



Operating instructions
Diagnostic electronics
with EtherNet/IP interface for vibration sensors

GB

VSE951

Contents

1	Safety instructions	3
2	Preliminary note	4
2.1	Symbols used	4
2.2	Warnings used	4
3	Intended use	5
4	Device functions	6
4.1	Function description	7
4.2	Firmware	7
5	Mounting	8
5.1	Emission	8
5.2	Sources of interference	8
5.3	Cable routing	8
5.4	Installation instructions	8
6	Electrical connection	9
6.1	Wiring	9
6.2	Connection of the sensors	10
7	EtherNet/IP	11
7.1	EtherNet/IP object classes, messages, services	11
7.2	Fieldbus parameter data model	11
7.3	Supported communication types	15
7.4	CIP object classes	15
7.4.1	Identity object (class code 0x01)	15
7.4.2	Message router object (class code 0x02)	16
7.4.3	Assembly object (class code 0x04)	16
7.4.4	Connection manager object (class code 0x06)	17
7.4.5	Device level ring object (class code 0x47)	17
7.4.6	Quality of service object (class code 0x48)	18
7.4.7	TCP/IP object (class code 0xF5)	18
7.4.8	Ethernet link object (class code 0xF6)	20
7.5	EtherNet/IP assembly instances	21
7.6	EtherNet/IP connection types	22
7.6.1	EtherNet/IP - defined connections in the standard EDS file	22
7.7	EtherNet/IP functions	23
7.7.1	Quality of service (QoS)	23
7.7.2	Device level ring (DLR)	23
7.7.3	Address conflict detection (ACD)	23
7.8	EtherNet/IP properties	23
7.9	Behaviour if parameter set is changed	24
8	Factory setting	25
8.1	General factory setting	25
8.2	Factory setting VSE951 - EtherNet/IP	25
9	Parameter setting	26
10	Operating and display elements	27
10.1	Operating states of the network (NET) and mode (MOD) status LED	27
11	Maintenance, repair and disposal	28



1 Safety instructions

- The unit described is a subcomponent for integration into a system.
 - The system architect is responsible for the safety of the system.
 - The system creator undertakes to perform a risk assessment and to create documentation in accordance with legal and normative requirements to be provided to the operator and user of the system. This documentation must contain all necessary information and safety instructions for the operator, the user and, if applicable, for any service personnel authorised by the architect of the system.
- Read this document before setting up the product and keep it during the entire service life.
- The product must be suitable for the corresponding applications and environmental conditions without any restrictions.
- Only use the product for its intended purpose (→ Intended use).
- If the operating instructions or the technical data are not adhered to, personal injury and/or damage to property may occur.
- The manufacturer assumes no liability or warranty for any consequences caused by tampering with the product or incorrect use by the operator.
- Installation, electrical connection, set-up, operation and maintenance of the product must be carried out by qualified personnel authorised by the machine operator.
- Protect units and cables against damage.




2 Preliminary note

You will find instructions, technical data, approvals, accessories and further information using the QR code on the unit / packaging or at www.ifm.com.

2.1 Symbols used

- ✓ Requirement
- ▶ Instructions
- ▷ Reaction, result
- [...] Designation of keys, buttons or indications
- Cross-reference
-  Important note
Non-compliance may result in malfunction or interference.
-  Information
Supplementary note

2.2 Warnings used

	ATTENTION Warning of damage to property
	CAUTION Warning of personal injury ▷ Slight reversible injuries may result.
	WARNING Warning of serious personal injury ▷ Death or serious irreversible injuries may result.

3 Intended use

The device has been designed for process value monitoring, vibration monitoring and analysis of dynamic signals.

4 Device functions

The diagnostic electronics has

- 2 analogue inputs
- 4 dynamic inputs
- 1 analogue or digital output
- 1 digital output
- 1 TCP/IP parameter setting interface
- 2 EtherNet/IP ports

Input IN1: connection for a pulse signal (HTL).

Input IN2: connection for an analogue current signal (4...20 mA).

The analogue inputs can be used

- as trigger for measurements (e.g. rotational speed for vibration diagnostics)
- as trigger of a counter
- for process monitoring

VSA, VSP or standard IEPE acceleration sensors can be connected to the dynamic inputs.

The dynamic inputs can be used for

- vibration monitoring
- vibration diagnostics
- analysis of other dynamic signals

The hardware outputs can be configured as 2 x binary (NO/NC) or as 1 x analogue (0/4...20 mA) and 1 x binary (NO/NC).

