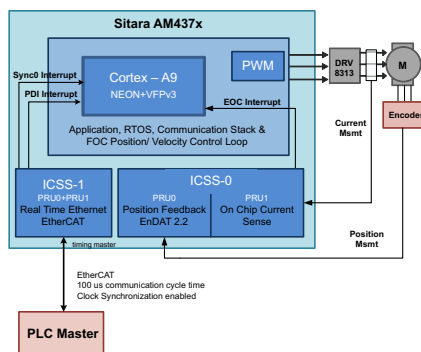|	
TIDEP0025	デザイン・ファイルを含むツール・フォルダ
Sitara_Processors	Sitara™製品情報
AM437x	AM437xのプロダクト・フォルダ

TI E2E™ Community

E2Eエキスパートに質問
WEBENCH®設計支援ツール

デザインの特長

- 産業用通信、電流/位置情報データの収集、磁界方向制御(FOC)を同時にデモンストレーション
- AM437x用のプロセッサSDK（ソフトウェア開発キット）上で動作
- 1つの永久磁石同期モータ(PMSM)の3相FOCを提供
- PRUオートメーションの下で、オンチップADCまたはシグマ・デルタ・デシメーション・フィルタを使用して、FOC駆動電流帰還を実現
- PRUオートメーションの下で、位置エンコーダへのEnDAT 2.2インターフェイスを使用して、FOC位置帰還を提供
- 外部の EtherCAT®互換マスタへの標準CiA402 Drive Profile Over EtherCAT産業用インターフェイスにより、コマンドを受信しステータスを出力
- サードパーティー製外部EtherCAT互換マスタをPC上にホストして使用し、アプリケーション制御用のコマンド・インターフェイスを提供
- 通信およびモータ制御アプリケーションでのAM437xのデモンストレーションとベンチマーク例を提供
- サポート対象の通信プロトコルと電流および位置帰還インターフェイスの実装を提供

主なアプリケーション

- 産業用アプリケーション
- 産業用通信プロトコル

1 AM437x Motor Control Application Overview

The Sitara™ AM437x family contains a Cortex A9 plus a powerful set of programmable and fixed-function peripherals for industrial communications, measurement, and control. This permits a single AM437x to replace multiple devices with an easy-to-use, compact, low-power, and cost-efficient solution.

The high-performance Cortex-A9 with NEON and VFPv3 performs the application floating point calculations using a real-time operating system (RTOS), a communication stack, and a motor-control FOC position/velocity control loop. The control loop provides drive inputs to the three-phase PWM controller to position the motor.

The AM437x Quadcore PRU performs the real-time processing for communication and data acquisition. Each of the two industrial-communication subsystems (ICSSs) contains two PRUs that perform industrial-Ethernet protocol, acquire encoder position-feedback data, and acquire ADC-current sense data. ICSS1 supports real-time industrial-Ethernet and provides system timing for all data acquisition and control.
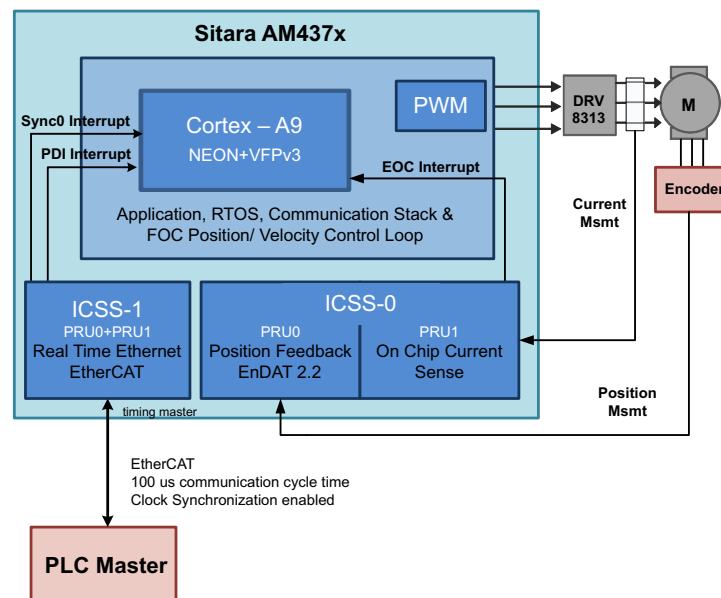

図 1. AM437x Application Overview

The ICSS system supports various industrial communications protocols such as EtherCAT, Profinet, PROFIBUS, Ethernet/IP, Powerlink, and Sercos3.

In addition to pulse-width modulators (PWM) and analog-to-digital converters (ADC), the AM437x includes a real-time clock, a quadrature-encoder pulse (QEP) module, a dual-camera interface, dual control area network ports (CANs), and dual-gigabit Ethernet interfaces. For an overview of the integrated subsystems and peripheral interfaces included in the AM437x, see 図 2.
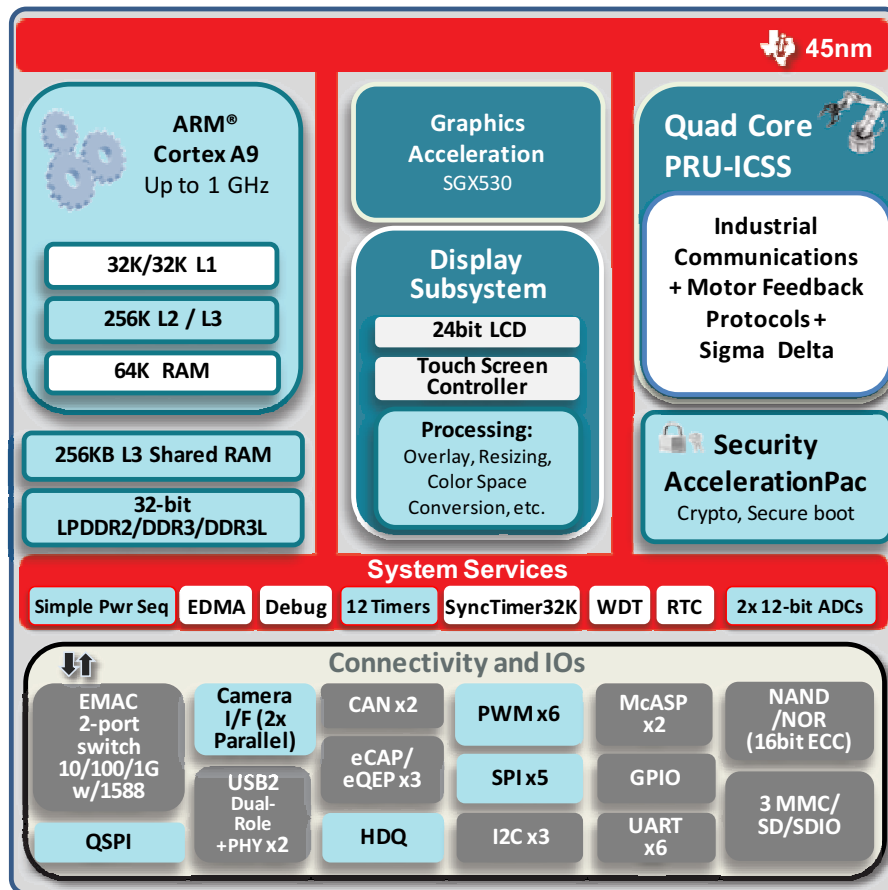
図 **2. AM437x Overview**

You can find additional information on the AM437x family at the Sitara AM437x portal page.

The application described in this document uses EtherCAT as the industrial communication protocol, and demonstrates single-chip motor control using the standard CiA402 Drive profile over an EtherCAT network.

1.1 *CiA402 Drive Profile*

The IEC standard of CiA 402 specifies a set of generic default PDOs available to all drives, as well as a set of specific default PDOs applicable only to a specific class of drives such as servo drives, frequency inverters, or stepper motors. IEC 61800-7-201 and IEC 61800-7-301 specify the CiA 402 drive profile, which is mapped to EtherCAT.

The EtherCAT Slave Stack Code (SSC) provides a sample implementation of the CiA 402 drive profile. In this application, the default implementation from the SSC is used for motor control demonstration. More details on the CiA 402 Drive profile and the stack implementation can be found in EtherCAT Slave Stack Code Application Note ET9300 and ETG6010 CiA402 Implementation Directive.

