



General Evaluation of EtherCAT-Based Techniques in Various Industrial Systems: Review and Applications

The Tri Bui^{1,2}, Jin-Ho Shin³, and Ha Quang Thinh Ngo^{1,2}✉

¹ Department of Mechatronics, Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet Street, District 10, Ho Chi Minh City, Vietnam

² Vietnam National University-Ho Chi Minh City (VNU-HCM), Linh Trung Ward, Thu Duc City, Ho Chi Minh City, Vietnam

³ Department of Electronic Engineering, Dong-Eui University, Busan, South Korea
nhqthinh@hcmut.edu.vn

Abstract. The need for fieldbus is increasing due to the advancements in industrial control systems. Long transmission distances, high transmission speeds, and reliable real-time performance are crucial requirements for fieldbus technologies. As the demands of the current industry continue to grow, conventional fieldbus solutions struggle to keep up. Consequently, real-time industrial Ethernet technology has become more prevalent in various industries. One of the most widely adopted real-time industrial Ethernet networks is EtherCAT. Its popularity can be attributed to its excellent real-time performance, precise synchronization capabilities, support for numerous topologies, and versatile applicability. This paper aims to provide a valuable scientific reference for researchers engaged in studying EtherCAT.

Keywords: Industrial Network · EtherCAT · Real-time Network · Synchronous Control · Multi-axes Motion Control

1 Introduction

With standard physical connections and increasing speeds, Ethernet has gained widespread adoption. Industrial Ethernet, a development of Ethernet technology, involves modifications to the Media Access Control (MAC) layer. This technology offers compatibility, cost-effectiveness, high bandwidth, and flexible topology, which has led to its gradual replacement of traditional Fieldbus technology. As a result, many industrial communication protocols are transitioning to industrial Ethernet-based solutions. In [1], researchers highlight five primary industrial real-time Ethernet networks, i.e. Powerlink [2], PROFINET [3], SERCOS III [4], Ethernet/IP [5], Real-time Ethernet/IP [6] and EtherCAT [7]. Among these protocols, EtherCAT has garnered significant attention from researchers due to its determinism and real-time control capabilities.

EtherCAT, developed by Beckhoff Automation, is a real-time Ethernet network that is gaining popularity in factory automation environments [8]. While it is based on conventional Ethernet technology, EtherCAT utilizes a unique approach to access slave devices. It also defines a logical addressing scheme that allows for efficient packing of small process data, resulting in high communication efficiency and very short cycle times. This efficiency makes it an attractive solution for connecting peripherals like I/O devices and servo drives to the application master. In addition to its efficiency, EtherCAT offers another compelling feature that appeals to various application domains, especially motion control. This feature is the ability to enable synchronous operation of up to 65535 devices through a simple yet effective mechanism. The Distributed Clock (DC) technology of EtherCAT [9] ensures that local clocks in all EtherCAT devices run in synchronization. This capability is increasingly essential in control systems and adds to the appeal of EtherCAT in different industrial applications.

Given the widespread usage of EtherCAT and the numerous studies conducted on this technology in various aspects, it is indeed an opportune moment to review the state of the art in this area. Our review should focus on studies related to master and slave systems [10], the synchronization mechanism [11], and its applications [12]. Conducting a comprehensive literature review is considered an appropriate approach to identify and understand modern approaches towards EtherCAT and to explore its capabilities fully. This investigation summarizes and analyzes a large scale of theoretical researches on EtherCAT, which contain main discussions of the collected data. The rest of our works is as follows. Section 2 synthesizes the hardware and software of EtherCAT master. Then, Sect. 3 investigates the utility of various EtherCAT slave controllers. Subsequently, the algorithms and methods for synchronization mechanism are explored in Sect. 4. Some applications of EtherCAT technologies are denoted in Sect. 5. Finally, conclusions and future works are mentioned in Sect. 6.