The outputs can be used for

- time-critical alarms (e.g. machine protection, response time up to 1 ms)
- alarm output
- analogue value output of values measured by the diagnostic electronics

The parameter setting interface (TCP/IP) is used for communication between the diagnostic electronics and a PC (e.g. VES004 parameter setting software).

The parameter setting interface can be used for

- parameter setting of the device
- online data monitoring
- reading the history memory
- firmware update

The EtherNet/IP ports are used for communication between the diagnostic electronics and an EtherNet/IP controller (e.g. PLC).

Functions of the EtherNet/IP interface

- transferring the current measured values, limits and alarm states of the diagnostic electronics to the PLC
- reading the counter readings of the diagnostic electronics
- writing rotational speeds and other values from the PLC to the diagnostic electronics
- writing teach values from the PLC to the diagnostic electronics

ATTENTION

The unit is not approved for safety-related tasks in the field of operator protection.

4.1 Function description

With the device

- vibration monitoring (total vibration to ISO)
- condition monitoring (condition-based monitoring on the basis of vibration characteristics)
- machine protection/process monitoring (monitoring vibration characteristics in real time with a very fast response time up to 1 ms)

can be implemented.

Monitoring of

- up to 24 objects (indicators for different machine parts, vibration characteristics or process values)
- dynamic values within the time range (e.g. v-RMS to ISO)
- dynamic values within the frequency range FFT or HFFT (e.g. unbalance or rolling element bearing)
- process values (analogue signals) for current value above or below the limit

The device has an internal history memory (> 850,000 values) with real-time clock and flexible memory interval per object. The memory is a ring memory (FIFO).

Up to 32 counters can be configured to measure the duration of operating times and/or how long a limit value is exceeded.

The signals at the inputs are permanently picked up and continuously monitored according to the set parameters.

With objects in the frequency range (unbalance, rolling element bearing,...), monitoring is done in multiplex mode.

With objects in the time range (v-RMS, a-RMS and a-Peak), all 4 dynamic inputs are monitored simultaneously and without interruption.

The two outputs OU1/2 can be used for alarms.

Parameter setting of monitoring tasks and alarming is done via the VES004 software. The software allows to display and record the current measured values, spectra and time signals (online data).

Via the Ethernet interface of the device, networking is possible to visualise data (measured values, alarm states,...) in other systems (e.g. SCADA, MES,...).

Data (e.g. measured values, alarm states, limits, rotational speeds, timer readings,...) is exchanged between the diagnostic electronics and the EtherNet/IP controller (e.g. PLC) via the EtherNet/IP ports.

4.2 Firmware

- ▶ Recommendation: Install/Use the latest firmware to use all device functions.

The firmware can only be updated via the VES004 PC software. Only the firmware of the entire device can be updated.



- ▷ Firmware and operating software → download area www.ifm.com
- ▷ A description of all firmware parameters and their meaning → VES004 PC software manual.



- During the firmware update, the parameter set and the history on the device are deleted. All counters are reset.
The IP settings of the configuration and fieldbus interface are retained.

- ▶ Recommendation: Run a parameter backup before the firmware update.

5 Mounting



- ▶ Disconnect the power to the machine during installation.
- ▶ Use a flat mounting surface for installation.
- ▶ Fasten the module onto the mounting surface using M6 screws and washers.
- ▶ Ground the device with the earthing screw provided.
- ▶ To guarantee the protection rating IP 67, M12 sockets from ifm must be used.
- ▶ Cover the unused sockets with protective caps (to be ordered separately).

When preparing for cable installation, the local conditions and the corresponding mounting regulations are very important. Cables can be installed, for example, in cable ducts or on cable bridges.

A minimum distance between the cabling and possible sources of interference (e.g. machines, welding equipment, power lines) is defined in the applicable regulations and standards. During system planning and installation, these regulations and standards must be taken into account and observed.

Protect the bus cables from sources of electric/magnetic interference and mechanical strain.

Observe the guidelines regarding "electromagnetic compatibility" (EMC) to keep mechanical risks and interference to a minimum.

5.1 Emission

This is a class A product. The device may cause radio interference in domestic areas.

- ▶ If necessary, take measures to prevent radio interference.

5.2 Sources of interference

- ▶ Signal cables and power supply lines should not be installed in parallel.
- ▶ If necessary, metal isolating segments should be placed between the power supply lines and signal cables.

