Science & Technology Development Journal – Engineering and Technology, 4(3):1093-1104

Open Access Full Text Article

Research Article

EtherCAT slave controller for AC servo motor

Nguyen Hoang Giap*, Vu Ngoc Duc, Le Ba Thanh Dat

ABSTRACT

The EtherCAT network was developed by Beckhoff Automation and it is considered the fastest technology among industrial Ethernet networks. Thanks to high data transmission rates, flexible connectivity and nanosecond synchronization, EtherCAT networks are becoming more and more popular in industrial applications. AC Servo motors are becoming more and more popular in industrial applications due to their high efficiency, fast dynamic response, low noise and high precision. Servo motor control technology is constantly being developed, one of the trends is integrating real-time network into the motor controller to improve efficiency and reduce hardware costs. This paper presents the design of an AC Servo motor controller with integrated EtherCAT network. The system architecture of the EtherCAT Slave as well as the communication method between the application microcontroller and the EtherCAT Slave device are presented. Peripheral components, processing tasks of EtherCAT network-integrated motor controllers for industrial applications are analyzed in detail. Then, we propose the optimal EtherCAT controller hardware architecture both in terms of motor control and EtherCAT. The controller architecture is a combination of EtherCAT chip, high performance microcontroller and FPGA, optimized for motor control application. In addition, a prototype controller has been designed based on the proposed structure with compact size, integrated Sigma Delta ADC, USB interface, Digital I/O and support for a variety of position feedback sensors. Some important points of the CIA-402 Driver profile standard such as operating structure, state machine management, Cyclic synchronous position control mode are analyzed and applied to closed-loop control of AC servo motor. The controller is connected to the EtherCAT master and operates in a real-time environment. Test results and measurements have been checked to demonstrate the controller's operability in the EtherCAT network as well as AC Servo motor controller responsiveness.

Key words: EtherCAT, AC Servo motor, CiA402 Drive profile

Department of Control Engineering and Automation, Faculty of Electrical & Electronics Engineering, Ho Chi Minh City University of Technology, VNU-HCM, Viet Nam

Correspondence

Nguyen Hoang Giap, Department of Control Engineering and Automation, Faculty of Electrical & Electronics Engineering, Ho Chi Minh City University of Technology, VNU-HCM, Viet Nam

Email: nhgiap@hcmut.edu.vn

History

- Received: 01/3/2021
- Accepted: 15/9/2021
- Published: 30/9/2021

DOI : 10.32508/stdjet.v4i3.811

INTRODUCTION

EtherCAT[1] is an Ethernet-based fieldbus system, invented by Beckhoff Automation and has been heavily deployed in real world practice[2]. It uses standard frames and the physical layer as defined in the Ethernet Standard IEEE 802.3. EtherCAT's key functional principle lies in how its nodes processes Ethernet frames: each node reads the data addressed to them and inserts its data back to the telegram while the telegram passes through the device, the data rate speed up to 100 Mbps. EtherCAT data frames are processed "on the fly" and entirely in hardware through the EtherCAT Slave Controller (ESC). As a result, the frame is delayed only by hardware propagation delay times, helping the network reach maximum bandwidth utilization. This unique frame processing turns it into the fastest Industrial Ethernet technology.

EtherCAT provide Distributed Clocks solution High-Precision Synchronization. Since the time sent from the reference, the clock arrived at the slave devices is going to be slightly delayed. Total delays are measured and compensated for each slave device in the network continuously during operation. As a result, the system generates the "Distributed Clocks" for all slave with the jitters within much less than 1μs of each other. This solution is ideal for applications which require synchronization such as motion control, robotics, measurements, etc.

Servo motor control is a vital part of motion control application as well as robotics due to the fact that it is easily found in industrial manufacture nowadays and is available as a standard commercial product from many vendors. The motor control technology is being continuously developed, one of the trends in servo drive technology is integrating field bus communication into servo driver for higher efficiency and cost savings.

Some research on the application of EtherCAT technology have been presented previously. A slave module based on EtherCAT technology is designed from the view of the theoretical[3]. Yong-Jin Kim[4] applied the EtherCAT technology in vector control system of PMSM. Z.Liu[5] and V. Setka[6] design the motor controller with Integrated EtherCAT created for robotics.

Cite this article : Giap N H, Duc V N, Dat L B T. **EtherCAT slave controller for AC servo motor**. *Sci. Tech. Dev. J. – Engineering and Technology;* 4(3):1093-1104.

In this paper, we proposed to develop an EtherCAT Slave Controller for AC Servo Motor with more flexible structure. Our EtherCAT Slave controller consists of high performance microcontroller ARM cortex M4, FPGA and EthetCAT Slave Controller. The controller focuses on compact size, supporting multiple protocols of position feedback, integrating USB interface, Digital I/O, real-time communications and so on. Figure 1 shows the function block diagram of motor controller. With those advancements, we had developed the EtherCAT motor controller and control algorithm for AC servo motor based on CIA402 drive profile.

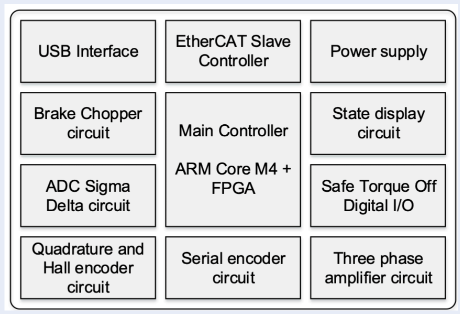

Figure 1: Function module of motor controller

This research is divided into six sections, including the introduction. Section II briefly explains the architecture of EtherCAT Slave while section III deals with overall architecture of the motor controller, the Process Data Interface between ESC and local host microcontroller (MCU) and the combination of MCU and FPGA in the motor control system. Section IV describes CiA402 drive profile in details. The results of this experiment are examined to verify the operation of the controller in section V while section VI has set the goal to point out the conclusion.