2 EtherCAT Master Station

The EtherCAT master research can be categorized into two main directions: hardware and software. On the software side, EtherCAT is an open Ethernet protocol adhering to global standards, which has led to the development of numerous commercial solutions and open-source projects. Notably, open-source real-time Linux platforms like RT-Preempt, RTAI, and Xenomai are widely utilized in combination with the EtherCAT master stack provided by IgH, as well as with open-source EtherCAT master stacks like Simple Open EtherCAT Master (SOEM) from the Open EtherCAT Society. The adoption of an open-source-based EtherCAT master system offers several advantages, including flexibility in application program development and cost reduction. Thanks to the open-source nature of these platforms, developers can create applications in the C/C++ programming language, without being restricted by certain limitations imposed by proprietary solutions like Beckhoff's TwinCAT. For instance, TwinCAT imposes restrictions on memory allocation and the use of standard C/C++ math functions, making it challenging to develop applications with complex features, such as dynamics-based manipulator control. In contrast, open-source real-time Linux platforms provide both kernel space and user space interfaces.

Table 1. Summary of the state-of-the-art for related researches in EtherCAT master

Author(s)	Publication year	Main research	Concern	Operating System	EtherCAT-based master software	Advantage(s)	Disadvantage(s)
Shi, H. et al. [1]	2022	Hardware	EtherCAT hardware based on development board	N/A	N/A	A hardware EtherCAT master is introduced based on the ARM architecture and has a PCIe port to connect to the computer. Hardware guarantees about Hard real-time	The hardware structure is quite complex and the connection to the computer is still underdeveloped
Yi, H. C. et al. [13]	2019		Cycle time improvement	Linux with RT-Preempt	Igh Etherlab	Presents a hardware architecture under the linux kernel that does not use standard drivers for all hardware, but is developed specifically for a certain network hardware. This reduces system time and delay by eliminating memory copy and NAPI	Just show the general structure, not detail the structure of the developed Direct Ethernet Drive
Zhang, H. et al. [14]	2019		EtherCAT hardware based on development board	N/A	N/A	EtherCAT master is developed based on FPGA chip, with this structure can be developed for other circuits	The experiment has not yet clearly stated the performance of the hardware, but only tested a certain system cycle
Kim, S. et al. [15]	2017	Software	Performance analysis of open-source software	Linux with RT-Preempt, RTAI, Xenomai	Open Source-based EtherCAT Master (SOEM, Igh Etherlab)	The article shows the experiments of open-source libraries on different Linux-based real-time operating systems and compares them with TwinCAT, thereby showing the low-delay capabilities of the Linux architecture	Only shows the results of the experiments, but there are no graphs to demonstrate the system's capabilities
Shi, B. H. et al. [16]	2017		Software development based on open-source software	Linux with RTAI	Igh Etherlab	Use Igh EtherCAT library to add LinuxCNC software to control EtherCAT servo drivers for CNC control purposes	The experimental section only mentions the software and setup, not if the experimental results, figures or graphs

(continued)

Table 1. (continued)

Author(s)	Publication year	Main research	Concern	Operating System	EtherCAT-based master software	Advantage(s)	Disadvantage(s)
Alex, B. et al. [17]	2021		Software development based on open-source software	Linux without RT patch	SOEM	Develop SOEM-based extension libraries for EtherCAT control, including fastcat and jsd. Libraries provide the ability to connect variables of functions and slaves together using only YAML config files. Guaranteed zero delay even without using real-time kernel	Support a small number of devices, if you want to add the devices you want, you need to have knowledge of that hardware
Cho, S. Y. et al. [18]	2023		Software development based on open-source software	Linux with RT-patch	Igh Etherlab	Develop an extension for Python programming on real-time Linux environment based on Igh Etherlab. Experimental comparison of delay is not too different when using C language when programming real-time tasks	The system cycle is limited to a minimum of 1ms, while currently aiming for smaller send and receive cycles