2   AM437x Industrial Development Kit Hardware

The Single-Chip Motor-Control Application uses the AM437x industrial development kit (IDK). The IDK is an application development platform to evaluate the industrial communication and control capabilities of Sitara AM4379 and AM4377 processors. For an overview of the AM437 IDK board and hardware configuration, see 図 3. For an overview of the interfaces that support run-time operations, see the interfaces highlighted in green in 図 3 and 図 4.
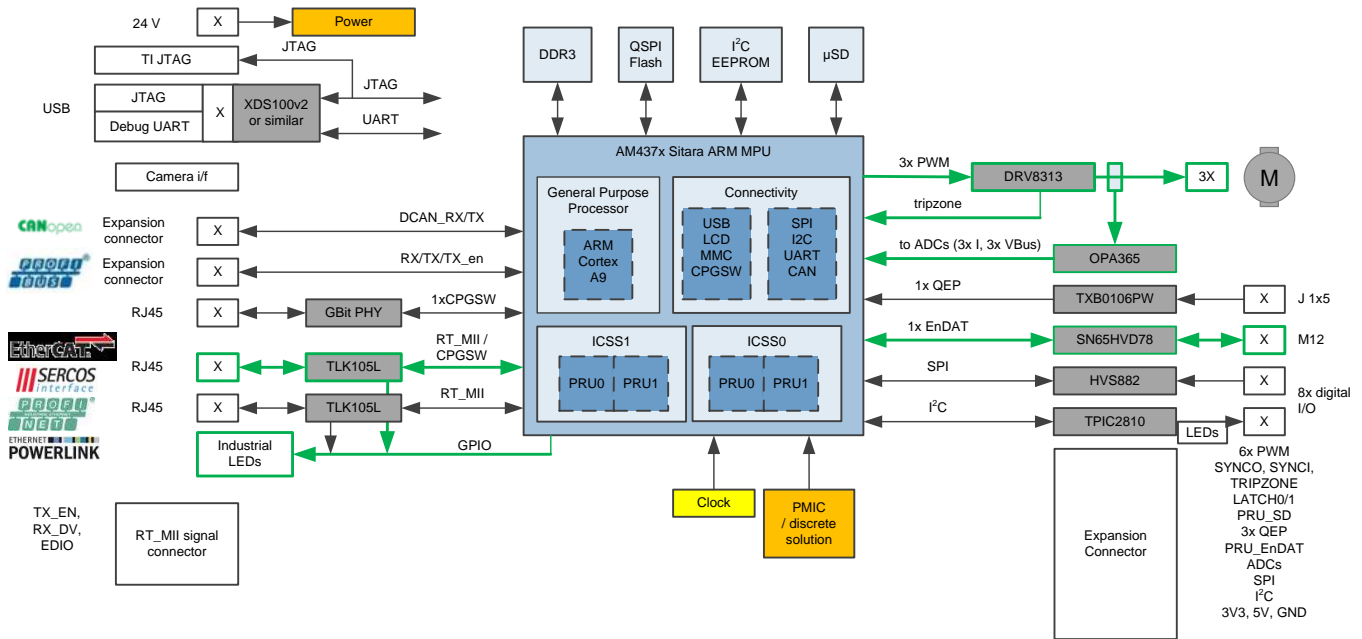
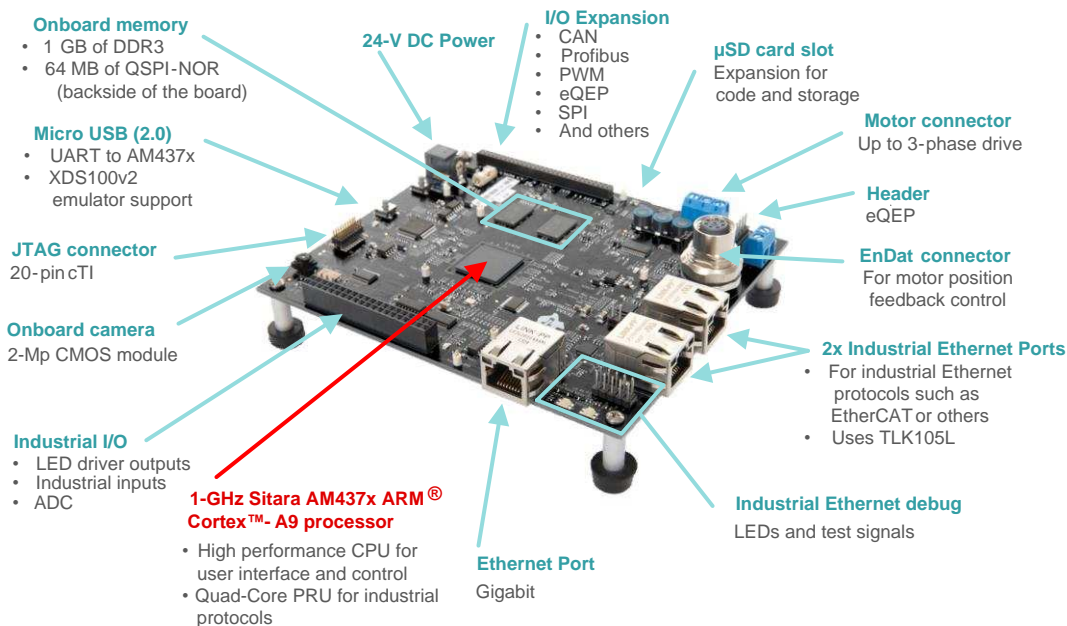**図 3. AM437x Industrial Development Kit**

図 4. AM437x Industrial Development Kit (IDK) Board Configuration

The application demonstrates a sensored three-phase, sensored field oriented control (FOC) for a single PMSM using the on-chip AM437x ADCs. The EnDat 2.2 master interface provides position information with a EnDAT encoder attached to the motor. The motor-control application has been validated using a permanent-magnet motor (BLY171D-24V-4000, Anaheim Automation). The motor can be coupled to an EnDat 2.2 encoder (ROQ 437, Heidenhain) to provide position information to the FOC algorithm. The IDK uses the three terminals of J17 to provide the motor drive.

The EVM uses three connections to the motor:

- Connect the red wire from the motor to J17-1.
- Connect the black wire from the motor to J17-2.
- Connect the yellow wire of the motor to J17-3.

Couple the encoder to the longer shaft of the motor. Insert the M12 connector of the encoder into the IDK J10 board connector. For an overview of the motor and encoder connections for AM437xIDK, see 図 5.

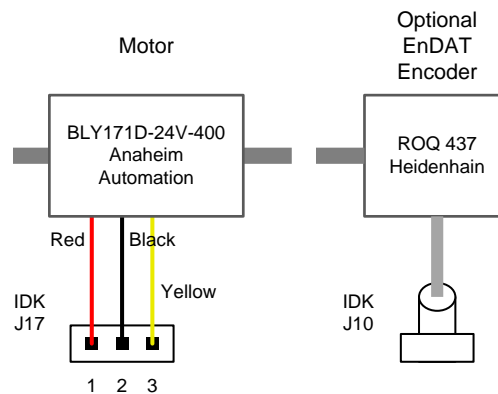図 **5. Motor and Encoder Connections to AM437x IDK**

For additional information on the AM437 IDK, see the AM437x IDK EVM HW User Guide.

The application uses EnDAT 2.2 encoder as the default encoder, but also provides a build time option to use other encoders such as Tamagawa encoder and Hiperface DSL encoder. Refer to the TI Designs for EnDAT 2.2 Encoder Master, Tamagawa Encoder Master, and Hiperface DSL encoder Master on the AM437x IDK for more details on the hardware setup

3 Software Requirement

Processor SDK RTOS for AM437x:

Processor Software Development Kit (SDK) is a unified software platform for TI-embedded processors providing easy setup and fast out-of-the-box access to benchmarks and demos. All releases of the Processor SDK are consistent across TI's broad portfolio, allowing developers to seamlessly reuse and migrate software across devices. Developing scalable platform solutions has never been easier than with the Processor SDK and TI's embedded processor solutions.

EtherCAT Slave Package:

PRU-ICSS EtherCAT package is designed for the Sitara processor family (with PRU-ICSS IP) to enable customers add EtherCAT Slave protocol support to their system.