ARCHITECTURE OF ETHERCAT SLAVE SYSTEM

There are three main components in an EtherCAT slave system: Physical layer, Data Link layer and Application layer[7]. Figure 2 explains the EtherCAT slave architecture. The Physical layer contains the physical components (RJ45 connectors, magnetics, standard PHYs) to process fieldbus signals. It forwards the network data to the ESC and applies signal from the ESC to the network based on the Ethernet standard IEEE 802.3.

The Data Link Layer consists of an EtherCAT Slave Controller and EEPROM. The ESC is the EtherCAT communication processor. The ESC handles the EtherCAT protocol in real-time by processing the data frames "on the fly" and providing the interface between EtherCAT master and the slave's local application controller exchange data via the Process Data Interface (PDI). The Dual Port memory (DPRAM) in ESC is used to exchange process data and parameters between EtherCAT master and slaves local application controller. To protect data in the DPRAM from being accessed simultaneously, the ESC use SyncManager mechanism to manager access rights. The ESC uses an external EEPROM to store configuration data. The data in EEPROM includes hardware setup information will be loaded into the ESC's during power-up.

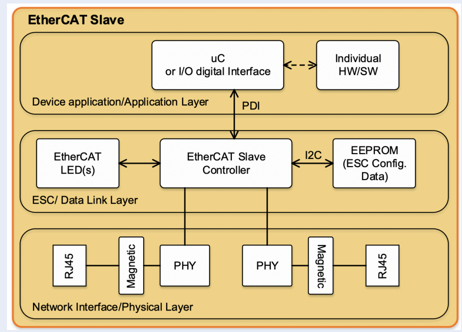

Figure 2: EtherCAT slave architecture

The Application layer is implemented on a MCU. This controller then handles the following parts: EtherCAT State Machine (ESM), PDI, Mailbox, EtherCAT application... Additional device hardware or software may also be used to implement the device functionality.

LAN9252[8] is the ESC from Microchip Company. LAN9252 is dual integrated Ethernet PHYs to process the physical layer and data link layer of EtherCAT system. It takes care of the communications infrastructure, including link control, mapping logical addresses to physical addresses, exchange data between master and local application... Figure 4 demonstrates the general functionality of LAN9252.

The LAN9252 supports 4 PDI modes: SPI, SQI, Host Bus Interface (HBI) and 16 bit Digital I/O. SPI and SQI interface provide synchronous communication between device and host system with low pin count. Whereas, the HBI facilitates the same process via a high-speed asynchronous slave interface. Digital I/O mode can be used for simple applications without microcontroller.

METHODOLOGY

In this section, we will present the hardware design architecture of motor controller with intergrated EtherCAT. The application controller is the high performance MCU STM32F407 from STMicroelectronics[9], which integrates a powerful ARM Cortex-M4 running at 168 MHz. It support both SPI for serial interface and Flexible Static Memory Controller (FSMC) for parallel memory bus interface with the LAN9252. To minimize the communication duration, the FSMC interface is utilized. Figure 3 shows the connection diagram between MCU and the ESC. In this mode, the master is MCU and slave is ESC, MCU exchange data with ESC based on 4 bits address and 16 bits data. The read/write operation is controlled via Chip select (CS), Read (RD) and Write (WR) pin. There are also 3 pins (IRQ, SYNC0, SYNC1) interrupted from the ESC to connect to the MCU to operate the EtherCAT Slave in various modes.

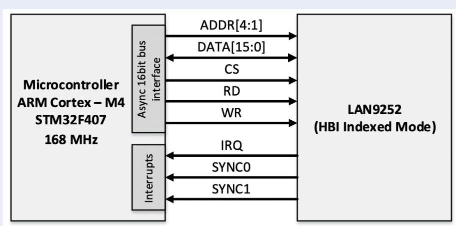

Figure 3: Connection diagram between MCU and LAN9252

The PDI operation is related to the synchronization mode of EtherCAT system. There are three modes of EtherCAT synchronization, which are Free-Run, Sync-Manager (SM) Synchronous and Distributed Clocks (DC) Synchronous respectively[10]. In Free Run mode, the EtherCAT slave is not synchronized with EtherCAT, ESC does not generate interrupt signal and MCU uses timer interrupts to access ESC periodically. In SM Synchronous mode, ESC is synchronized with EtherCAT Frame with low temporal jitter. The ESC will generate the IRQ interrupt to trigger MCU executes the PDI and application program when successfully exchanges process data with EtherCAT frame. In DC Synchronous mode, the controller event is synchronized by the SYNC event managed by EtherCAT Distributed Clock system which clock jitter less than $1\mu s$. As a results, all ESCs in network will generate the SYNC event simultaneously. Process Data must be fully processed in IRQ interrupt before the next SYNC event is received.

The EtherCAT motor controller includes motor control, EtherCAT communication and the application to manage control and communication functions.

- EtherCAT communications interface: Mapping input/ output data, Mailbox, Emergency message, EtherCAT application, etc. Although the PDI between MCU and the ESC is parallel interface, this tasks still take a considerable amount of time.
- The application task provide the high-level control of managing the overall drive communications and control functions. This can include configuration, status, operation, motion control and other management functions.
- The control portion includes the Field Oriented Control (FOC) algorithm, the current and position feedback system components which provide quiet operation and long life.

The FOC algorithm includes three control loops for current, speed, position and they are arranged in cascade structure. The innermost loop is the current (torque) control loop. The outer loop the speed control loop and the outermost loop is the position control loop, result of the outer loop is the input of the inner loop. The current loop control the torque in a servo motor by manipulating the pulse-with modulator (PWM) output, it is the most critical loop. The cycle of current loop is process synchronous with PWM frequency typically greater than 10 KHz to achieve a higher bandwidth. The complex of control algorithm requires MCU to have enough processing power at these frequencies.

The industrial driver uses isolated Sigma Delta ($\Sigma\Delta$) ADC to measure the current phases and input voltage to reduce noise and increase the resolution. The $\Sigma\Delta$ ADC convert the analog signal into a 1-bit data stream and it requires the digital filter to reconstruct the signal. There are different types of position feedback depending on the servo motor manufacturer. The simplest type is to use a hall-effect sensor embedded in the motor (usually used for electronic commutation) combined with an encoder quadrature signal. High resolution motors usually use digital data transmission protocol to encode the position feedback data, such as T-Format, SSi, EnDat, BiSS, etc. It is clear that decoding encoder protocols is a time-consuming task and very difficult for an MCU to be able to support multiple protocols without the need for additional processing modules.