N/A: Not applied

In terms of hardware, developers are often driven to develop solutions that aim to reduce latency and enhance the processing speed of the EtherCAT master. The effectiveness of control systems heavily relies on an EtherCAT master with low latency and high synchronization capabilities, enabling fast and precise execution of actions. However, as high-end control fields advance, conventional EtherCAT masters face limitations in meeting the performance demands of such advanced control systems, mainly due to real-time constraints. Therefore, the development of a new EtherCAT master with improved real-time performance becomes crucial to enable its effective utilization in cutting-edge industrial applications. Cycle time and clock synchronization are vital benchmarks for evaluating EtherCAT's real-time performance. Achieving optimal cycle times and precise clock synchronization is essential to ensure efficient operation and coordination of industrial processes. Table 1 presents various studies focusing on the EtherCAT master, along with their respective advantages and limitations. These studies aim to address the challenges posed by real-time performance in EtherCAT systems and strive to enhance the technology's capabilities for advanced industrial applications. By analyzing these studies, researchers can gain valuable insights into the progress made in EtherCAT master development and identify potential areas for further improvement.

3 EtherCAT Slave Station

An EtherCAT slave system comprises the Physical layer, Data Link layer, and Application layer. The architecture of the EtherCAT slave is depicted in Fig. 1. The Physical layer includes essential physical components such as RJ45 connections, magnetics, and

standard PHYs, which are responsible for processing fieldbus signals [19, 20]. By adhering to the Ethernet standard IEEE 802.3, the Physical layer transmits network data to the EtherCAT slave controller (ESC) and applies signals from the ESC to the network.

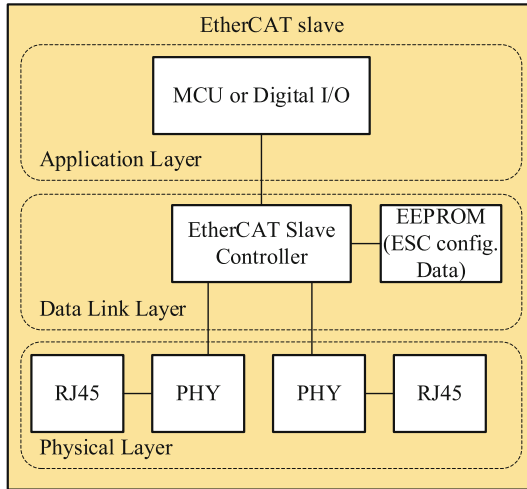


Fig. 1. Architecture of the EtherCAT slave station

Moving to the Data Link layer, both the ESC and EEPROM store the initial configuration. The ESC serves as the primary communication processor for the slave, and it stores the frame data processed in the Dual-port RAM (DPRAM). When dealing with a straightforward application, the application layer can be implemented on the digital I/O interface. However, for more complex processing requirements, the application layer is implemented on the local host MCU (Microcontroller Unit). This setup allows for more sophisticated data processing and control capabilities. By understanding the architecture and components of an EtherCAT slave system, engineers can effectively design and implement EtherCAT-based solutions to meet the specific needs of industrial automation and control applications.

EtherCAT slave controllers are typically built on either an ASIC (Application-Specific Integrated Circuit) or FPGA (Field-Programmable Gate Array) platform. These ESCs can be classified based on various factors, including type, manufacturer, package type, size, DPRAM capacity, FMMU (Fieldbus Memory Management Unit) support, distributed clock capability, and other relevant features, depending on the specific requirements of the research or application. Table 2 provides an overview of various controllers available in selected studies, offering valuable insights into the different options researchers have explored in the realm of EtherCAT slave controllers. This information is beneficial for understanding the existing ESC landscape and for making informed decisions when selecting the appropriate controller for a particular EtherCAT-based project.