5.3 Cable routing

Network/bus cables: Install network/bus cables in separate cable ducts or separate cable bundles.



- ▶ Where possible, do not install network/bus cables parallel to power supply lines.
- ▶ Install network/bus cables at least 10 cm away from power lines.

5.4 Installation instructions

Electrostatic discharge

The device contains components that can be damaged or destroyed by electrostatic discharge.

- ▶ When handling the device, observe the necessary safety precautions against electrostatic discharge (ESD) according to EN 61340-5-1 and IEC 61340-5-1.

6 Electrical connection

The national and international regulations for the installation of electrical equipment must be adhered to. Avoid contact with dangerous contact voltages.

- ▶ Disconnect power.
- ▶ Connect the unit.
- ▶ To prevent negative effects on the functions caused by noise voltages, lay sensor cables and load cables separately.

Maximum cable length for the sensors 1...4: 250 m.

Maximum cable length for the inputs IN1 and IN2: 30 m.

The outputs are short-circuit proof and can be configured as either normally closed or normally open.

In addition, an analogue signal can be provided on output [OU 1] (0/4...20 mA) (e.g. acceleration values).


6.1 Wiring


Sensor 1...4	
M12 socket, A-coded	
1: L+ (bn)	
2: signal (wh)	
3: GND (bu)	
4: test (bk)	
5: not used	

Config / IE1 / IE2	
M12 socket, D-coded	
1: TxD+	
2: RxD+	
3: TxD-	
4: RxD-	
5: not used	

IN 1	
M12 socket, A-coded	
1: 24 V DC (bn)	
2: not used	
3: GND (bu)	
4: IN 1 pulse (bk)	
5: not used	

IN 2	
M12 socket, A-coded	

IN 2	
1: 24 V DC (bn)	
2: IN 2 4...20 mA (wh)	
3: GND (bu)	
4: not used	
5: not used	

OU / Supply	
M12 connector, A-coded	
1: 24 V DC (bn)	
2: analogue or digital (wh)	
3: GND (gn)	
4: OU2: switch	

6.2 Connection of the sensors

Adhere to the SELV criteria (safety extra-low voltage, circuit electrically isolated from other circuits, ungrounded) when the sensors are connected so that no dangerous contact voltages are applied to the sensor or transferred to the device.

If the DC circuit is to be grounded (e.g. due to national regulations), the PELV criteria must be adhered to (protective extra-low voltage, circuit electrically isolated from other circuits).

Sensor and diagnostic electronics supply are not electrically isolated.

7 EtherNet/IP

7.1 EtherNet/IP object classes, messages, services

The device supports the Common Industrial Protocol (CIP) according to the ODVA specification 3.20. EtherNet/IP™ uses the Common Industrial Protocol as the application layer. IP and TCP or UDP are used for the network and transport layers. CIP and EtherNet/IP™ are standardised by the ODVA on a manufacturer-neutral basis. The Common Industrial Protocol is an object-oriented protocol with two different types of communication between a controller and termination devices.

7.2 Fieldbus parameter data model

Input (PLC)				
Source		Data type / Byte order / Unit	Data size	Use
Dynamic inputs				
	<input name>	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Value of the signal connected to dynamic input (sensor 1...4) if it has been configured as a "DC input".
Analogue inputs				
	<input name>	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Value of the signal connected to the analogue input (IN1, IN2)
External inputs				
	<input name>	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Value of the external input (External_xx)
Objects (time domain, frequency domain, upper/lower limit monitor)				
	<object name>			
	Value	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian Has a unit (default SI unit) 	4 bytes	Object value has a unit
	Status	Byte	1 byte	Current status/state of the object 0: OK 1: warning alarm 2: damage alarm 3: inactive 4: error