In order to reduce the MCU processing time our Controller uses additional Spartan 3 FPGA from Xilinx[11].

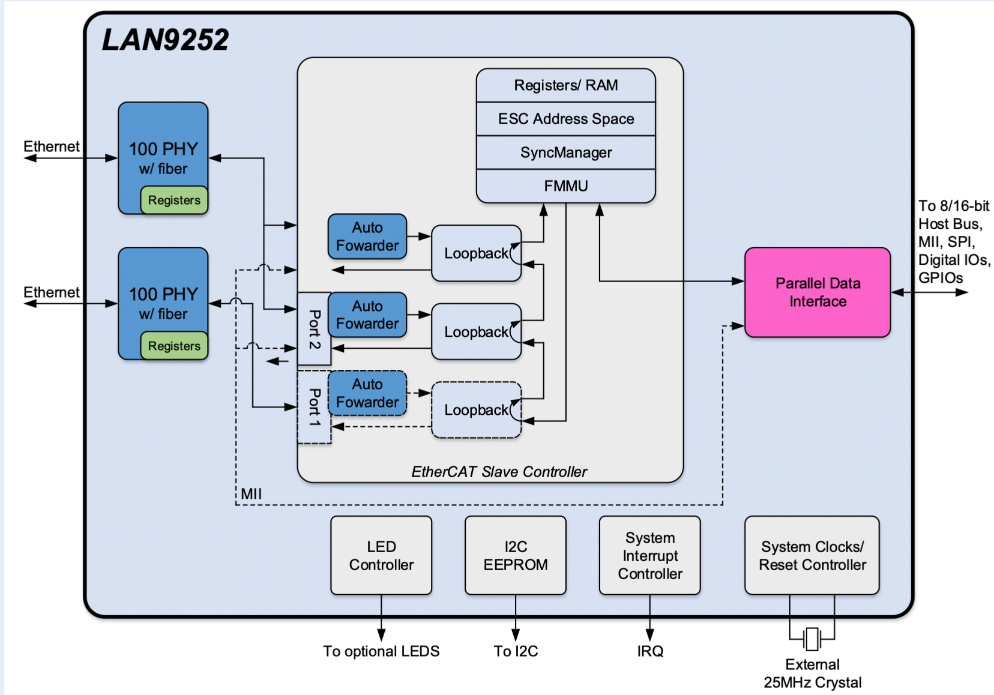

Figure 4: LAN9252 ESC block diagram

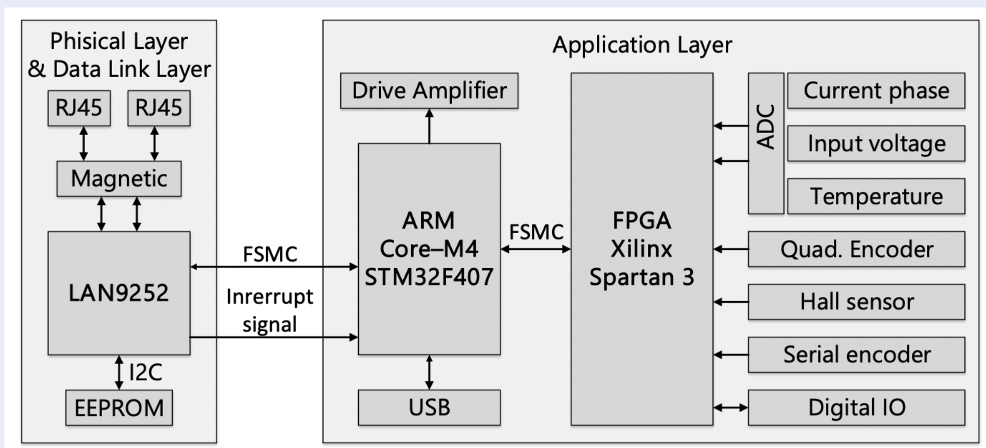

Figure 5: Overall Hardware structure of motor controller

The hardware structure of motor controller is show in Figure 5, the MCU communicates with the FPGA through parallel memory bus interfaces. In this structure, all signals from the current sensor, positional feedback and Digital I/O are connected to the FPGA. The ΣΔ filters can be easily implemented to process data from ΣΔ ADC for current phases, input voltage and temperature. For different types of position sensors, each individual module can also be built to handle each respective protocol, the FPGA design allows these modules to work independently from each other and not to rely on MCU. Then the MCU gets the current phases and positions by reading data from the FPGA, thus significantly saving computation time on the MCU. Now the MCU focuses on carrying out the remaining two tasks: the servo motor control algorithm and the EtherCAT communication.

CIA402 DRIVE PROFILE

CANopen has been development for many years and has become a standardized embedded network for many applications. The CANopen over EtherCAT (CoE) protocol allows the complete CANopen profile to be utilized in EtherCAT. The CiA402 drive profile is specified by IEC for controlling drive and motion industry [12]. It is mapped to EtherCAT to provide defines setting parameters corresponding to operating modes as object dictionaries. Furthermore, it includes the state machine that defines that defines internal and external operations in each state. In this paper, EtherCAT application layer is designed according to CiA402 drive profile. Figure 6 shows the CiA402 servo controller based on EtherCAT.