Table 2. List of the state-of-the-art for related researches in EtherCAT slave

Author(s)	Publication year	EtherCAT-based slave controller	μ C Interface	Main μ C	Sync signal	Application
Nguyen, V. Q. et al. [21]	2017	ET100	SPI	TMS320F2812	N/A	Closed-Loop Stepper Motor Drive
Liu, J. et al. [22]	2020	ET100	Parallel 16-bit	Microprocessor Zynq-7020	x	I/O module
Fey, J. H. et al. [23]	2019	FPGA with EtherCAT IP Core	Parallel 16-bit	FPGA	x	Modular Multilevel Converter
Zheng, L. et al. [24]	2020	AX58100	SPI	STM32G431	N/A	Step servo
Jiao, B. et al. [25]	2014	ESC20	Parallel 16-bit	TMS320F2812 DSP	x	Steel Plate Loading and Unloading System
Jing, H. et al. [26]	2023	LAN9252	Parallel 16-bit	STM32F767	x	DC Servo driver
Herron, C. et al. [27]	2023	LAN9252	SPI	TM4C123GXL	x	DC Servo driver
Mishra, H. et al. [28]	2022	CIFX90E-RE	mPCIe	SBC MIG 5251	N/A	HMI module

4 EtherCAT-Based Synchronization Mechanism

The EtherCAT DC allows all EtherCAT devices to share a system time for managing device node synchronization. Accurate synchronization is known to be crucial to the synchronization process. For instance, in a CNC system, multiple joint points must operate concurrently and move in together. Serious repercussions will result if there is a significant difference between the nodes during the synchronization period. The accuracy of the EtherCAT clock synchronization algorithm needs to be improved, despite having reached a certain level.

On the EtherCAT bus with DC, there are three types of clocks: local clock, master clock and reference clock. Each slave has an internal clock known as a local clock, which presents the time that begins at zero when a slave is turned on. The master clock is the clock of EtherCAT master, which is usually an IPC with Windows or Linux OS. Therefore, compared to an EtherCAT slave clock, the precision of the master clock is often lower, and jitters are higher. The first DC-capable EtherCAT slave's local clock serves as the reference clock. Because the DC Master will read the reference clock and

distribute it to the other slaves in the same frame, the first slave is commonly selected to become a reference slave.

The latency or time difference in the EtherCAT bus comes from three sources: propagation delay, offset, and drift. All slaves must be reached before the EtherCAT frame returns to the master. When a data frame enters a slave, the slave also needs a small amount of time to process the data before moving on to the next slave. All of these delays are known as propagation delays. Propagation delay measurement and offset compensation can be done once at the starting up and during the Pre-OP state. The clock drift can also be dynamically adjusted by using the master’s data frame.

The clock synchronization algorithm of EtherCAT contains two main concerns, master - reference slave synchronization and reference slave–slave synchronization. Table 3 discusses the clock synchronization algorithm studies, including methods, advantages and disadvantages.

5 Application of EtherCAT

In the past decade, EtherCAT technology has gained popularity in robot applications [35, 36] and multi-axis systems [37, 38] mostly because of its adaptability, substantially decreased wiring, and low cycle time. In this article, representative examples from the various robotic application domains are provided below and in Table 4.

CW Hung et al. [39] proposed the delta robot control system based on EtherCAT for painting applications in the robot field. The experimental results show that the Cyclic Sync Velocity mode with PD controller is better accuracy when painting than the Cyclic Sync Position mode. However, it is necessary to adjust the proportional coefficient and derivative coefficient of the motor drivers to prove the accuracy.

Table 3. Summary of the state-of-the-art for related researches in synchronization mechanism

Author(s)	Publication year	Concern	OS	Methods	Advantage	Disadvantage
Chen, X. et al. [29]	2016	Synchronization between master and reference slave	Windows with real-time extension (RTX)	The algorithm changes the frequency of the master clock every cycle by taking the master clock value from the previous cycle and calculating it with the cycle of the reference slave	Offering an option of controlling and synchronizing masters with EtherCAT slaves on real-time Windows is RTX. The synchronous method requires no setup time at runtime	Experiments do not clearly show the master - reference slave synchronization of the algorithm, but only demonstrate system cycle has low jitter when using RTX

