SEMI E54.20-0624
SPECIFICATION FOR SENSOR/ACTUATOR NETWORK COMMUNICATIONS FOR EtherCAT®

This Standard was technically approved by the Information & Control Global Technical Committee. This edition was approved for publication by the global Audits and Reviews Subcommittee on April 26, 2024. Available at www.semiviews.org and www.semi.org in June 2024; originally published November 2007; previously published March 2016.

NOTICE: This Standard is no longer published with the SEMI E54 suite of Standards. It is published and sold separately.

1 Purpose

1.1 This Specification is part of the SEMI Sensor/Actuator Network (SAN) suite of Standards and defines a specific communications protocol based on the EtherCAT®[1] standard. This Network Communication Standard (NCS) taken together with the SEMI Sensor/Actuator Network Standard suite and the EtherCAT standard completely and unambiguously defines an open standard providing an industry specific solution to off-the-shelf interoperability of networked devices in semiconductor manufacturing equipment.

1.2 EtherCAT is a vendor independent, open field bus standard for a wide range of applications in manufacturing, process and building automation, supported by the EtherCAT Technology Group (ETG)[2].

1.3 The application model of EtherCAT is compatible with the model of CANopen®[3]. The same mapping rules apply.

1.4 EtherCAT is optimized for high speed and inexpensive connectivity between automation control systems and distributed I/O at the device level.

2 Scope

2.1 This Standard specifies a SAN communications standard based on the EtherCAT specification that is in compliance with SEMI E54.1. As such, it specifies the protocol, services, and behavior that compliant intelligent devices must support in order to interchange information over this SAN in a method compatible with SEMI E39.

2.2 In conjunction with a SEMI Standard SAN Common Device Model (CDM) Specification and one or more SEMI Standard Specific Device Model (SDM) Specifications (e.g., for a mass flow controller), this Network Communication Standard (NCS) with the related EtherCAT standard describe the data structures, interactions, and behavior that are characteristic of the various devices on the network. This composite model forms a complete interoperability standard for communications among intelligent sensors, actuators, and controllers in semiconductor manufacturing equipment.

NOTICE: SEMI Standards and Safety Guidelines do not purport to address all safety issues associated with their use. It is the responsibility of the users of the Documents to establish appropriate safety and health practices, and determine the applicability of regulatory or other limitations prior to use.

3 Limitations

3.1 This Standard specifies a semiconductor equipment SAN based solely on EtherCAT and is a companion document to the EtherCAT specification, including, by reference, the EtherCAT standard; thus, a complete specification of this standard necessarily includes the EtherCAT specifications.

3.2 The specifications within are strictly enhancements that provide additional capabilities over and above those currently required by EtherCAT. Included throughout this Standard, primarily in § 6, is information paraphrased from the EtherCAT specifications such as: protocol structure, capabilities, options, and limitations. This information is provided here for reference only and is not intended to provide specification definitions. In all such areas, refer to the

[1] Trademark is owned by Beckhoff Automation GmbH.

[2] EtherCAT Technology Group, Ostendstr. 196, 90482 Nuremberg, Germany. Telephone: +49.911.540.56.20; Fax: +49.911.540.56.29; http://www.ethercat.org

[3] CANopen — CAN based communication standard defined in CENELEC Standard EN50325-4, CENELEC - European Committee for Electrotechnical Standardization.

Documents Document Sharing Hub
Full document details: SEMI E54.20-0624

EtherCAT specification documents for information, available from IEC and ETG. This Document is limited to describing enhancements or limitations to the EtherCAT specification that are imposed by this Standard.