Drive state machine is an important part of CiA402 drive profile. It defines the possible states of the drive and the transition sequence between states. It also provides the ability to react to faults and to disable the drive if necessary. Figure 7 shows the state machines transition diagram. State transitions are requested by setting the object with index 0x6040 called ControlWord, details of ControlWord show in Table 1. The actual state is shown in object index 0x6041 called StatusWord, details of StatusWord register are shown in Table 2. Starting the state machine, the controller automatically switches to the NOT_READY_TO_SWITCH_ON state. After completing the EtherCAT network initialization, the state machine will automatically transition to the SWITCH_ON_DISABLED state. The high level voltage will be applied to the drive in the state of READY_TO_SWITCH_ON, however the drive operation is still disabled. SWITCHED_ON is the state that the power amplifier is ready but the drive operation is disabled all the same. The drive function shall be enabled when the state is changed to OPERATION_ENABLE. If any faults has been detect, the state will be transmitted to FAULT_REACTION_ACTIVE state to react the fault and automatically transitions to FAULT state. If the fault recovery completed, the state machine will return to SWITCH_ON_DISABLED state.

CiA402 supports various modes of operation: Position control mode (Profile position, Interpolated position, Cyclic sync position), Velocity control mode (Profile velocity, Cyclic sync velocity), Torque control mode (Profile torque, Cyclic sync torque), Homing mode, etc. In this paper, cyclic synchronous position mode is chosen to carry out the process.

Cyclic synchronous position (CSP) mode: the block diagram for CSP mode is illustrated in Figure 8.

Table 1: Controlword Register

Bit	Meaning
0	Switch on
1	Enable voltage
2	Quick stop
3	Enable operation
4 - 6	Operation mode specific
7	Fault reset
8	Halt
9	Operation mode specific
10	Reserved
11 - 15	Manufacturer specific

Table 2: Statusword Register

Bit	Meaning
0	Ready to switch on
1	Switched on
2	Operation enabled
3	Fault
4	Voltage enabled
5	Quick stop
6	Switch on disabled
7	Warning
8	Manufacturer specific
9	Remote
10	Operation mode specific
11	Internal limit active
12-13	Operation mode specific
14-15	Manufacturer specific

In this mode, the trajectory generator is located in EtherCAT master, target position is generated according to the value of the synchronous cycle time. The MCU will update the target position by reading the object 0x607A in the ESC and returning the actual position by write to object 0x6064. Due to its cascade structure, the speed command value is the proportion of the positional error, while the torque command is the result of PI speed controller and the control value of the Amplifier is the result of PI torque controller.

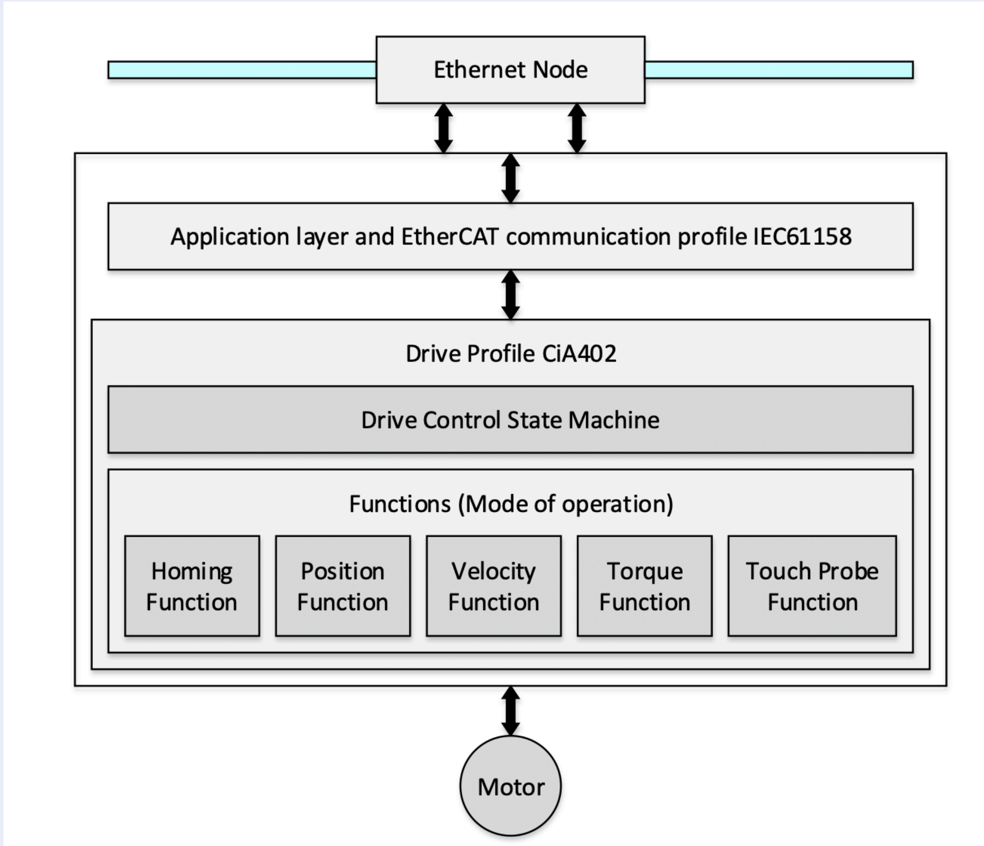

Figure 6: CIA402 servo drive profile based on EtherCAT

RESULTS AND DISCUSSION

To verify the performance of EtherCAT network, a prototype of motor controller integrate EtherCAT is designed. The controller has a compact size, integrated sigma delta ADC and USB interface while supports many types of position feedback: hall sensor, quadrature encoder, serial encoder. The Tamagawa AC servo motor with serial encoder is connected to the Driver. The controller using FOC algorithm with current control loop cycle is 100 us, speed control loop cycle 200 us and position control loop cycle 400 us.

The test system contains the EtherCAT master and the motor controller is shown in Figure 9. The master includes a PC with integrated standard network card and TWINCAT 3.1 software. There is a connection from the controller to the network card via straight Ethernet cable. The operation of the EtherCAT Slave controller is verified in both SM mode and DC mode. After successfully connecting to the EtherCAT master, the EtherCAT slave is listed in the device list and in the Operation state, Figure 11 gives data about how slave state is connected. Now the system is working in SM mode, the value of the StatusWord is 0x1221, which signifies the state of READY_TO_SWITCH_ON.