(continued)

Table 3. (continued)

Author(s)	Publication year	Concern	OS	Methods	Advantage	Disadvantage
Park, S. M. et al. [30]	2020		Linux with RT patch	Firstly, synchronize the master - reference slave in the init state to calculate and give the offset time to the reference clock. When in the operation state, the master will take the drift time of the reference slave and use the EMA filter to calculate and shift the system time for other slaves	This article presents a method of synchronizing the clocks by writing to the system time register of all the slaves in the network with the clock shift value. This doesn't require master time shifting	The system requires a long setup time to stabilize each time it works
Park, S. M. et al. [31]	2021		Linux with RT patch	Using GPIO signals to measure the GPI of the master and the time when the external processor starts sending values to shared memory to compensate for the synchronization delay on the maste	Provides a synchronization method for external processors - EtherCAT master shares memory and ensures that data will not be lost	The number of samples is quite large, so even in init or operation state, it takes a long time for the system to stabilize
Libo, C. et al. [32]	2023		Non-OS	Use the master and reference slave cycle times and then apply the PD algorithm to predict the next cycle shift of the master	Given the synchronous algorithm used in ARM microcontrollers with not too high frequency and no OS in embedded CNCs	The algorithm has time values that are not taken from the operation state but calculated by formulas, which leads to sometimes different calculated and actual values, leading to the system output may be wrong

Table 4. Summary of the state-of-the-art for related researches in EtherCAT applications

Author(s)	Publication year	Platform	Robot configuration	Number of slaves	Cycle time	HMI	Control Algorithm	Real-time OS included
Hung, C. W. et al. [39]	2022	Delta robot	A type of parallel robot that consists of three arms connected to universal joints at the base. The key design feature is the use of parallelograms in the arms, which maintains the orientation of the end effector	3	4	x	x	x
Delgado, R. et al. [40]	2016	Omnidirectional Mobile Robot	A simple omnidirectional mobile robot using 4 mecanum wheels. Common mecanum wheel mobile robots are either in the form of a parallelogram. The mecanum wheels are attached to the robot system to constitute a basic 45° driving system	4	1	N/A	x	x
Jia, H. et al. [41]	2017	CNC	A Wear-resistant coating testing system based on the CO-TRUST C37 motion controller. Connected via EtherCAT bus, the CO-TRUST C37 motion controller and the servo system are mainly responsible for the motion control of all the axes of the wear-resistant coating testing equipment, executing the motion code, determining the logic function of the system PLC and communicating with the host computer	6	N/A	x	x	N/A
Ahn, J. W. et al. [7]	2023	Humanoid Robot	The Humanoid Robot TOCABI was designed to reflect the height of an adult male, whose weight and height are 100 kg and 1.8 m, respectively. TOCABI has 33 joints, with two DOFs in the neck, eight in each arm, three in the waist, and six in each leg	33	0.2–0.25	x	N/A	x

(continued)

Table 4. (continued)

Author(s)	Publication year	Platform	Robot configuration	Number of slaves	Cycle time	HMI	Control Algorithm	Real-time OS included
Yuan, L. et al. [37]	2022	Multi-axis high-precision position control in material transportation	The system is a multi-axis position control system composed of four drive servo mechanisms of storing-storage, storing-supply, stop-material and supply	4	2.5	x	x	N/A
Zhang, G. et al. [42]	2019	7-DoF light-weight cooperative robot	A 7-DoF cooperative manipulator based on EtherCAT bus only needs 4 cables to communicate with the PC, and 2 cables to obtain power	1	1	N/A	N/A	x

N/A: Not applied

For mobile robot applications, the investigation in [40] developed an EtherCAT-based four-wheel omnidirectional mobile robot using mecanum wheels. The results of this experiment demonstrate that, with a minimum amount of jitter and an acceptable execution time, an open-source EtherCAT Master can function as the main controller of a mobile robot control system. In [41], there is an example of a CNC application. The CO-TRUST C37 motion controller was programmed by the authors using CODESYS software. The hardware platform employed an EtherCAT bus. The CNC controller supports I/O devices through the EtherCAT interface in addition to servo drivers.