Input (PLC)				
	Error	Word	2 bytes	Error codes for description of error in object state 0x0000: no error 0x0001: internal error 0x0002: calculation error 0x0004: speed out of range 0x0008: speed not stable 0x0010: invalid baseline 0x0020: invalid reference value (1) 0x0040: invalid reference value (2) 0x0100: deactivated by signal weighting 0x0200: reference value out of range 0x1000: warning alarm 0x3000: warning and damage alarm 0x8000: object inactive (by variant)
	Rotational speed	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian 	4 bytes	Trigger - rotational speed
	Reference value	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian 	4 bytes	Trigger - reference value
	Warning alarm	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian 	4 bytes	Limit - warning alarm (relative)
	Damage alarm	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian 	4 bytes	Limit - damage alarm (relative)
	Baseline	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian • Has a unit (default SI unit) 	4 bytes	Limit - unit-based baseline for time and frequency objects Note: This parameter is not applicable for the upper/lower limit monitor objects
Counter				
	<counter name>	<ul style="list-style-type: none"> • DINT • Big or Little Endian 	4 bytes	Counter value (in seconds)
History				
	<object name>			
	History value	<ul style="list-style-type: none"> • Real or DINT with factor • Big or Little Endian • Has a unit (default SI unit) 	4 bytes	Unit-based current history entry of the object

Input (PLC)				
	Average value of the history	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian Has a unit (default SI unit) 	4 bytes	Unit-based average value of the current history entry of the object
	Speed of the history	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Trigger - speed of the current history entry of the object
	Reference value of the history	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Trigger - reference value of the current history entry of the object
	Input counter of the history values	<ul style="list-style-type: none"> DINT Big or Little Endian 	4 bytes	Counter of the received history entries of the object
Alarms (OUT1 / OUT2)				
	<alarm name (OUT1)>	<ul style="list-style-type: none"> Float or DINT with factor Big or Little Endian 	4 bytes	Configuration as analogue alarm output: value of alarm output
		Byte	1 byte	Configuration as digital alarm output: alarm state Configuration as NC No alarm: 1 Alarm: 0 Configuration as NO No alarm: 0 Alarm: 1
	<alarm name (OUT2)>	Byte	1 byte	Alarm state Configuration as NC No alarm: 1 Alarm: 0 Configuration as NO No alarm: 0 Alarm: 1
Alarms (IO1 - IO8)				
	<alarm name>	Byte	1 byte	Alarm state Configuration as NC No alarm: 1 Alarm: 0 Configuration as NO No alarm: 0 Alarm: 1
General				
	Variant	Byte	1 byte	Value of the active variant (0...31)
	System mode	Byte	1 byte	Current system mode of the device 0x00: reserved 0x01: supervise (normal monitoring) 0x02: set-up (parameter setting) 0x03: measure (spectrum, raw data) 0x04: start-up (system booting) 0x05: self-test (self-test active)

Input (PLC)				
	Self-test result	Byte	1 byte	Bit pattern Bit1 - sensor 1 Bit2 - sensor 2 Bit3 - sensor 3 Bit4 - sensor 4 Note on evaluation 0x00: Sensors OK 0x01: sensor 1 self-test failed 0x02: sensor 2 self-test failed 0x04: sensor 3 self-test failed 0x08: sensor 4 self-test failed 0x0F: sensor1...4 self-test failed
	Current queue level	Byte	1 byte	Current queue level of the internal device communication
	Queue overflow counter	<ul style="list-style-type: none"> DINT Big or Little Endian 	4 bytes	Overflow counter of the internal device communication
	Checksum error counter	<ul style="list-style-type: none"> DINT Big or Little Endian 	4 bytes	Checksum error counter of the internal device communication
	Read time	DINT	4 bytes	Read device time (UTC) PROFINET devices: U32:0x00ssmmhh EtherNet/IP devices: U32:0x00hhmmss EtherCAT devices: U32:0x00hhmmss MODBUS devices: U32:0x00hhmmss
Placeholder				
	<placeholder>	Byte	xx byte	Placeholder for fieldbus transmission

Output (PLC)				
Source		Data type / Byte order / Unit	Data size	Fieldbus representation
External inputs				
	<input name>	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian 	4 bytes	Set value of the external input (External_xx)
Objects (time domain, frequency domain)				
	<object name>			
	Baseline	<ul style="list-style-type: none"> Real or DINT with factor Big or Little Endian Has a unit (default SI unit) 	4 bytes	Limits - set unit-based baseline for time and frequency objects to adjust damage limits
General				
	Variant	Byte	1 byte	Set current variant (0...31)