Setting EtherCAT slave controller works in DC mode with cycle time 2ms, after "Active configuration" device, now the system can be operated in DC mode and the TwinCAT motor control program is enabled. Figure 10 shows the window of motor control program. Activating the flag "Controller", "Feed Fw" and "Feed Bw" in the motor control program. Now the value of Statusword is 0x4663, indicating that the controller is in the OPERATION_ENABLE state

Verify the setting cycle time for EtherCAT Slave indicated by the signal in IRQ pin and the SYNC0 pin on LAN9252 chip are measured by oscilloscope. The cycle of each signal is illustrated in Figure 12, based on the falling edge of each signal, the period of the IRQ is 2ms and the period of the SYNC0 signal is exactly 2ms.

The communication time between MCU and ESC, MCU and FPGA is measured using the GPIO pin at the beginning/end of the task corresponding to

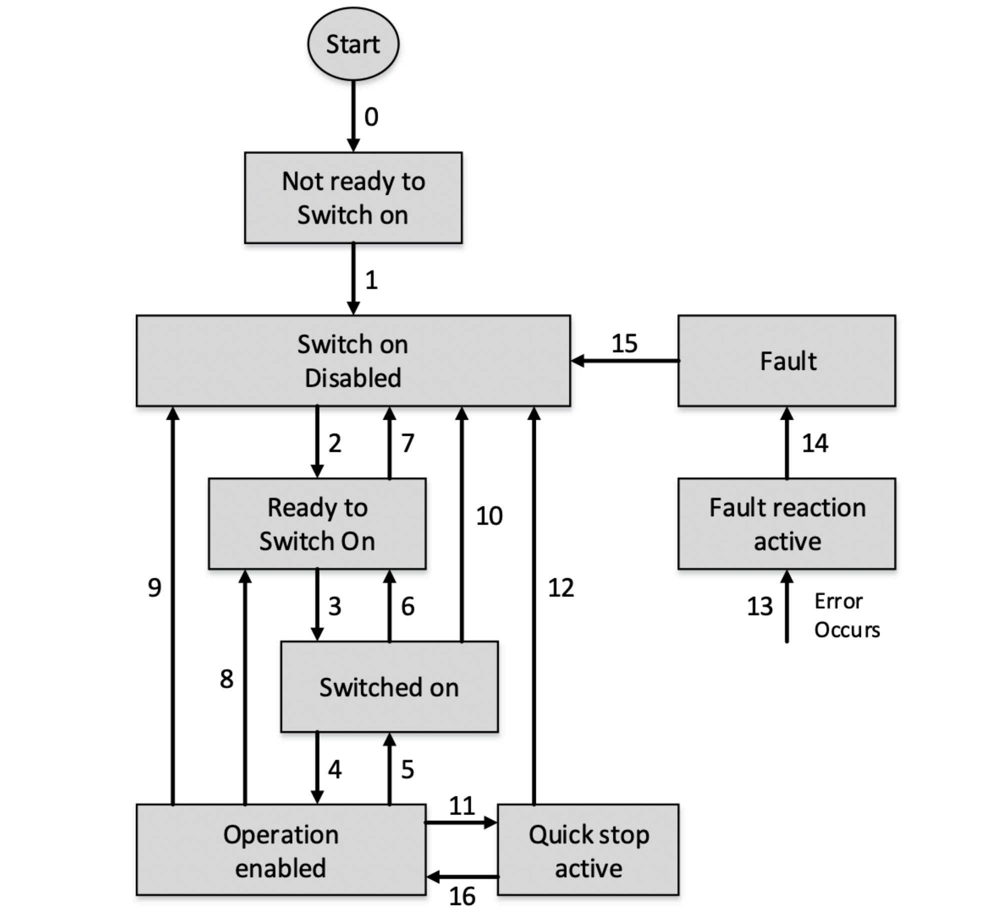

Figure 7: CIA 402 State machine of servo drive

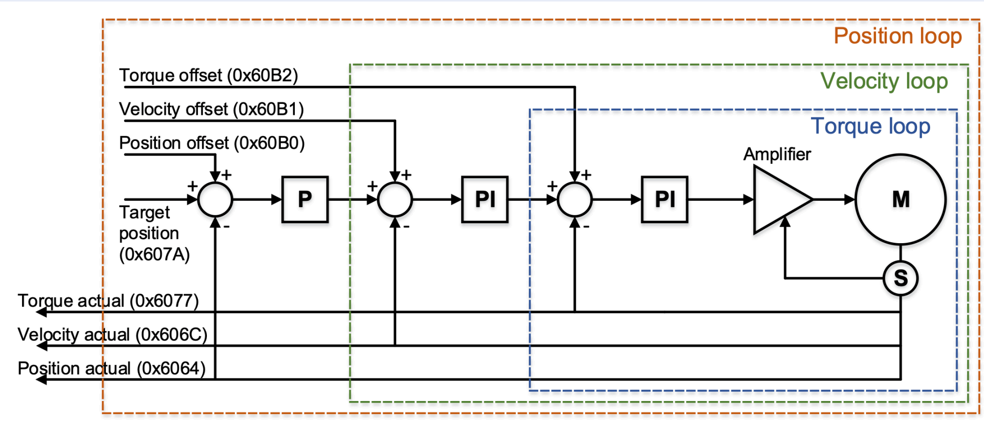

Figure 8: Cyclic synchronous position mode overview

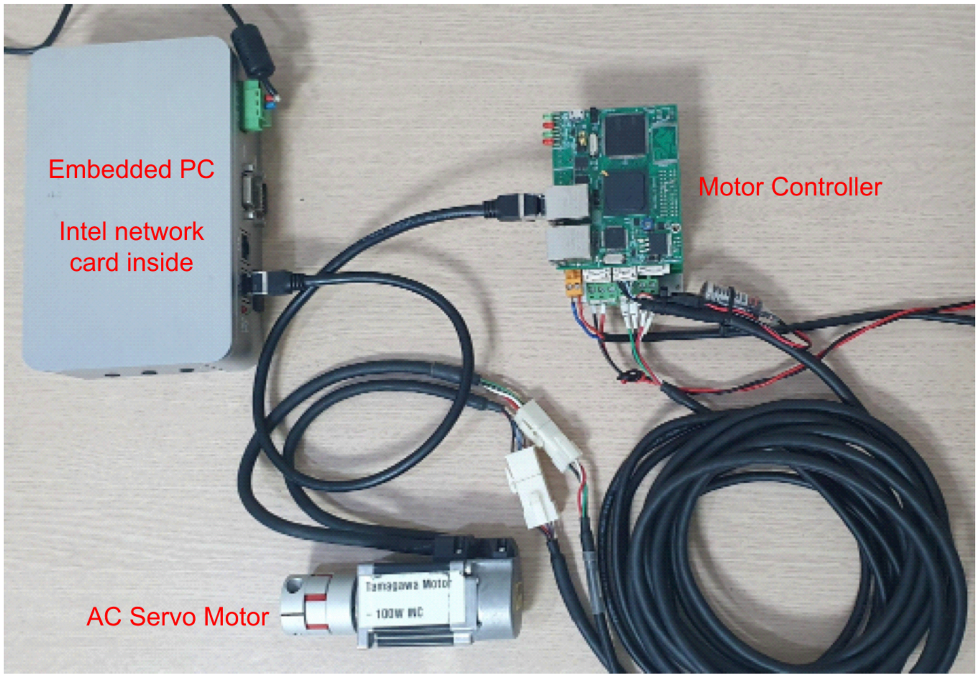

Figure 9: Motor controller prototype, Dimension 80x60x35mm

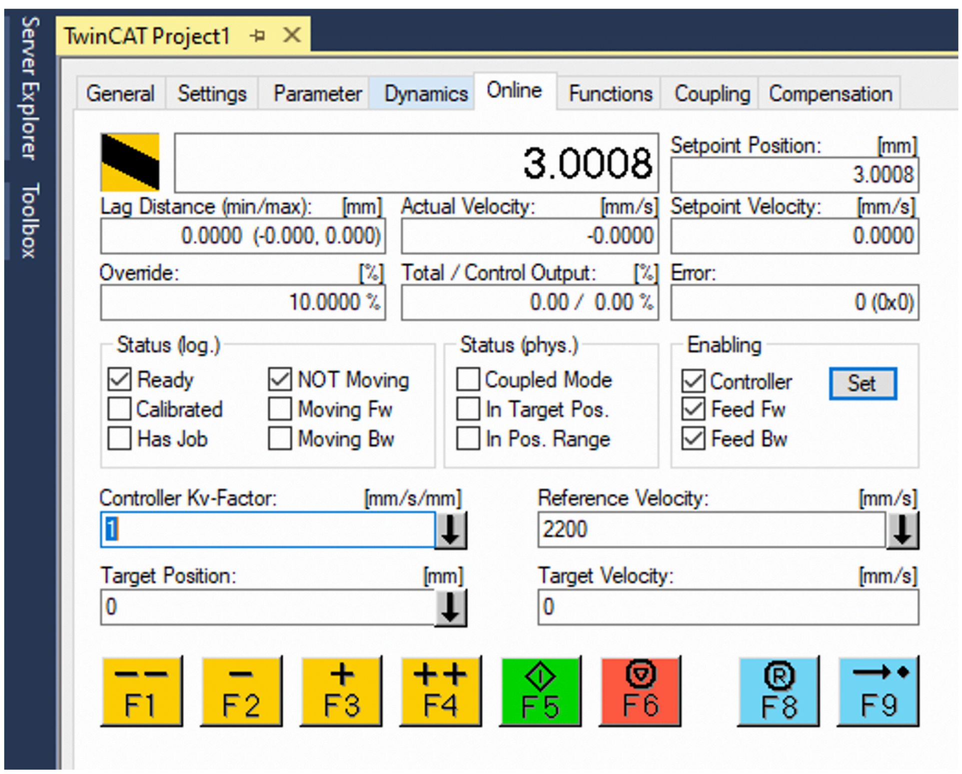

Figure 10: TwinCAT motor control program

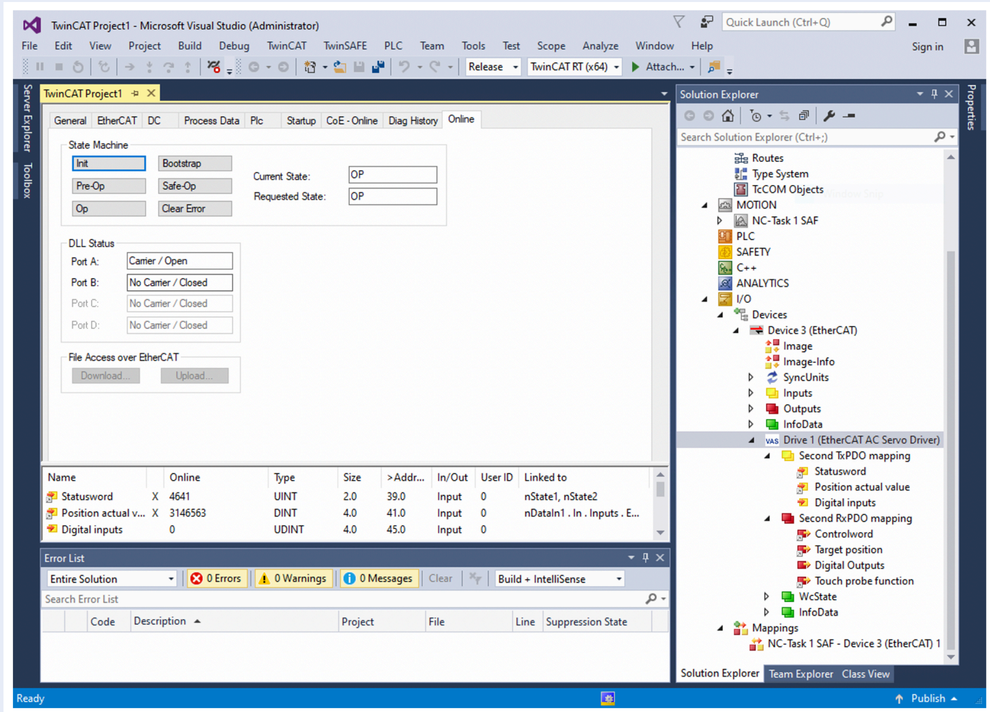

Figure 11: Device of EtherCAT slave show in this list of device

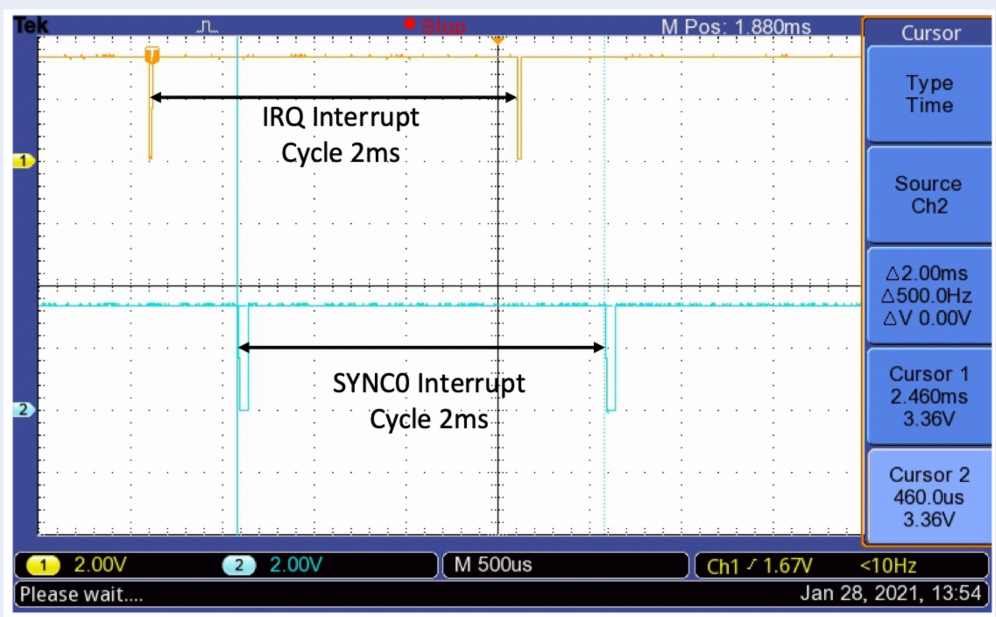