The ability to control many axes is shown in the research [7] which presents a dual-channel real-time EtherCAT control system for the 33 degrees of freedom (DOF) TOCABI humanoid robot. The performance validation showed how to set up the dual-channel EtherCAT MainDevice so that it can drive robots with a lot of degrees of freedom at a faster communication cycle.

Some researchers [37] present the application of EtherCAT technology in material transportation to solve the problem of multi-axis high-precision position control PLC master controller is used as a EtherCAT master with integrated input and output module, power failure hold module. Furthermore, another development [42] use the EtherCAT bus as communication between the 7-DoF lightweight robot and ROS controller. The SOEM library with the real-time kernel Xenomai controls the robot when it receives the command from the ROS. The results show that the real-time control cycle is stable.

6 Conclusions

To the best of our knowledge, EtherCAT has the potential to be an applicable protocol for all industrial and manufacturing applications, thanks to its advanced technology and superior performance. This paper provides a review of researches in the past decade on specific aspects of EtherCAT that have garnered a lot of attentions. The analysis of EtherCAT master focuses on two key issues: hardware and software. Additionally,

different types of ESCs and their respective uses are compared. Furthermore, the advantages, disadvantages, and methods of each synchronization algorithm are explained and analyzed. Finally, several applications through illustrative examples are presented.

Acknowledgement. We acknowledge Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for supporting this study.

References

1. Shi, H., Lin, W., Liu, C., Jinyong, Yu.: A novel heterogeneous parallel system architecture based EtherCAT hard real-time master in high performance control system. *Electronics* **11**(19), 3124 (2022). <https://doi.org/10.3390/electronics11193124>
2. Romanov, A., Slepynina, E.: Real-time Ethernet POWERLINK communication for ROS. Part I. General concept. In: 2020 Ural Smart Energy Conference (USEC), pp. 159–162. IEEE (2020)
3. Turcato, A.C., Negri, L.H.B.L., Dias, A.L., Sestito, G.S., Flauzino, R.A.: A cloud-based method for detecting intrusions in profinet communication networks based on anomaly detection. *J. Control Autom. Elect. Syst.* **32**(5), 1177–1188 (2021)
4. Sestito, G.S., Turcato, A.C., Dias, A.L., Ferrari, P., da Silva, M.M.: Evaluating Real-Time Ethernet performance indicators for SERCOS III networks. In 2021 14th IEEE International Conference on Industry Applications (INDUSCON) (pp. 1191–1197). IEEE (2021).
5. Lindner, S., Häberle, M., Menth, M.: P4TG: 1 Tb/s Traffic Generation for Ethernet/IP Networks. *IEEE Access* **11**, 17525–17535 (2023)
6. Truong, Q. V., Thinh Ngo, H. Q.: Control and Implementation of Positioning System with Symmetrical Topology for Precision Manufacturing. *Mathematical Problems in Engineering*, (2022)
7. Ahn, J., Park, S., Sim, J., Park, J.: Dual Channel EtherCAT Control System for 33-DOF Humanoid Robot TOCABI. *IEEE Access* (2023)
8. Jansen, D., Buttner, H.: Real-time Ethernet: the EtherCAT solution. *Comput. Control. Eng.* **15**(1), 16–21 (2004)
9. Shen, H., Li, P., Luo, X.: Synchronous multi-axis motion control based on modified EtherCAT distributed clock. In 2020 Chinese Automation Congress (CAC), pp. 3674–3678. IEEE (2020).
10. Song, G., & Lei, J.: Control system design of heterogeneous master-slave robot for fracture reduction surgery of long bone. In 2023 IEEE International Conference on Mechatronics and Automation (ICMA) (pp. 2303–2308). IEEE (2023)
11. Paprocki, M., Erwiński, K.: Synchronization of electrical drives via EtherCAT fieldbus communication modules. *Energies* **15**(2), 604 (2022)
12. Nguyen, T.P., Nguyen, H., Ngo, H.Q.T.: Developing and Evaluating the Context-Aware Performance of Synchronization Control in the Real-Time Network Protocol for the Connected Vehicle. *Mobile Netw. Appl.* (2023). <https://doi.org/10.1007/s11036-023-02182-y>
13. Yi, H.C., Choi, J.Y.: Cycle time improvement of EtherCAT networks with embedded linux-based master. *IEICE Trans. Inf. Syst.* **102**(1), 195–197 (2019)
14. Huawei Zhang, Y., Xiang, H.S., Qin, F., Niu, Z.: Design and Implementation of Ethercat Master Based On ZYNQ. *IOP Conf. Ser: Mater. Sci. Eng.* **612**(4), 042052 (2019). <https://doi.org/10.1088/1757-899X/612/4/042052>
15. Kim, S., Shin, E.: A performance evaluation of open source-based EtherCAT master systems. In: Proceedings 4th International Conference Control, Dynamics, Systems, Robot, pp. 128–1 (2017).