4 Referenced Standards and Documents

4.1 *SEMI Standards and Safety Guidelines*

SEMI E39 — Object Services Standard: Concepts, Behavior, and Services

SEMI E54.1 — Specification for Sensor/Actuator Network Common Device Model

SEMI E54.3 — Specification for Sensor/Actuator Network Specific Device Model for Mass Flow Device

SEMI E54.10 — Specification for Sensor/Actuator Network Specific Device Model for an In Situ Particle Monitor Device

SEMI E54.11 — Specification for Sensor/Actuator Network Specific Device Model for Endpoint Devices

SEMI E54.18 — Specification for Sensor/Actuator Network Specific Device Model for Vacuum Pump Device

SEMI E54.22 — Specification for Sensor/Actuator Network Specific Device Model for Vacuum Pressure Gauges

SEMI E54.24 — Specification for Sensor/Actuator Network Specific Device Model of a Generic Equipment Networked Sensor (GENSen)

4.2 *IEC® Standards*[4]

IEC 61158-2 — Digital Data Communication for Measurement and Control – Fieldbus for Use in Industrial Control Systems – Part 2: Physical Layer Specification and Service Definition

IEC 61158-6-12 — Digital Data Communication for Measurement and Control – Fieldbus for Use in Industrial Control Systems – Part 6: Application Layer Protocol Specification – Type 12 Elements

4.3 *IEEE® Standards*[5]

IEEE 802 — Local and Metropolitan Area Networks: Overview and Architecture

IEEE 802.1Q — Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Common Specifications – Virtual Bridged Local Area Networks

IEEE 802.3 — Telecommunications and Information Exchange Between Systems – Local and Metropolitan Area Networks – Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications

4.4 *IETF® Documents*[6]

RFC 768 — UDP: User Datagram Protocol[7]

RFC 791 — IP: Internet Protocol[8]

NOTICE: Unless otherwise indicated, all documents cited shall be the latest published versions.

5 Terminology

5.1 *Abbreviations and Acronyms*

5.1.1 *CoE* — CANopen over EtherCAT

[4] International Electrotechnical Commission, 3 rue de Varembé, Case Postale 131, CH-1211 Geneva 20, Switzerland; Telephone: +41.22.919.02.11, http://www.iec.ch. IEC trademark is owned by International Electrotechnical Commission.

[5] Institute of Electrical and Electronics Engineers, 3 Park Avenue, 17th Floor, New York, NY 10016-5997, USA; Telephone: +1.212.419.7900, Fax: +1.212.752.4929, http://www.ieee.org. IEEE trademark is owned by The Institute of Electrical and Electronics Engineers, Inc.

[6] Internet Engineering Task Force, IETF Administration LLC, 1000 N West Street, Suite 1200, Wilmington, DE 19801, USA; Telephone: +1.510.492.4080, Fax: +1.510.492.4001, http://www.ietf.org. IETF trademark is owned by the IETF Trust.

[7] https://tools.ietf.org/html/rfc768

[8] https://tools.ietf.org/html/rfc791

Documents
Document Sharing Hub

Full document details: SEMI E54.20-0624

5.1.2 *CSMA/CD* — Carrier sense multiple access with collision detection

5.1.3 *ECAT* — Ethernet Control and Automation Technology

5.1.4 *EoE* — Ethernet over EtherCAT

5.1.5 *FoE* — File transfer over EtherCAT

5.1.6 *IO* — Input Output

5.1.7 *IP* — Internet Protocol

5.1.8 *NCS* — Network Communication Standard

5.1.9 *OSI* — Basic Reference Model for Open Systems Interconnection [ISO 7498]

5.1.10 *PDO* — Process Data Object

5.1.11 *PHY* — Physical Layer Component

5.1.12 *SAC* — Sensor, Actor, Controller (Object)

5.1.13 *SAC* — Sensor/Actor Network

5.1.14 *SDM* — Specific Device Model

5.1.15 *SDO* — Service Data Object

5.1.16 *SoE* — Servo profile over EtherCAT

5.1.17 *UDP* — User Datagram Protocol

5.2 *Terminology Defined in Sensor/Actuator Network Common Device Model, SEMI E54.1*

5.2.1 *attribute*

5.2.2 *behavior*

5.2.3 *byte*

5.2.4 *device*

5.2.5 *device model*

5.2.6 *instance*

5.2.7 *object*

5.2.8 *service*

5.2.9 *state diagram*

5.3 *Terminology Mapping*

5.3.1 As this Standard defines the mapping of CDM data structure and behavior over a network it makes use of many of the terms in SEMI E54.1. Table 1 provides a mapping of fundamental terminology of the CDM Standard into this Standard which uses the terminology of EtherCAT.