Figure 12: EtherCAT cycle time in DC mode

the set/reset state. Output waveforms were measured by oscilloscope and measurement results are summarized in Table 3. Comparing SPI and parallel memory bus interface between MCU and ESC, the communication time of parallel memory bus interface is four times smaller than SPI interface.

Finally, the results of closed loop motor control in CSP mode are verified. The data of each closed loop is recorded by the PC software and communicated with the motor controller via USB. Configure the "Reverse Sequence" on the Functions tab of the TwinCAT motor control program and then start the program. The Target Position is generated continuously by the master. The response of speed and torque closed loop is shown in Figure 13 when control motor at no load and Figure 14 when a load is applied to the motor. Due to the PI controller, the actual values respond well to the control value.

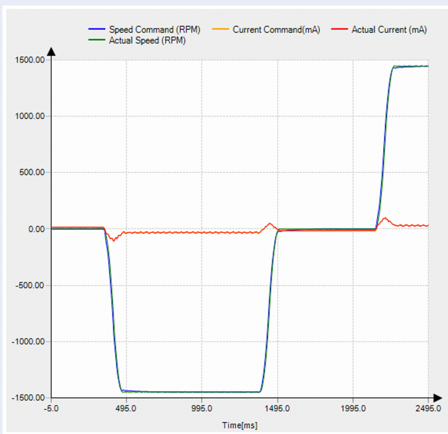

Figure 13: Speed and Torque (Current) loop response

CONCLUSIONS

In this paper, we have successfully developed the EtherCAT Slave controller for AC Servo motor. The structure and communication in the EtherCAT Slave device were presented in details. We analyzed some issues affecting the performance of the slave system and also proposed the optimal hardware structure for the AC servo motor control. Real-time functionality of the EtherCAT network, CiA402 Drive configuration, as well as controller operation are verified through testing. The motor control algorithm will be developed in the future to work optimally with EtherCAT.