Industrial Drives Package:

PRU-ICSS Industrial Drives package is designed for the Sitara processor AM437x (with PRU-ICSS IP) to enable customers leverage Industrial Drives (position, current sense, and control algorithm) capabilities.

Code Composer Studio:

Code Composer Studio is an integrated development environment (IDE) that supports TI's Microcontroller and Embedded Processors portfolio.

4   SDK Setup

To build and run the single-chip motor drive, install the following the packages/software:

- EtherCAT Slave Package (version 01.00.07 or later)
- Industrial Drives Package (version 01.00.02 or later)
- Processor SDK RTOS for AM437x
- Code Composer Studio

Refer to the EtherCAT Slave Package release notes for the compatible versions of the Processor SDK RTOS and Code Composer Studio. TI recommends using only the compatible versions.

The packages install the source files and the required driver libraries. The EtherCAT Slave Package User Guide provides detailed instructions on generating CCS project files.

For the single-chip drive application, the "DRIVES_SDK_INSTALL_PATH" variable must also be updated in the projectCreate.sh/.bat script to point to the installation directory of Industrial Drives Package.

Provide the "Project_Name" argument as "ethercat_slave_cia402" to the projectCreate script to generate the single-chip drive application files. The final command will look like:

```
"projectCreate.sh/.bat AM437x arm ethercat_slave_cia402"
```

5   EtherCAT Overview

Ethernet for control automation technology (EtherCAT) by Beckhoff is a high-performance Ethernet-based architecture that overcomes many limitations of fieldbus systems. This protocol is standardized in IEC61158. EtherCAT differs from other Ethernet solutions by providing on-the-fly processing. This processing lets slave devices read data addressed to them while the frame simultaneously passes through the nodes. The EtherCAT protocol avoids the delays when an Ethernet packet or frame is received, interpreted, and copied as process data at each node. On-the-fly processing forwards a received frame while slave devices read data addressed to them. The slave devices input data as the frame passes through the node with nanosecond-level delays. EtherCAT provides high-speed low-latency communications with cycle times of ≤ 100 µs.

TI™ Sitara industrial automation products support EtherCAT technology. Additional information on TI EtherCAT solutions can be found at EtherCAT on Sitara Microprocessors.

For more information about EtherCAT technology, see the EtherCAT Technology Group website.

5.1   EtherCAT Stack

The motor-control application requires access to a third-party stack source code developed by Beckhoff and licensed by the EtherCAT Technology Group (ETG). The software is the Beckhoff Slave Stack Code 5.12. TI is currently unable to distribute the stack in source format. To build the motor-control application, download the SSC 5.12 source code from the ETG website and apply a patch provided in the EtherCAT Slave Package. Refer to section 9 of the EtherCAT Slave user guide for details on patching the EtherCAT stack and building the application.

6 Field Oriented Control (FOC) Motor-Control Application

The FOC algorithm controls the motor.

The FOC has 6 levels. Most applications use only LEVEL4 and LEVEL5. For this demonstration, LEVEL6 is used. During system bring up and tuning in this demonstration, LEVEL1, LEVEL2, and LEVEL3 are used. All levels in this demonstration use a closed current loop.

For descriptions of each level, see the following list:

LEVEL3— Open-speed loop

LEVEL4— Open-speed loop, closed-speed loop with encoder feedback

LEVEL5— Position control with encoder feedback

LEVEL6— A combination of LEVEL4 and LEVEL5. You can select between them during run-time.

CiA402 Drive Profile defines Modes of Operation for the servo drives.

LEVEL 4 is equivalent to cyclic synchronous velocity mode (csv).

LEVEL 5 is equivalent to cyclic synchronous position mode (csp).

LEVEL 6 is a combined mode which can switch dynamically between csv and csp modes.

Refer to the ETG6010 CiA402 Implementation Directive for more details.

6.1 Position Offset Compensation and PI Tuning

As reported by the encoder at zero electrical angle of the motor, the position value may be something other than zero. The position offset compensation value helps correct this value. The position-offset compensation between the motor and the encoder have different values with a different relative angular position between them.

The FOC algorithm uses several PI controllers. For a setup, tune the parameters of current, speed, and position PI controller.

The Industrial Drives User Guide explains the procedure for position offset compensation and PI tuning.

7 Running the Single Chip Motor-Control Application With TwinCAT 3 EtherCAT Master on a PC

To test the single-chip motor-control application, use TwinCAT or another compatible EtherCAT master with CiA402 Drive profile support.

Additional details on TwinCAT configuration can be found at Configuring TwinCAT For TI EtherCAT Slave.

To use TI slave with TwinCAT, do the following:
1. Install TwinCAT3. (A one-month evaluation is available for free download from TwinCAT3 Download.)
    a. Select the eXtended Automation Engineering (XAE) mode of installation
2. Copy [EtherCAT SDK]\protocols\ethercat_slave\ecat_appl\esi\TI_ESC.xml to <Drive>:\TwinCAT\3.1\Config\Io\EtherCAT folder.
3. Start the TwinCAT system manager.
4. Create a new TwinCAT XAE project.
    • Select New under the File menu option.
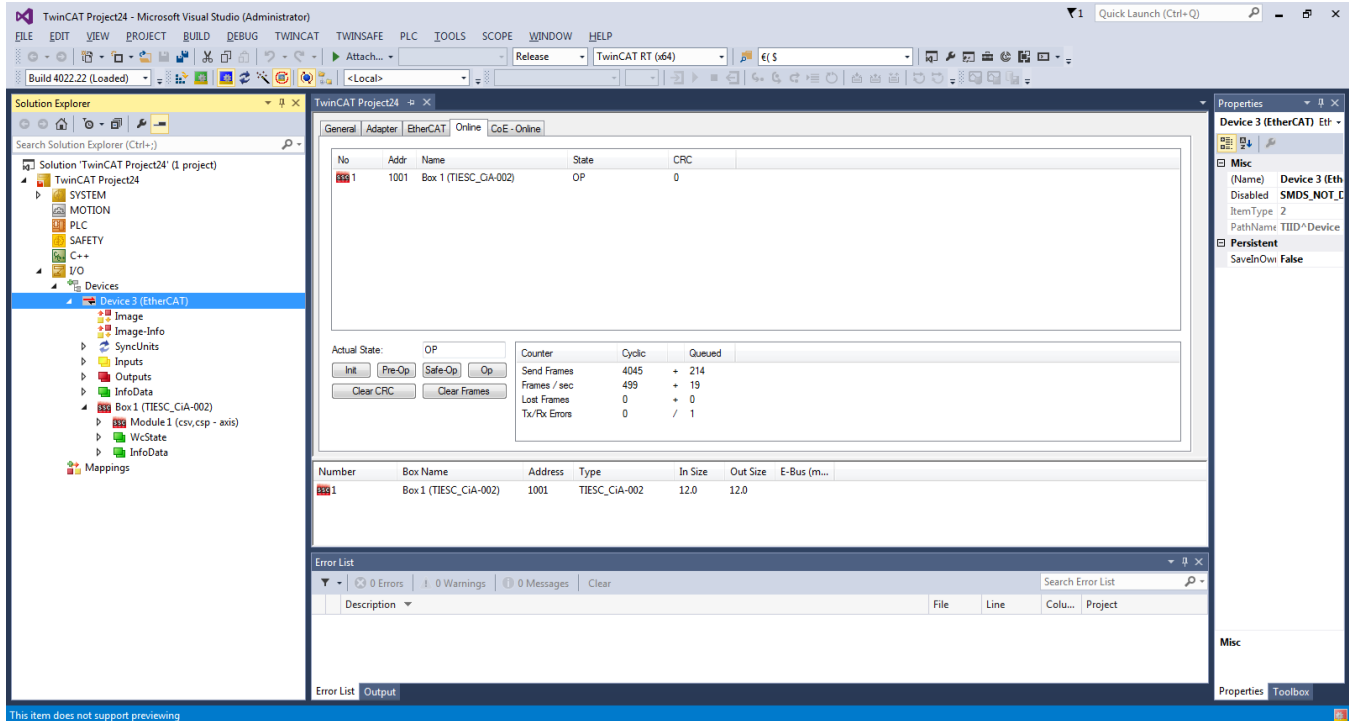   • Select Project.