16. Bu-Hai, S., Yong-Zhi, W., Chuan, D.: A design of realtime communication based on EtherCAT in industrial robot control system based on LinuxCNC. In 2017 29th Chinese Control And Decision Conference (CCDC) (pp. 5776–5780). IEEE (2017).
17. Brinkman, A., Morris, J., Chen, I., Sheikh, N., Warren, P.: Fastcat: an open-source library for composable EtherCAT control systems. In 2021 IEEE Aerospace Conference (50100), pp. 1–8. IEEE (2021).
18. Cho, S.Y., Delgado, R., Choi, B.W.: Feasibility Study for a Python-Based Embedded Real-Time Control System. *Electronics* **12**(6), 1426 (2023)
19. Park, S.M., Kim, H.W., Cho, H.M., Choi, J.Y.: Development of EtherCAT slave based on multi-core DSP. In 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), pp. 157–161. IEEE (2018)
20. Nguyen, Q.V., Kim, T.W., Moon, J.Y., Jeon, J.W.: Development of independent EtherCAT slave module and application to closed loop step motor drive with multi-axis. In 2016 International Conference on Computing, Communication and Automation (ICCCA) (pp. 912–917). IEEE (2016).
21. Nguyen, V.Q., Tran, N.V.P., Tran, H.N., Le, K.M., Jeon, J.W.: A closed-loop stepper motor drive based on EtherCAT. In IECON 2017–43rd Annual Conference of the IEEE Industrial Electronics Society (pp. 3361–3365). IEEE (2017)
22. Liu, J., Zhang, H., Guo, X., Chen, W.: Design of ethercat slave system based on zynq-7020 chip. In 2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA) (pp. 1916–1920). IEEE (2020)
23. Fey, J.H., Hinrichsen, F., Carstens, G., Mallwitz, R.: Development of a modular multi-level converter demonstrator with EtherCAT communication. In 2019 IEEE 13th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG) (pp. 1–6). IEEE (2019)
24. Zheng, L., Zhangyu, L., Liu, Z., Tan, C.: Design of Step Servo Slave System Based on EtherCAT. In: Liu, Q., Liu, X., Li, L., Zhou, H., Zhao, H.H. (eds.) Proceedings of the 9th International Conference on Computer Engineering and Networks. AISC, vol. 1143, pp. 193–205. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-3753-0_19
25. Jiao, B., He, X.: Application of the real-time EtherCAT in steel plate loading and unloading system. In: Li, K., Xue, Y., Cui, S., Niu, Q. (eds.) Intelligent Computing in Smart Grid and Electrical Vehicles, pp. 268–275. Springer Berlin Heidelberg, Berlin, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45286-8_28
26. Jing, H., Chen, W., Bai, S., Bai, Y.: EtherCAT industrial ethernet slave design and application study. In Second International Symposium on Computer Applications and Information Systems (ISCAIS 2023) (Vol. 12721, pp. 198–204). SPIE (2023)
27. Herron, C.W., Fuge, Z.J., Kogelis, M., Tremaroli, N.J., Kalita, B., Leonessa, A.: Design and validation of a low-level controller for hierarchically controlled exoskeletons. *Sensors* **23**(2), 1014 (2023)
28. Mishra, H., Saini, L. M., Bhandwale, A.: Design of EtherCAT Slave Controller using CIFX 90E-RE for HMI Display. In 2022 International Conference on Connected Systems & Intelligence (CSI) (pp. 1–6). IEEE (2022)
29. Chen, X., Li, D., Wan, J., Zhou, N.: A clock synchronization method for EtherCAT master. *Microprocess. Microsyst.* **46**, 211–218 (2016)
30. Park, S.M., Kim, H.W., Kim, H.J., Choi, J.Y.: Accuracy improvement of master–slave synchronization in EtherCAT networks. *IEEE Access* **8**, 58620–58628 (2020)
31. Park, S.M., Kwon, Y., Choi, J.Y.: Time synchronization between EtherCAT network and external processor. *IEEE Commun. Lett.* **25**(1), 103–107 (2020)
32. Libo, C., Taiyong, W., Songhui, J., Chong, T., Ying, T.: Innovation of EtherCAT adaptive synchronization control in embedded CNC. *Int. J. Commun. Syst.* **36**(8), e5462 (2023)

