

EtherNet/IP Protocol

EtherNet/IP is an industrial protocol that allows bidirectional data transfer with PLCs. It encapsulates the object-oriented Common Industrial Protocol (CIP).

This section describes the EtherNet/IP messages and data formats. EtherNet/IP communication enables the client to:

- Switch jobs.
- Align and run sensors.
- Receive sensor states, stamps, and measurement results.

To use the EtherNet/IP protocol, it must be enabled and configured in the active job.

 The Gocator 4.x firmware uses mm, mm², mm³, and degrees as standard units. In all protocols, values are scaled by 1000, as values in the protocols are represented as integers. This results in effective units of mm/1000, mm²/1000, mm³/1000, and deg/1000 in the protocols.

For information on configuring the protocol using the Web interface, see *Ethernet Output* on page 326.

Concepts

To EtherNet/IP-enabled devices on the network, the sensor information is seen as a collection of objects, which have attributes that can be queried.

Gocator supports all required objects, such as the [Identity](#) object, [TCP/IP](#) object, and [Ethernet Link](#) object. In addition, [assembly objects](#) are used for sending sensor and sample data and receiving commands. There are four assembly objects: the command assembly (32 bytes), the runtime variable configuration assembly (64 bytes), the sensor state assembly (100 bytes), and the sample state assembly object (380 bytes). The data attribute (0x03) of the assembly objects is a byte array containing information about the sensor. The data attribute can be accessed with the GetAttribute and SetAttribute commands.

The PLC sends a command to start a Gocator. The PLC then periodically queries the attributes of the assembly objects for its latest measurement results. In EtherNet/IP terminology, the PLC is a scanner and the Gocator is an adapter.

The Gocator supports unconnected or connected explicit messaging (with TCP). Implicit I/O messaging is also supported. For more information, see http://lmi3d.com/sites/default/files/APPNOTE_Implicit_Messaging_with_Allen-Bradley_PLCs.pdf.

The default EtherNet/IP ports are used. Port 44818 is used for TCP connections and UDP queries (e.g., list Identity requests). Port 2222 for UDP I/O Messaging is not supported.

Basic Object

Identity Object (Class 0x01)

Attribute	Name	Type	Value	Description	Access
1	Vendor ID	UINT	1256	ODVA-provided vendor ID	Get
2	Device Type	UINT	43	Device type	Get
3	Product Code	UINT	2000	Product code	Get
4	Revision	USINT	x.x	Byte 0 - Major revision	Get
		USINT		Byte 1 - Minor revision	
6	Serial number	UDINT	32-bit value	Sensor serial number	Get
7	Product Name	SHORT STRING 32	"Gocator"	Gocator product name	Get

TCP/IP Object (Class 0xF5)

The TCP/IP Object contains read-only network configuration attributes such as IP Address. TCP/IP configuration via Ethernet/IP is not supported. See Volume 2, Chapter 5-3 of the CIP Specification for a complete listing of TCP/IP object attributes.

Attribute	Name	Type	Value	Description	Access
1	Status	UDINT	0	TCP interface status	Get
2	Configuration Capability	UINT	0		Get
3	Configuration Control	UINT	0	Product code	Get
4	Physical Link Object	Structure (See description)		See 5.3.3.2.4 of CIP Specification Volume 2: Path size (UINT) Path (Padded EPATH)	Get
5	Interface Configuration	Structure (See description)		See 5.3.3.2.5 of CIP Specification Volume 2: IP address (UDINT) Network mask (UDINT) Gateway address (UDINT) Name server (UDINT) Secondary name (UDINT) Domain name (UDINT)	Get

Ethernet Link Object (Class 0xF6)

The Ethernet Link Object contains read-only attributes such as MAC Address (Attribute 3). See Volume 2, Chapter 5-4 of the CIP Specification for a complete listing of Ethernet Link object attributes.