Table 1 Mapping of CDM to NCS Terminology

CDM Term	*NCS Equivalent*
Device	=
Object	=
Instance	=
Attribute	=
Behavior	=
Service	=

Documents Document Sharing Hub

Full document details: SEMI E54.20-0624

CDM Term	*NCS Equivalent*
State Diagram	State Machine
Byte	=, Unsigned8
Nibble	Specific field coding
Character String	Visible String

5.4 *EtherCAT Specific Definitions*

5.4.1 *Context Management* — network-accessible information (communication objects) that supports managing the operation of the system, including the application layer.

NOTE 1: Managing includes functions such as controlling, monitoring, and diagnosing.

5.4.2 *Device Description* — an electronic file that provides a clear and comprehensive description of the characteristics of a device type in a precisely defined format. Also called an xml File.

5.4.3 *MainDevice* — a device that manages its assigned SubDevices and handles user data exchange; usually a programmable controller.

5.4.4 *SubDevice* — a device that is configured and managed by MainDevice; a SubDevice initiates no unsolicited communications.

5.4.5 *module* — hardware or logical component of a physical device.

5.4.6 *channel* — single physical or logical link of an input or output application object of a server to the process.

5.4.7 *manufacturer ident number* — central administrative number can be assigned by ETG.

5.4.8 *ECAT datagram* — part of an Ethernet frame containing an EtherCAT service.

5.4.9 *SubDevice information interface* — object which contains device description data values in non-volatile attributes.

5.4.10 *Error counter* — containing information about invalid received Ethernet frames.

5.4.11 *CoE read service* — is described as SDO Upload service in the EtherCAT protocol specification.

5.4.12 *CoE write service* — is described as SDO Download service in the EtherCAT protocol specification.

5.4.13 *CoE Attribute* — is described as object in the EtherCAT protocol specification.

5.4.14 *CoE Attribute ID* — is described as index and subindex in the EtherCAT protocol specification.

5.4.15 *IO data exchange* — cyclic writing of output data and reading of input data.

5.4.16 *process data object* — object with attributes designated to be transferred with the IO data exchange.

5.4.17 *PDO Mapping* — defines how CoE Attributes shall be assembled to a process data object.

5.4.18 *PDO Assign* — defines how process data objects shall be assembled to the output and input data which will be transferred cyclically with IO data exchange.

6 Communication Protocol High Level Structure

6.1 The EtherCAT protocol constitutes a collapsed form of the OSI seven layer architecture. EtherCAT uses physical, data link, network, transport, and application layers of the Reference Model. For real time communication a three layer approach with physical, data link and application is specified. This section has been formatted to be aligned with the Basic Reference Model for OSI. Figure 1 gives an overview of the architecture.

Documents
Document Sharing Hub

Full document details: SEMI E54.20-0624

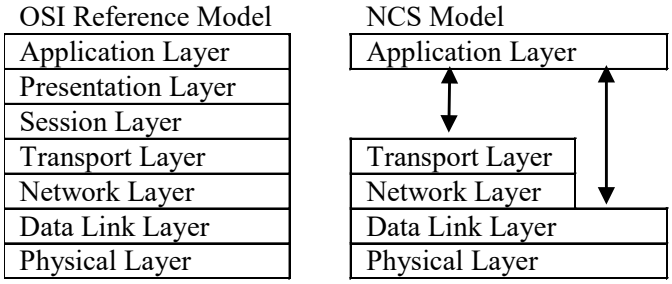

Figure 1
Layering of EtherCAT in Relation to OSI

6.2 *General Behavior*

6.2.1 In a typical remote I/O configuration, single MainDevice architectures are used to optimize response times.

6.2.2 Message transfer is organized in cycles. A message cycle mainly consists of a set of Ethernet frames sent from the MainDevice and received again when all SubDevices have processed these Ethernet frames. Every outgoing Ethernet frame is associated to an incoming Ethernet frame. The monitoring of the data transfer is done by the receiver of an Ethernet frame.