- Select TwinCAT Project.
5. Navigate to Options.
6. Select Show Real-Time Ethernet Compatible Devices
7. Install the TwinCAT RT Ethernet intermediate driver.

注: TI recommends to using a NIC-compatible card listed at Supported Network Controller by Beckhoff Ethernet Driver. Check that the ethernet adapter is listed under *Installed and ready to use devices* before run running the TwinCAT demonstration.

---

8. Connect the CAT5 Ethernet cable from the TwinCAT PC to the EtherCAT IN/Port0 (J6) of the IDK. (If you have multiple IDKs in the chain, connect from the EtherCAT OUT/Port1 [J9] to the Port0 of the next IDK. For the last IDK in chain, leave Port1 open.)
9. In Solution Explorer, go to a new TwinCAT project.
10. Select I/O.
11. Right-click Device1 (EtherCAT).
12. If Scan Boxes is grayed out, select Restart TwinCAT (Config Mode).

注: TI Box n (ti-esc) is automatically detected.

- If running TwinCAT 2 on Windows XP, select Device1 (EtherCAT)
  a. Navigate to TwinCAT.
  b. Select Reload Devices.
- If running TwinCAT 3 on Windows 7, select Device1 (EtherCAT).
  a. Navigate to TwinCAT.
  b. Select Reload Devices.
13. Select Scan Boxes.
14. When asked to load I/O devices, select Yes.
15. When asked to confirm to activate free run, select Yes. (TI ESC will enter OP mode.)
16. The application turns on the digital out LEDs 1 through 8. (This indicates that slave is on and in INIT state.)
   a. To display the EtherCAT device state, select a device.
   b. Select the Online tab. (The device should be in the OP state with no lost frames or Tx / Rx / Errors.)

If another mode is shown or if errors appear, reinitialize the interface by performing step 13.

図 6. TwinCAT Status UI

17. Right-click on MOTION → Select "Add new Item..."
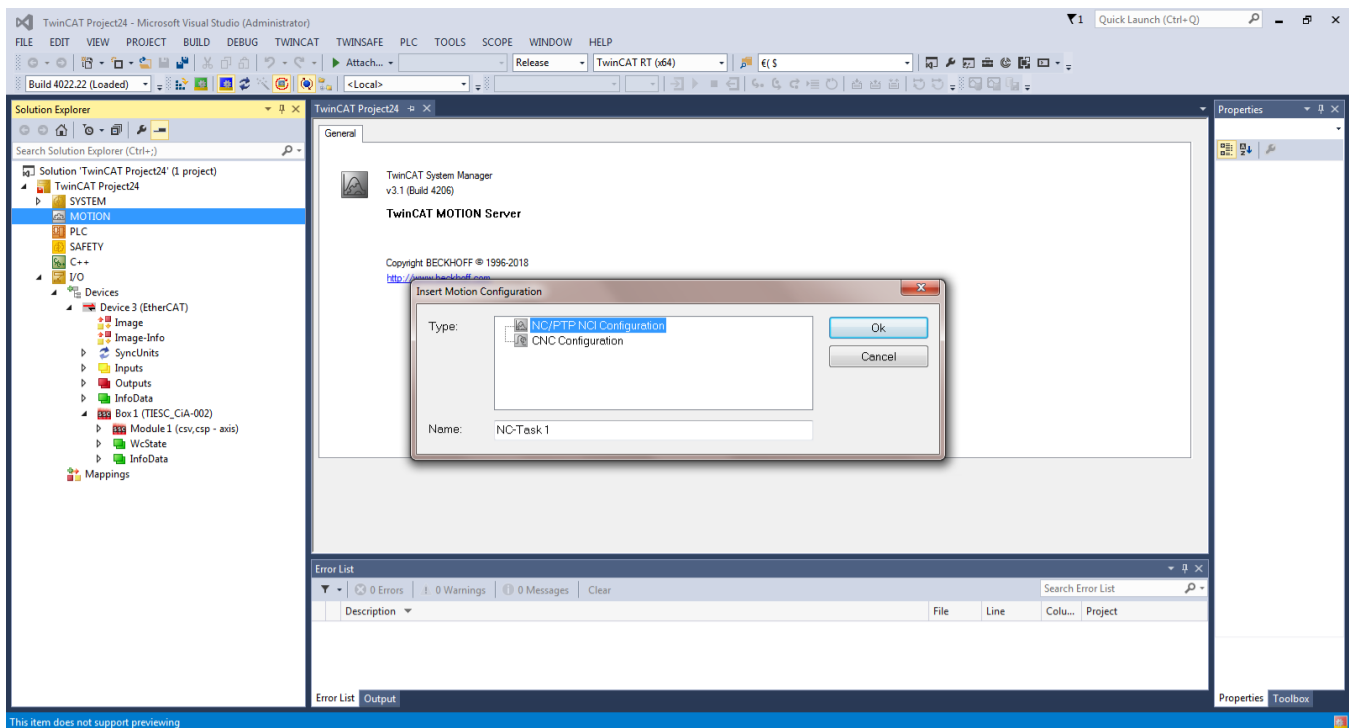18. Select "NC/PTP NCI Configuration" to put "NC-Task 1 SAF" under MOTION.

図 7. NC/PTP NCI Configuration

19. Right-click on "MOTION→ NC-Task 1 SAF→ Axes" and select Add new Item...

20. Select "Continuous Axis" Type and Click on OK. This adds "Axis 1" under Axes.

21. Go to the Settings Tab under Axis 1. Select "CANopen DS402/Profile MDP 742" as the Axis Type.

図 8. Select Axis Type

22. Now the Device variables must be linked to the axis variables, as shown in 表 1.

表 1. Axis Variables

Device	NC Axis
Box1 → Module 1→ Inputs → Status Word	Axis 1 → Drive → In → nState1 + nState2
Box1 → Module 1→ Inputs → ActualPosition	Axis 1 → Enc → In → nDataIn1
Box1 → Module 1→ Outputs → Control Word	Axis 1 → Drive → Out → nCtrl1 + nCtrl2
Box1 → Module 1→ Outputs → TargetPosition (required only in csp mode)	Axis 1 → Drive → Out → nDataOut1
Box1 → Module 1→ Outputs → TargetVelocity (required only in csv mode)	Axis 1 → Drive → Out → nDataOut2

For the Status Word and Control Word, continuous process data mapping is required. This can be performed in the linking window (for example, the Status Word link window in 図 9). Enable "All Types", "Continuous" and select the desired variables. TwinCAT maps "nState1" to the low byte and "nState2" to the high byte of the Status Word.
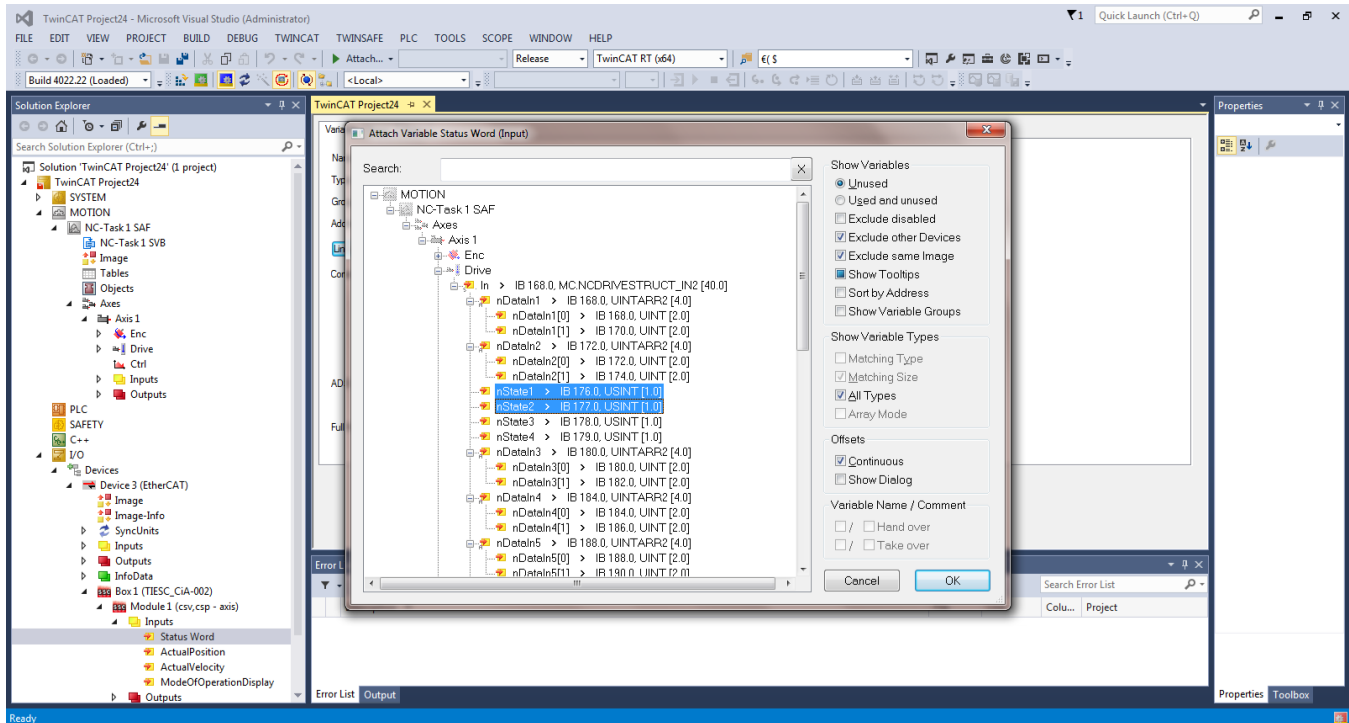