Attribute	Name	Type	Value	Description	Access
1	Interface Speed	UDINT	1000	Ethernet interface data rate (mbps)	Get
2	Interface Flags	UDINT		See 5.4.3.2.1 of CIP Specification Volume 2: Bit 0: Link Status 0 – Inactive 1 – Active Bit 1: Duplex 0 – Half Duplex 1 – Full Duplex	Get
3	Physical Address	Array of 6 USINTs		MAC address (for example: 00 16 20 00 2E 42)	Get

Assembly Object (Class 0x04)

The Gocator Ethernet/IP object model includes the following assembly objects: command, runtime variable configuration, sensor state, and sample state, implicit messaging command, and implicit messaging output.

All assembly object instances are static. Data in a data byte array in an assembly object are stored in the big endian format.

Command Assembly

The command assembly object is used to start, stop, and align the sensor, and also to switch jobs on the sensor.

Command Assembly

Information	Value
Class	0x4
Instance	0x310
Attribute Number	3
Length	32 bytes
Supported Service	0x10 (SetAttributeSingle)

Attributes 1 and 2 are not implemented, as they are not required for the static assembly object.

Attribute 3

Attribute	Name	Type	Value	Description	Access
3	Command	Byte Array	See Below	Command parameters Byte 0 - Command. See table below for specification of the values.	Get, Set

Command Definitions

Value	Name	Description
0	Stop running	Stop the sensor. No action if the sensor is already stopped
1	Start Running	Start the sensor. No action if the sensor is already started.
2	Stationary Alignment	Start the stationary alignment process. Byte 1 of the sensor state assembly will be set to 1 (busy) until the alignment process is complete, then back to zero.
3	Moving Alignment	Start the moving alignment process. Byte 1 of the sensor state assembly will be set to 1 (busy) until the alignment process is complete, then back to zero.
4	Clear Alignment	Clear the alignment.
5	Load Job	Load the job. Set bytes 1-31 to the file name (one character per byte, including the extension). File name must be null-terminated and end with ".job".

Runtime Variable Configuration Assembly

The runtime variable configuration assembly object contains the sensor's intended runtime variables.

Runtime Variable Configuration Assembly

Information	Value
Class	0x04
Instance	0x311
Attribute Number	3
Length	64 bytes
Supported Service	0x10 (SetAttributeSingle)

Attribute 3

Attribute	Name	Type	Value	Description	Access
3	Command	Byte Array	See below	Runtime variable configuration information. See below for more details.	Get

Sensor State Information

Byte	Name	Type	Description
0-3	Runtime Variable 0	32s	Stores the intended value of the Runtime Variable at index 0.
4-7	Runtime Variable 1	32s	Stores the intended value of the Runtime Variable at index 1.
8-11	Runtime Variable 2	32s	Stores the intended value of the Runtime Variable at index 2.
12-15	Runtime Variable 3	32s	Stores the intended value of the Runtime Variable at index 3.
16-63	Reserved		

Sensor State Assembly

The sensor state assembly object contains the sensor's states, such as the current sensor temperature, frame count, and encoder values.

Sensor State Assembly

Information	Value
Class	0x04
Instance	0x320
Attribute Number	3
Length	100 bytes
Supported Service	0x0E (GetAttributeSingle)

Attributes 1 and 2 are not implemented, as they are not required for the static assembly object.

Attribute 3

Attribute	Name	Type	Value	Description	Access
3	Command	Byte Array	See below	Sensor state information. See below for more details.	Get

Sensor State Information

Byte	Name	Type	Description
0	Sensor State		Sensor state: 0 - Stopped 1 - Running
1	EtherNet/IP Command in Progress		Command busy status: 0 - Not busy 1 - Busy performing the last command Bytes 2 and 19-83 below are only valid when there is no command in progress.
2	Alignment State		Alignment status: 0 - Not aligned 1 - Aligned The value is only valid when byte1 is set to 0.
3-10	Encoder	64s	Current encoder position
11-18	Time	64s	Current timestamp
19	Current Job Filename Length	8u	Number of characters in the current job filename. (e.g., 11 for "current.job"). The length includes the .job extension. Valid when byte 1 = 0.
20-83	Current Job Filename		Name of currently loaded job, including the ".job" extension. Each byte contains a single character. Valid when byte 1 = 0.
84-87	Runtime Variable 0	32s	Runtime variable value at index 0