6.2.3 A brief description of the EtherCAT protocol follows in the sections below. For protocol efficiency, EtherCAT does not define layers 3 through 7 for cyclic IO data transfer and alarms. Layer 7 is the interface between the Application Process and the communication stack.

NOTE 2: The information contained in this section is for reference only. It in no way represents specifications for EtherCAT. Refer to related documentation for these specifications.

6.3 *Physical Layer — Layer 1*

6.3.1 The Physical Layer of IEEE 802.3 is adopted. There are two recommendations specified for the Physical Layer (PHY): twisted pair (100BASE-TX) and optical (100BASE-FX) in EtherCAT. Refer to IEEE 802.3 standard for more information about these options. Additionally an optional physical layer is specified in IEC 61158-2—the so called EBUS interface. It can be used for backplane applications to optimize cost and performance. If an accepted physical standard other than IEEE 802.3 is being used it shall be clearly specified in the product documentation.

6.4 *Data Link Layer — Layer 2*

6.4.1 *Data Transfer*

6.4.1.1 The Data Link Layer provides the functions for sending and receiving data over the network. Protocol Data Units (PDU) are packaged, delivered, and checked. Checks are used to guard against Line Protocol Errors (e.g., frame, overrun, and coding violations). IEEE 802.3 Data Link MAC sublayer is the preferred Data link technology of EtherCAT. If an accepted data link standard other than IEEE 802.3 is being used it shall be clearly specified in the product documentation.

6.4.1.2 A PDU is restricted to 1500 bytes. A protocol overhead of 28 Bytes is needed for addressing, error detection and protocol selection.

6.4.2 *Conveyance of Data*

6.4.2.1 In IEEE 802 networks bridges (switches) are used to transport data. These bridge functions are defined in IEEE 802.1Q. Real time extensions can be used to enhance performance and availability of bridged networks.

6.4.3 *Extensions to Ethernet Data Link*

6.4.3.1 In addition to the Definitions of IEEE 802 EtherCAT uses a specific Ethertype (0x88A4) and subdivides an Ethernet Frame in several ECAT datagrams. Each datagram can read or write memory locations within one or more EtherCAT SubDevices. This is done on the fly. With this method a MainDevice can address several SubDevices with a single Ethernet frame. This increases protocol efficiency as well as MainDevice performance.

Documents Document Sharing Hub
Full document details: SEMI E54.20-0624

6.5 *Network Layer — Layer 3*

6.5.1 All data transfer can use a protocol with no network layer, but also with IP (RFC 791) as network layer.

6.6 *Transport Layer — Layer 4*

6.6.1 All services can use UDP (RFC 768) as transport layer.

6.6.2 The UDP transport allows the same type of services as the direct access to Data Link layer using UDP destination port 34980.

6.7 *Session Layer — Layer 5*

6.7.1 There is no distinct Session Layer in this CDM.

6.8 *Presentation Layer — Layer 6*

6.8.1 There is no distinct Presentation Layer in this CDM. The Encoding is part of the application layer.

6.9 *Application Layer — Layer 7*

6.9.1 The EtherCAT application layer is structured in a so called service definition and protocol specification. The service definition uses an object orientated approach and specifies the services for remote access and local functions together with their objects. The protocol specification includes both coding and state machines.

6.9.2 *Service Definition*

6.9.2.1 Application layer services are structured to reflect the needs of flexible configurable automation devices.

6.9.2.2 The Application layer offers services for communication management, for exchange of nonvolatile information about the SubDevices and means for running synchronized applications.