図 9. Status Word Link Window

23. Update the Operation Mode of the Device to DC-Synchron: Box 1 → DC Tab → Operation Mode → Select DC-Synchron.

24. Update the Encoder Scaling Factor: Axis 1 → Enc → Parameter Tab → Set value of "Scaling Factor Numerator" to "0.000343323".

The formula to calculate the Scaling Factor is enc.scaling = (mm/rev)/encoder resolution. In this application, the encoder resolution is set to 2^{20} = 1048576. Taking 360 mm per rev. The scaling factor is calculated as (360/1048576) = 0.000343323.
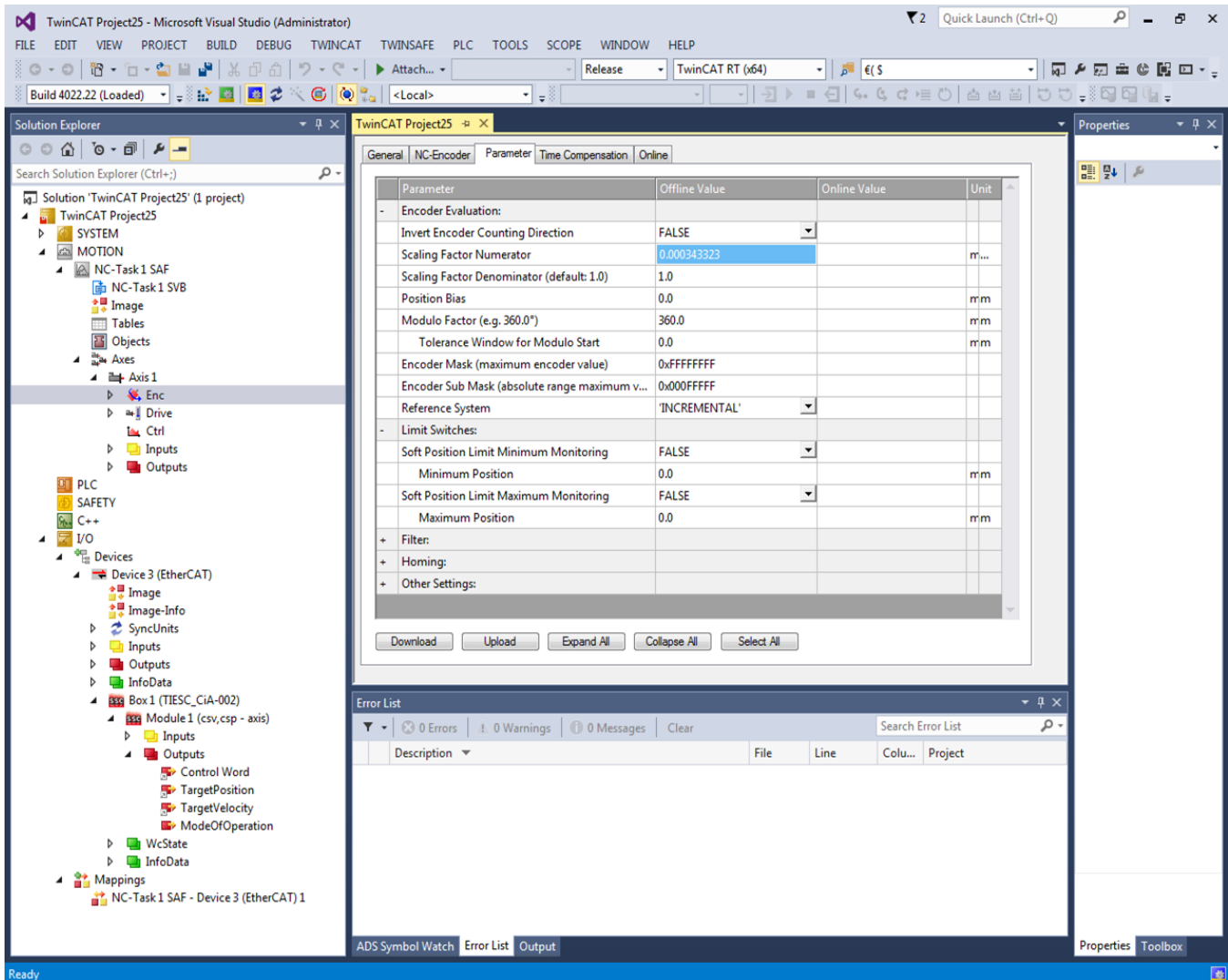
(Only required for csv mode) Update the Output Velocity Scaling Factor : Axis 1 → Drive → Parameter Tab → Position and Velocity Scaling → Set value of "Output Scaling Factor (Velocity)" to "7500.007152564"

The formula to calculate the Output Velocity Scaling Factor is velo.scaling = (2^{20} / encoder resolution) * (velo resolution / 139.81). In this application, the encoder and velocity resolution are both set to 2^{20} - 1048576. Therefore, the scaling factor is calculated as ($2^{20}/2^{20}$) * ($2^{20}/139.81$) = 7500.007152564. The encoder and velocity resolution are set as macros in the the patched cia402appl.c.

```
#define TWINCAT_SINGLE_TURN_RES pow(2, 20)
#define TWINCAT_VELOCITY_RES pow(2, 20)
```
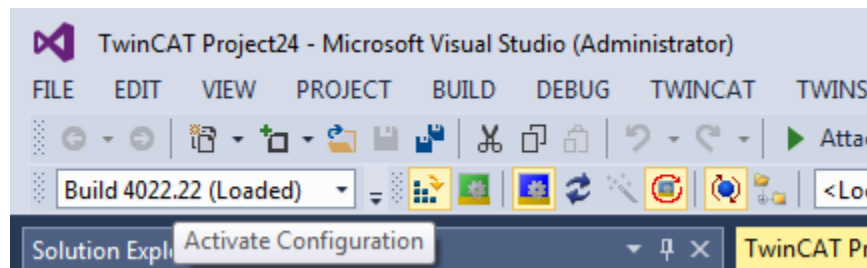
Updating these values will also require updating the scaling factors accordingly.