ACKNOWLEDGEMENT

This research is funded by Ho Chi Minh City University of Technology –VNUHCM, under grant number T-ĐĐT-2020-16.

ABBREVIATIONS

EtherCAT: Ethernet for Control Automation Technology
MCU: Micro Controller
PDI: Process Data Interface
SM: Sync-Manager
DC: Distributed Clocks
FPGA: Field-programmable gate array

CONFLICT OF INTERESTS

The authors guarantee that there is no conflict of interest in the publication of the article.

AUTHOR'S CONTRIBUTION

Nguyen Hoang Giap comes up with research ideas and is responsible for the research and wrote the article.

Vu Ngoc Duc contributes to design hardware, firmware and the control algorithm of the controller and wrote the article.

Le Ba Thanh Dat contributes to implement the EtherCAT network and verify the controller.

REFERENCES

1. EtherCAT Technology Group. EtherCAT Introduction. 2012. [Online];Available from: http://www.ethercat.org/.
2. EtherCAT Technology Group. EtherCAT: what are the User benefits or Competitive Advantages through EtherCAT. 2011;.
3. Kang C, et al. Design of EtherCAT Slave Module. in IEEE International Conference on Mechatronics and Automation, Beijing, China. 2011;Available from: https://doi.org/10.1109/ICMA.2011.5985953.