6.9.2.3 Different protocol suites are specified to support user tasks:

- CANopen profile for exchange of process data and service data (CoE)
- File transfer (FoE)
- Tunnelling of Ethernet services (EoE)

6.9.2.4 The default protocol is the CoE protocol. CoE supports an object orientated access to up to 65536 CoE attributes.

6.9.2.5 Several attribute classes are specified:

- 0x0000-0x0fff for data types
- 0x1000-0x1fff for communication settings
- 0x2000-0x5fff is vendor specific
- 0x6000-0xffff is used for profiles

6.9.2.6 PDO are handled mainly by buffered services which allow decoupling between application and communication. Client/Server service structure (request/response mailboxes) is used for Service Data Transfer, Context and Diagnosis. (Service parameters associated with services are communicated as referenced in the syntax conventions sections of the EtherCAT specification. For example, refer to IEC 61158-6-12.)

6.9.2.7 Cyclic functionalities offered in EtherCAT:

- Exchange of IO data with related SubDevices

6.9.2.8 Acyclic functionalities offered in EtherCAT:

- Read diagnosis from SubDevices
- Configuration of SubDevices
- Write parameter data to SubDevices (startup or application parameter)

Documents
Document Sharing Hub

Full document details: SEMI E54.20-0624

- Acyclic access to service data of SubDevices

6.9.2.9 A modular device consists of a structured set of CoE Attributes and PDO (refer to Figure 2). There are two structuring elements:

- CoE Attributes
- Modules

6.9.2.10 CoE Attributes are defined for:

- PDO for periodic reporting
- Configuration for settings
- Information
- Diagnosis for event collection
- Service Transfer for polled access

A set of services are defined for these CoE Attributes.

6.9.2.11 The Coe Attribute IDs from 0xf000 to 0xffff are used to address the device itself. There is a set of identification and maintenance parameters defined that are mandatory for enhanced EtherCAT devices.

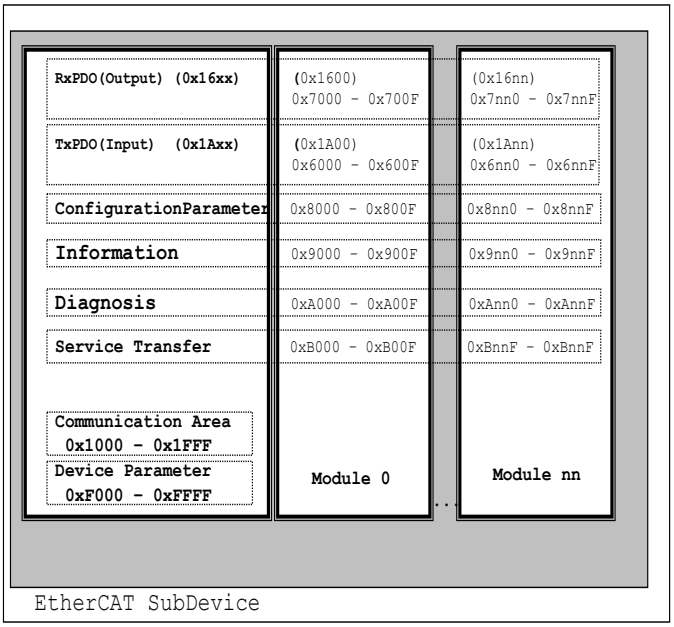

Figure 2
Modular Device Model

6.9.3 *Protocol Specification*

6.9.3.1 The Protocol specification is divided in a coding section and in a state machine description. The coding defines the frame structure of the different mailbox protocols and the exact representation of the CoE Attributes.

6.9.3.2 The most important behavioral description is the EtherCAT State Machine describing the access to a SubDevice. A MainDevice has to establish the context first by checking the SubDevice parameter. A check is done to ensure that the appropriate device type is accessed and resources can be assigned to enable application layer communication. This is done in the INIT state of an EtherCAT SubDevice.

Documents Document Sharing Hub
Full document details: SEMI E54.20-0624