The formula to calculated the scaling factor is described in the *EtherCAT Slave Stack Code Application Note ET9300*

図 10. Scaling Factor

25. Start the TwinCAT System in Run Mode. Click on "Activate Configuration" to do the same.
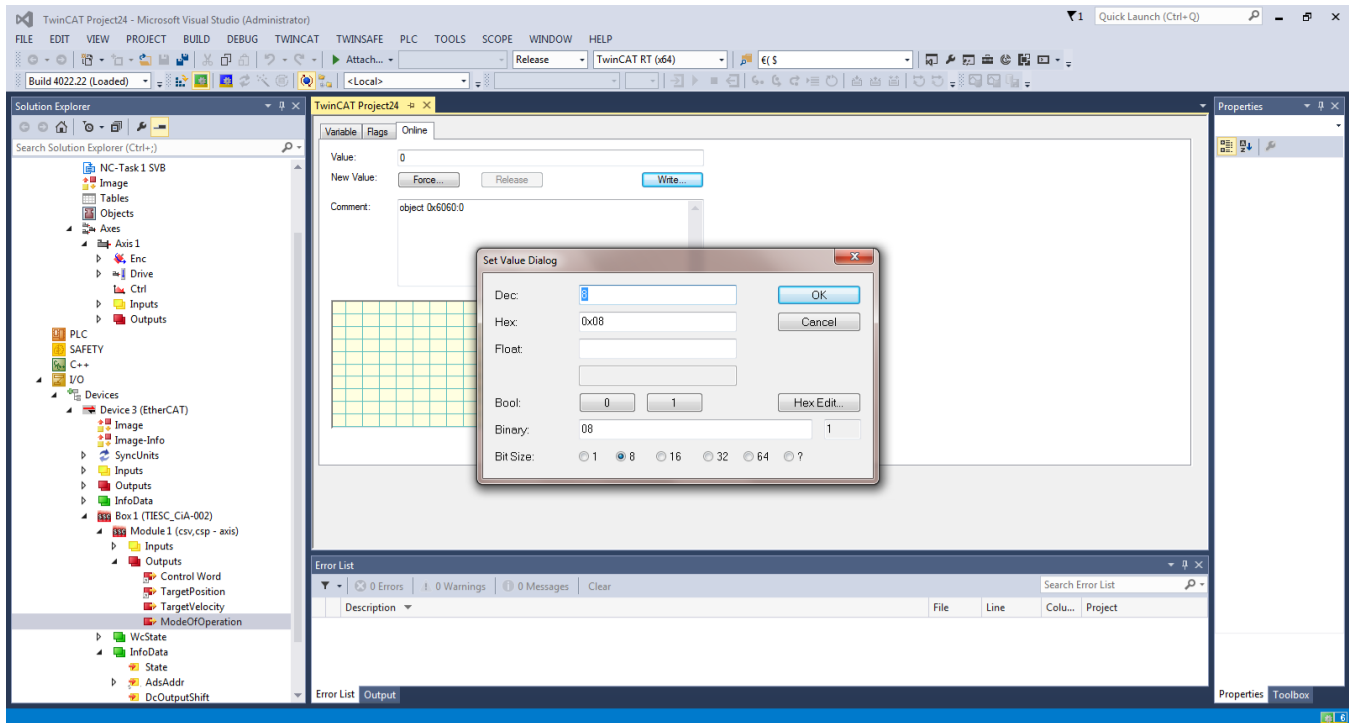

26. The default module added in the Box is the combined "csv, csp - axis". To select either csv or csp, the Box 1 → Module 1 → Outputs → ModeOfOperation variable must be set..
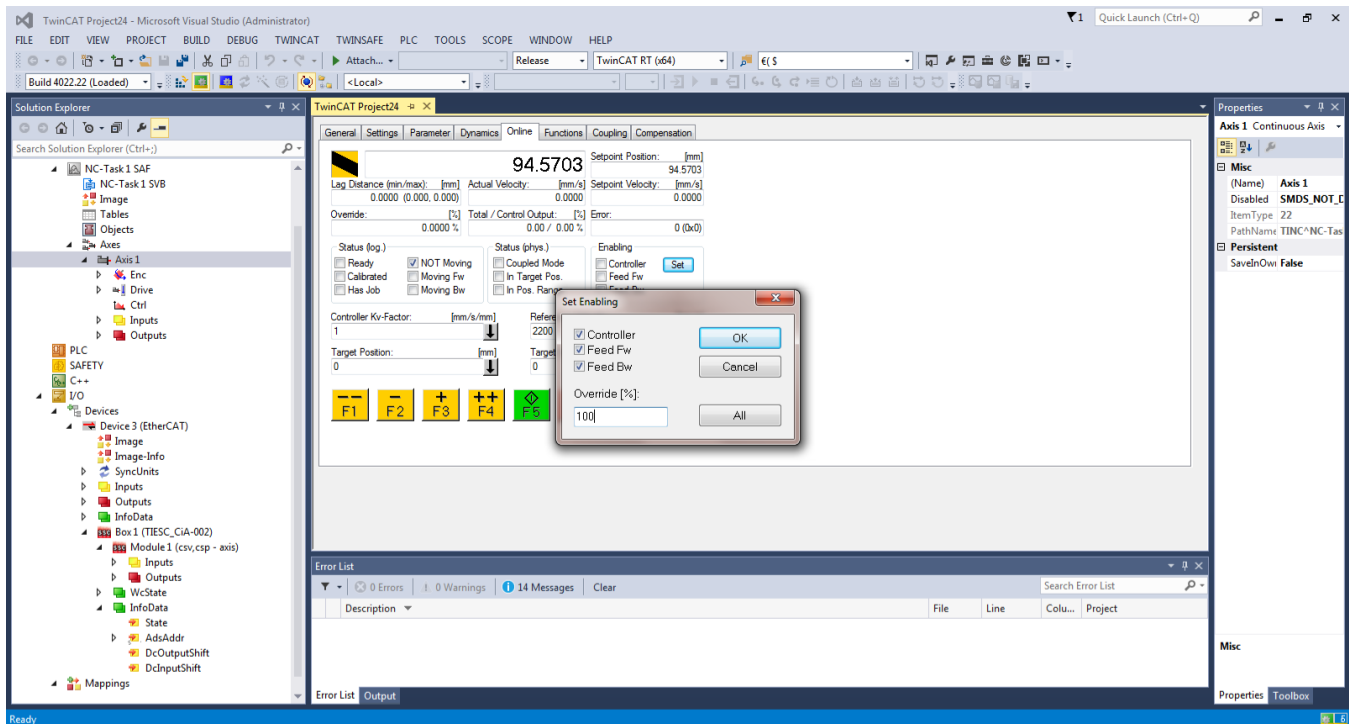
For csp → ModeOfOperation = 8

For csv → ModeOfOperation = 9

Refer to the ETG 6010 CiA402 Implementation Directive for details on the ModeOfOperation variable values.
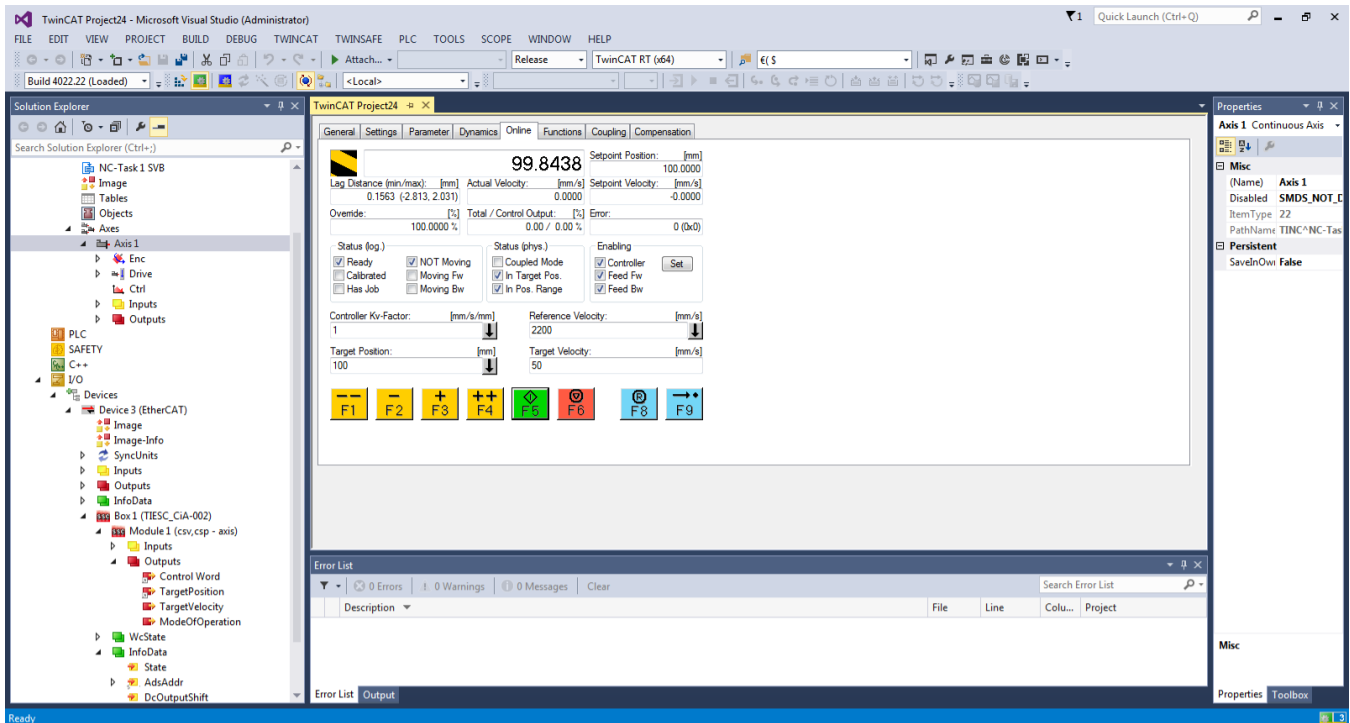
図 11. Set Mode of Operation