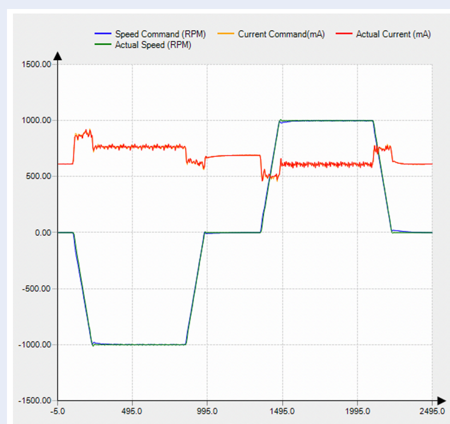

Figure 14: Speed and Torque (Current) response when applied the load to the motor

Table 3: Communcation Time Evaluation

Tasks	Parallel interface	SPI baud rate 30Mbps
Copy 20 bytes from MCU to the ESC	11.6us	61us
Copy 24 bytes from ESC to the MCU	15.6us	67us
copy 20 bytes from FPGA to MCU	800ns	-

4. Kim Y, et al. Implementation of Vector Control of PMSM by Applying Optical-EtherCAT Communication for Robot. in International Journal of Humanoid Robotics. 2014;Available from: https://doi.org/10.1142/S0219843614420092.
5. Liu Z, Liu N, et al. EtherCAT Based Robot Modular Joint Controller. in International Conference on Information and Automation. 2015;Available from: https://doi.org/10.1109/ICInfA.2015.7279562.
6. Setka V, Tolar D. Motor Controller Designed for Robotics Based on Microcontroller with Integrated EtherCAT. in International Carpathian Control Conference (ICCC), Pilsen, Czech Republic. 2018;Available from: https://doi.org/10.1109/CarpathianCC.2018.8399643.
7. EtherCAT Technology Group, EtherCAT Slave Implementation Guide. 2012;.
8. Microchip. LAN9252 EtherCAT controller with dual integrated Ethernet PHYs. Microchip. [Online];Available from: https://www.microchip.com/wwwproducts/en/LAN9252.
9. STMicroelectronics. STM32F407/417. [Online];Available from: https://www.st.com/.../microcontroller.../stm32f407-417.html.
10. EtherCAT Technology Group, EtherCAT Slave Stack Code;Available from: 2017.
11. Xilinx. Spartan-3 FPGA Family. [Online];Available from: https://www.xilinx.com/.../silicon.../fpga/spartan-3.html.
12. EtherCAT Technology Group. ETG.6010 Implementation Directive for CiA402 Drive Profile. 2014;.

Tạp chí Phát triển Khoa học và Công nghệ – Engineering and Technology, 4(3):1093-1104

Open Access Full Text Article

Bài Nghiên cứu

Bộ điều khiển EtherCAT Slave cho động cơ AC Servo

Nguyễn Hoàng Giáp*, Vũ Ngọc Đức, Lê Bá Thành Đạt

TÓM TẮT

Mạng EtherCAT được phát triển bởi Beckhoff Automation và nó được coi là công nghệ nhanh nhất trong số các mạng Ethernet công nghiệp. Nhờ vào tốc độ truyền thông cao và khả năng đồng bộ hóa với độ chính xác đến nano giây, mạng EtherCAT ngày càng trở nên phổ biến hơn trong các ứng dụng công nghiệp. Động cơ AC Servo ngày càng trở nên phổ biến hơn trong các ứng dụng công nghiệp do hiệu suất cao, phản ứng động nhanh, tiếng ồn thấp và độ chính xác cao. Công nghệ điều khiển động cơ AC servo không ngừng được phát triển, một trong các xu hướng đó là tích hợp mạng truyền thông thời gian thực vào trong bộ điều khiển động cơ để nâng cao hiệu suất và giảm chi phí phần cứng. Bài báo này trình bày thiết kế bộ điều khiển động cơ AC Servo tích hợp mạng EtherCAT. Cấu trúc hệ thống của EtherCAT Slave cũng như phương thức giao tiếp giữa vi điều khiển ứng dụng và thiết bị EtherCAT Slave được trình bày. Các thành phần ngoại vi, các tác vụ cần xử lý của bộ điều khiển động cơ tích hợp mạng EtherCAT ứng dụng trong công nghiệp được phân tích cụ thể. Sau đó, chúng tôi đề xuất cấu trúc phần cứng bộ điều khiển EtherCAT tối ưu cả về điều khiển động cơ và EtherCAT. Kiến trúc bộ điều khiển là sự kết hợp của chip EtherCAT, vi điều khiển hiệu suất cao và FPGA, được tối ưu hóa cho việc sử dụng điều khiển động cơ. Ngoài ra, một bộ điều khiển thử nghiệm đã được thiết kế dựa trên cấu trúc đề xuất với kích thước nhỏ gọn, tích hợp Sigma Delta ADC và hỗ trợ nhiều loại cảm biến phản hồi vị trí. Một số điểm quan trọng của chuẩn CIA-402 Driver profile như hoạt động máy trạng thái, chế độ điều khiển Cyclic Position Mode được phân tích và ứng dụng vào bộ điều khiển động cơ AC servo. Bộ điều khiển được kết nối với EtherCAT master và hoạt động trong môi trường thời gian thực. Kết quả thử nghiệm và các phép đo đã được kiểm tra để chứng minh khả năng thời gian thực của mạng EtherCAT cũng như hoạt động của bộ điều khiển động cơ AC Servo.

Từ khoá: EtherCAT, Động cơ AC Servo, Tiêu chuẩn CiA402 Drive profile

Bộ môn Điều khiển và Tự động hóa, khoa Điện – Điện tử, Trường Đại học Bách Khoa, ĐHQG-HCM, Việt Nam

Liên hệ

Nguyễn Hoàng Giáp, Bộ môn Điều khiển và Tự động hóa, khoa Điện – Điện tử, Trường Đại học Bách Khoa, ĐHQG-HCM, Việt Nam

Email: nhgiap@hcmut.edu.vn

Lịch sử

- Ngày nhận: 01/3/2021
- Ngày chấp nhận: 15/9/2021
- Ngày đăng: 30/9/2021

DOI : 10.32508/stdjet.v4i3.811

Trích dẫn bài báo này: Giáp N H, Đức V N, Đạt L B T. **Bộ điều khiển EtherCAT Slave cho động cơ AC Servo**. *Sci. Tech. Dev. J. - Eng. Tech.;* 4(3):1093-1104.