Output (PLC)				
	Do self-test	Byte	1 byte	Execute self-test Note A value change from 0 to ≠ 0 starts the self-test After completion of the self-test, the unit automatically switches to the "Monitoring" system mode
	Set time	DINT	4 bytes	Set time (always UTC format) PROFINET devices: U32:0x00ssmmhh EtherNet/IP devices: U32:0x00hhmmss EtherCAT devices: U32:0x00hhmmss MODBUS devices: U32:0x00hhmmss
	Set counter ID	Byte	1 byte	Set the ID of the counter (1...32)
	Set counter value	<ul style="list-style-type: none"> • DINT • Big or Little Endian 	4 bytes	Set value of the counter selected with the ID (in seconds)
Placeholder				
	<placeholder>	Byte	xx byte	Placeholder for fieldbus transmission

7.3 Supported communication types

Request	Parameter
Explicit messaging	Based on the request/response principle (e.g. use for device configuration). Message protocol: TCP
Implicit messaging	Based on the producer/consumer model (e.g. cyclic transmission of I/O data). Message protocol: UDP

7.4 CIP object classes

The device supports the following CIP object classes:

Class ID	Object
0x01	Identity
0x02	Message router
0x04	Assembly
0x06	Connection manager
0x47	Device level ring (DLR)
0x48	Quality of service (QoS)
0xF5	TCP/IP interface
0xF6	Ethernet link

7.4.1 Identity object (class code 0x01)

The identity object is required by all devices and provides the device ID and general information about the device.

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	1
2	Max instance	Get	UINT	1

Instance attributes

Attribute	Name	Access	Data type	Value / description
1	Vendor ID	Get	UINT	322 (dec)
2	Product type	Get	UINT	43 (generic device, keyable)
3	Product code	Get	UINT	151
4	Revision Major revision Minor revision	Get	STRUCT of: USINT USINT	E.g. 1 1
5	Status	Get	WORD	Bit 0 - n.a. (default value = 0) Bit 1 - n.a. (default value = 0) Bit 2 - n.a. (default value = 0) Bit 3 - n.a. (default value = 0) Bit 4...7- n.a. (default value = 0) Bit 8 - n.a. (default value = 0) Bit 9 - n.a. (default value = 0) Bit 10 - major recoverable fault (address conflict detection) Bit 11 - n.a. (default value = 0) Bit 12...15 n.a. (default value = 0)
6	Serial number	Get	UDINT	Is defined in the production process
7	Product name	Get	STRING	VSE951

Common services

Service code	Class	Instance	Service name
0x01	Yes	Yes	Get_Attribute_All
0x05	no	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single

7.4.2 Message router object (class code 0x02)

The message router object provides a messaging connection point through which an EtherNet/IP client may address a service to any object class or instance. The device does not support any access to object attributes.

7.4.3 Assembly object (class code 0x04)

The assembly object combines attributes of several objects to allow data to be sent to or received from each object via a one connection.

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	2

Attribute	Name	Access	Data type	Value
2	Max instance	Get	UINT	255

Instance attributes

Attribute	Name	Access	Data type	Value
3	Data	Get	Array of byte	Current process data of the correspondent assembly instance

Common services

Service code	Class	Instance	Service name
0x0E	Yes	Yes	Get_Attribute_Single

7.4.4 Connection manager object (class code 0x06)

The connection manager object allocates and manages the internal resources that are used for I/Os and explicit messaging connections. Forward Open/Close is supported. The device does not support any access to object attributes.

7.4.5 Device level ring object (class code 0x47)

The device level ring object (DLR) is the interface for configuration and status information for the DLR protocol.

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	3

Instance attributes

Attribute	Name	Access	Data type	Value / description
1	Network topology	Get	USINT	0 = linear 1 = ring
2	Network status	Get	USINT	0 = normal 1 = ring fault 2 = unexpected loop detected 3 = partial network fault 4 = rapid fault/restore cycle
10	Active supervisor address	Get	STRUCT of: • UDINT • ARRAY of 6 USINTs	IP and/or MAC address of the active ring supervisor: supervisor IP address supervisor MAC address
12	Capability flags	Get		Bit 0 announced-based ring node (value = 1) Bit 1 beacon-based ring node (value = 0) Bit 2...4 reserved (value = 0) Bit 5 supervisor capable (value = 0) Bit 6 redundant gateway capable (value = 0) Bit 7 flush table frame capable (value = 1) Bit 8...31 reserved (value = 0)

Common services

Service code	Class	Instance	Service name
0x01	Yes	Yes	Get_Attribute_All
0x0E	Yes	Yes	Set_Attribute_Single

7.4.6 Quality of service object (class code 0x48)

Quality of service (QoS) affects the forwarding and handling of data streams and results in individual data streams being given differential treatment (usually preferential). QoS can be used to ensure a transmission bandwidth for separate data flows. The device uses QoS in connection with prioritisation.