27. To enable the axis control, Select "Online" Tab under "Axis 1". Click on Set button under Enabling section. Tick all the options and set Override % to 100.

図 12. Enable Axis Control

28. The setup is now complete. To test the Motor Control, set the Target Position and Target Velocity. Click on Start Button (F5). The Motor turns and stops at the specified target position.

図 13. Setup Complete

8　Switching the Encoder for the Single-Chip Motor Control Application

The single-chip motor control application currently supports Endat 2.2, Tamagawa, and Hiperface DSL encoders as the position feedback source. By default, Endat 2.2 is set as the position feedback encoder. To build the application with a different encoder, follow the below process:

1. After importing the project in Code Composer Studio, go to Project Explorer and right click on the "ethercat_slave_cia402_AM437x_arm" project. Select Properties.
2. Go to Build→ GNU Compiler → Symbols.
3. Move the desired encoder's symbol to Define symbols and move the default "ENDAT_ENCODER" symbol to Undefine symbols. For example, to use the Hiperface DSL encoder, move the "HDSL_ENCODER" symbol to Define symbols and keep the "ENDAT_ENCODER" and "TAMAGAWA_ENCODER" symbols in Undefine symbols.
4. Rebuild the application.

9　Design Files

9.1　Schematics

To download the schematics, see the design files at TIDEP0025.

9.2　Bill of Materials

To download the bill of materials (BOM), see the design files at TIDEP0025.

9.3　Layer Plots

To download the layer plots, see the design files at TIDEP0025.

9.4 *Altium Project*

To download the Altium project files, see the design files at TIDEP0025.

9.5 *Layout Guidelines*

9.6 *Gerber Files*

To download the Gerber files, see the design files at TIDEP0025

9.7 *Assembly Drawings*

9.8 *Software Files*

To download the software files, see the design files at TIDEP0025

10 Related Documentation

1. EtherCAT Slave Stack Code Application Note ET9300
   (https://www.ethercat.org/en/products/54FA3235E29643BC805BDD807DF199DE.htm)
2. ETG6010 CiA402 Implementation Directive
   (https://www.ethercat.org/en/downloads/downloads_733ABB98E11545EA901D80D9A4CA7F80.htm)
3. https://www.can-cia.org/can-knowledge/canopen/cia402/
4. IEC 61800-7-201, 61800-7-301
5. EnDAT 2.2 Encoder Master TI Design (http://www.ti.com/tool/TIDEP0050)
6. Tamagawa Encoder Master TI Design (http://www.ti.com/tool/TIDEP-0101)
7. Hiperface DSL encoder Master TI Design (http://www.ti.com/tool/TIDEP0035)
8. EtherCAT Slave TI Design (http://www.ti.com/tool/TIDEP0001)

10.1 商標

Sitara, TI, Keystone are trademarks of Texas Instruments.
EtherCAT is a registered trademark of Beckhoff Automation GmbH, Germany.
すべての商標および登録商標はそれぞれの所有者に帰属します。

11   About the Author

PRATHEESH GANGADHAR TK is a systems architect at TI where he manages firmware architecture and implementation of Industrial Communication/Fieldbus and Control protocols using PRU-ICSS technology available in Embedded Processor Sitara and Keystone™ product lines. Pratheesh has been with TI since 2002 and has been involved in designing and supporting products in embedded systems. Pratheesh earned his B.Tech degree from National Institute of Technology, Calicut, India

DAVID ZAUCHA is an applications engineer at TI where he supports customer applications in the Industrial Communications segment. David has been with TI since 1999 and has been involved in designing and supporting products in analog and embedded systems. David earned his BSEE and University of of Massachusetts and his MSEE at University of Rochester. David has been with TI since 1999 and has been involved in designing and supporting products in analog and embedded systems.

MANMOHAN MANDHANA is a software engineer for the Embedded Processing Group at Texas Instruments, where he is responsible for developing Industrial Communication/Fieldbus and Control protocols using PRU-ICSS technology available in the Embedded Processor Sitara and Keystone product lines. Manmohan earned his bachelor of technology in electrical engineering from the Indian Institute of Technology Bombay, India.

Appendix A

A.1 Field Oriented Control (FOC)

The FOC attempts to separately control the torque-producing and magnetizing flux components of the stator current. By decoupling control of the magnetization, the torque producing component of the stator flux becomes an independent torque control. To decouple the torque and flux, several mathematical transforms must be implemented. The FOC consists of controlling the stator currents represented by a vector. This control is based on projections which transforms a three-phase time- and speed-dependent system into a two co-ordinate (d and q co-ordinates) time-invariant system. Field-orientated controlled machines need two constants as input references: the torque component (aligned with the q co-ordinate) and the flux component (aligned with d co-ordinate).

The first set of inputs to FOC are the motor phase currents. These measurements feed the Clarke transformation module. The outputs of this projection are designated Isα and Isβ. 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 Isd and Isq components are compared to the references Isd* (the flux reference) and Isq* (the torque reference). This control structure can control either PMSM or HVPM machines by changing the flux reference and obtaining rotor flux position. When controlling a PMSM, Isd* is set to zero. The torque command Isq* is the output of the speed regulator. The outputs of the current regulators are Vsd * and Vsq *. These are the inputs to the inverse Park transformation. The outputs of this transformation are Vsα * and Vsβ *, which are the components of the stator vector voltage in the α and β stationary orthogonal reference frame. These are the inputs of the Space Vector PWM. The outputs of the Space Vector PWM block are the signals that drive the inverter.