33. Liu, J., Yang, L., Xu, D., Wu, X.: A high precision clock synchronization algorithm for the EtherCAT. In 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA) (pp. 1369–1374). IEEE (2017).
34. Park, S.M., Kim, H., Kim, H.W., Cho, C.N., Choi, J.Y.: Synchronization improvement of distributed clocks in EtherCAT networks. *IEEE Commun. Lett.* **21**(6), 1277–1280 (2017)
35. Phan, D.Q., Ngo, H.Q.T.: Implementation of multiple controllers for context-inspired collaboration between human and robot by integrating the uni-axial motion and real-time operating system. *Internet of Things* **22**, 100788 (2023)
36. Sygulla, F., et al.: An EtherCAT-based real-time control system architecture for humanoid robots. In 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE) (pp. 483–490). IEEE (2018)
37. Yuan, L., Guan, X., Guan, S., Baoqi, W.: Design of Multi-axis Motion Control System Based on EtherCAT. In: Yang, Q., Li, J., Xie, K., Jianlin, H. (eds.) *The Proceedings of the 17th Annual Conference of China Electrotechnical Society: Volume I*, pp. 321–332. Springer Nature Singapore, Singapore (2023). https://doi.org/10.1007/978-981-99-0357-3_33
38. Nguyen, H., Nguyen, T.P., Ngo, H.Q.T.: Improving the tracking performance under nonlinear time-varying constraints in motion control applications: from theoretical servo model to experimental validation. *Math. Probl. Eng.* **2021**, 1–15 (2021)
39. Hung, C.W., Tseng, Y.H., Jiang, G.Y., Song, C.C.: An EtherCAT based delta robot synchronous control application. *J. Robot. Netw. Artif. Life* **9**(2), 183–186 (2022)
40. Delgado, R., Shin, W.C., Hong, C.H., Choi, B.W.: Development and control of an omnidirectional mobile robot on an ethercat network. *Int. J. Appl. Eng. Res.* **11**(21), 10586–10592 (2016)
41. Jia, H., Yao, P., Li, B., Tian, X.: Four axes wear-resistant coating testing system based on EtherCAT. In 2017 Chinese Automation Congress (CAC) (pp. 2842–2846). IEEE (2017)
42. Zhang, G., Li, Z., Ni, F., Liu, H.: A real-time robot control framework using ROS control for 7-DoF light-weight robot. In 2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) (pp. 1574–1579). IEEE (2019).