Both Park and inverse Park transformations need the rotor flux position. In a synchronous machine, the rotor speed is equal to the rotor flux speed. This is calculated from the rotor position of the QEP encoder. The rotor flux position (θ) is directly measured by position sensor or by integration of rotor speed. The rotor position permits the variable transformation from a stationary reference frame to a synchronously-rotating reference frame. As a result of the Park transformation, the q-axis current will control torque while d-axis current is forced to zero. The torque and flux decouple from each other. To view the overall block diagram of this project, see 図 14.
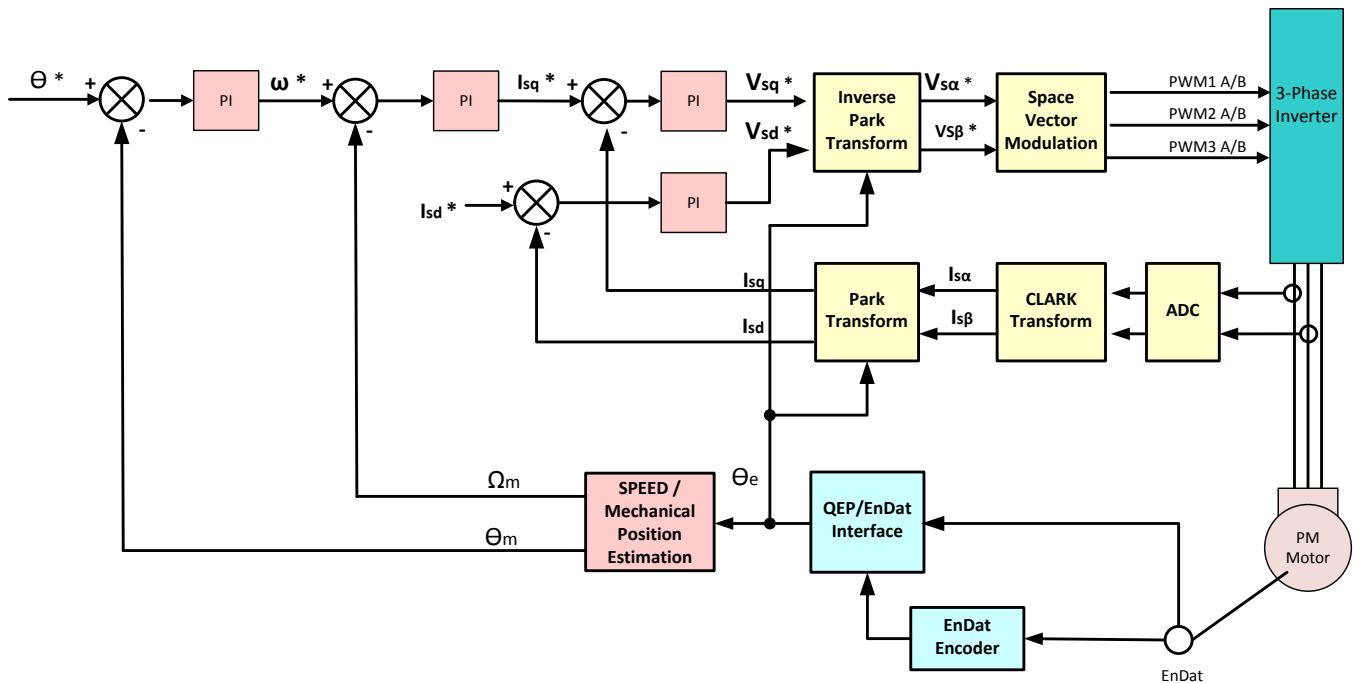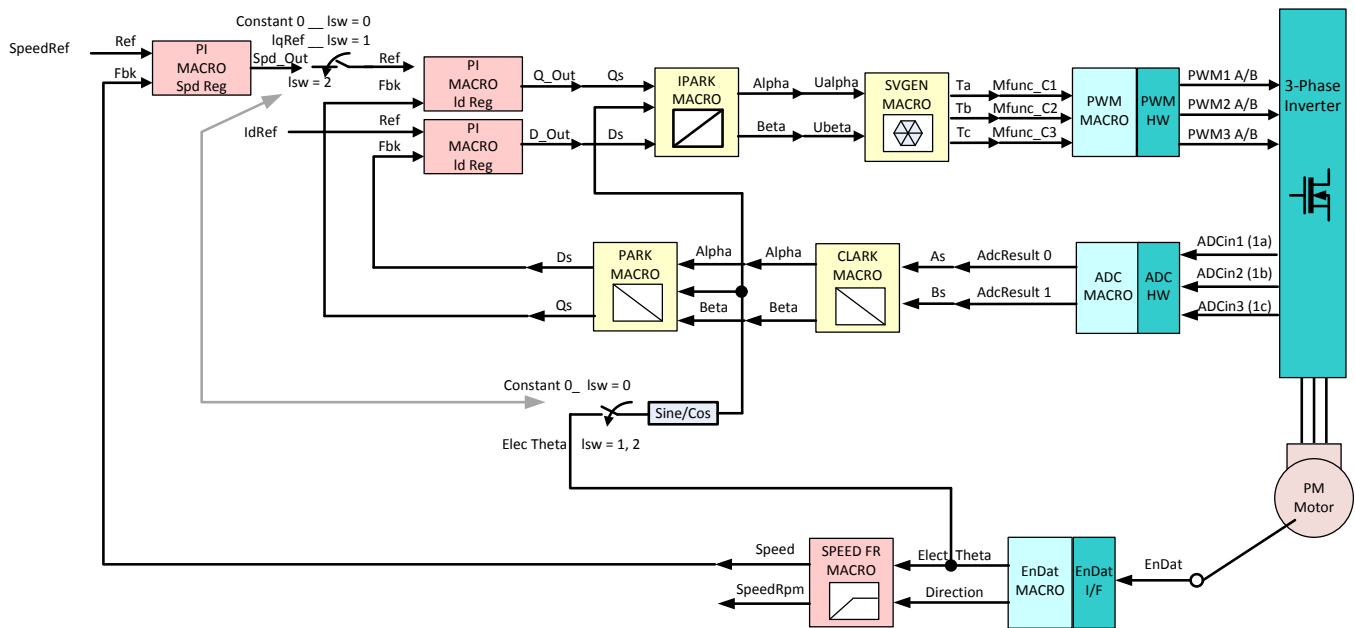
図 14. Block Diagram of Sensored Field Oriented Control
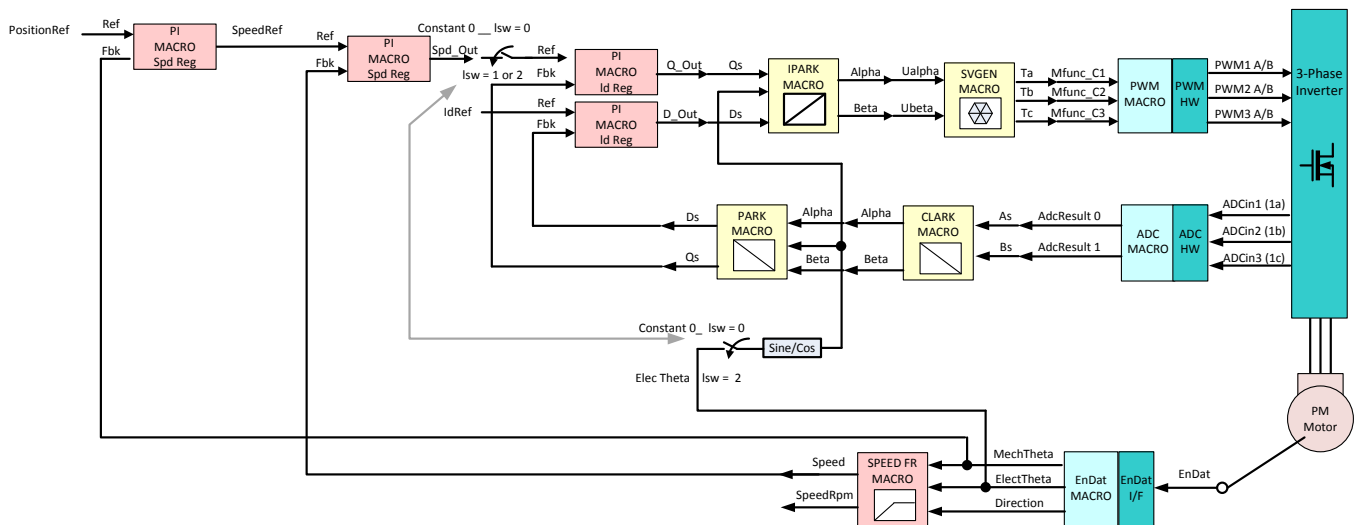
A.2   Position and Speed Motor Control

To view a diagram of the speed and position FOC loop for a PMSM, see 図 15. By reducing the system components and increasing efficiency, the AM437x family enables cost-effective integrated designs of a communications system and intelligent controllers for three-phase motors. The FOC algorithm maintains efficiency across a wide range of speeds while using a space vector modulation technique to compensate for the sinusoidal-voltage waveform applied to the motor by controlling the flux from the rotor coordinates. This approach produces a minimum amount of torque ripple when driving a sinusoidal BEMF motor with sinusoidal currents.

In LEVEL4 the controlling variables are lsw and speed.

図 15. LEVEL4 - Speed Closed Field Oriented Control Loop

In LEVEL5, the controlling variables are lsw and angle.

図 16. LEVEL5 - Position Closed Field Oriented Control Loop

LEVEL6 is a combination of LEVEL4 and LEVEL5. In Level 6, the control variables are msw, lsw, speed, and angle.

図 17. FOC Motor Control Level 6

改訂履歴B

資料番号末尾の英字は改訂を表しています。その改訂履歴は英語版に準じています。

Revision A (October 2018) から **Revision B** に変更 **Page**