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	1

Instance attributes

Attribute	Name	Access	Data type	Value / description
4	DSCP urgent	Get, Set	USINT	DSCP value for CIP transport class 0/1 urgent priority message (default 55)
5	DSCP scheduled	Get, Set	USINT	DSCP value for CIP transport class 0/1 scheduled priority message (default 47)
6	DSCP high	Get, Set	USINT	DSCP value for CIP transport class 0/1 high priority message (default 43)
7	DSCP low	Get, Set	USINT	DSCP value for CIP transport class 0/1 low priority message (default 31)
8	DSCP explicit	Get, Set	USINT	DSCP value for CIP transport class 0/1 low priority message (default 31)

Note: DSCP - Differentiate Service Code Points

Common services

Service code	Class	Instance	Service name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

7.4.7 TCP/IP object (class code 0xF5)

The TCP/IP interface object makes it possible to configure the physical network interface. This includes, for example, the IP address, subnet mask and gateway address.

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	4

Instance attributes

Attribute	Name	Access	Data type	Value / description
1	Status	Get	DWORD	Bit 0...3 - interface configuration status Bit 4 - Mcast pending (always 0) Bit 5 - interface configuration pending Bit 6 - ACD status Bit 7 - ACD fault Bit 8...31 - reserved
2	Configuration capability	Get	DWORD	Bit 0 - BOOTP client Bit 1 - reserved Bit 2 - DHCP client Bit 3 - reserved Bit 4 - TCP/IP config settable via ETH/IP Bit 5 - not supported Bit 6 - not supported Bit 7 - ACD capable Bit 8...31 reserved
3	Configuration control	Get, Set	DWORD	Bit 0...bit 3 - (0 = the device uses static IP configuration; 1 = the device uses BOOTP; 2 = the device uses DHCP) Bit 4 - reserved Bit 5...31 - reserved Note: BOOTP is not supported
4	Physical link object - path size - path	Get	STRUCT of - UINT - padded EPATH	Path to physical link object Size of path Logical segments identifying the physical link object
5	Interface configuration - IP address - network mask - gateway address - name server - name server 2 - domain name	Get, Set	STRUCT of: - UDINT - UDINT - UDINT - UDINT - UDINT - STRINT	TCP/IP network interface configuration The device's IP address The device's network mask Default gateway address Primary name server (always 0.0.0.0) Secondary name server (always 0.0.0.0) Default domain name (always empty)
6	- Host name	Get, Set	STRING	Default domain name (always empty)
10	- Select ACD	Get, Set	Bool	0 = disable; 1 =enable (default)
11	Last conflict detected - ACD activity - remote MAC - ArpPd	Set	STRUCT of: USINT ARRAY of 6 USINTs ARRAY of 28 USINTs	Structure containing information related to the last conflict detected State of ACD activity when last conflict detected (0 = no conflict detected (default) 1 = probe Ipv4 address 2 = ongoing detection 3 = semi active probe) MAC address of remote node from the ARP PDU in which a conflict was detected Copy of the raw ARP PDU in which a conflict was detected

Attribute	Name	Access	Data type	Value / description
12	Ethernet/IP quick connect	Get, Set	BOOL	0 = disable (default); 1 = enable
13	Encapsulation inactivity timeout	Get, Set	UINT	Number of seconds of inactivity before TCP connection or DTLS session is closed.

Common services

Service code	Class	Instance	Service name
0x01	Yes	Yes	Get_Attribute_All
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

7.4.8 Ethernet link object (class code 0xF6)

The Ethernet link object contains specific status information of the Ethernet interface (IEEE 802.3).

Class attributes

Attribute	Name	Access	Data type	Value
1	Revision	Get	UINT	4
2	Max instance	Get	UINT	2
3	Number of instances	Get	UINT	2

Instance attributes

Attribute	Name	Access	Data type	Value / description
1	Interface speed	Get	UDINT	Interface speed currently in use. Speed in bps (e.g. 10,100)
2	Interface flags	Get	DWORD	Bit 0 - link status Bit 1 - half/full duplex status (0= half duplex, 1= full duplex) Bit 2...4 - auto negotiation status (0=auto negotiation in progress; 1 = auto negotiation and speed detection failed; 2 = auto negotiation failed but detected speed; 3 = successfully negotiated; 4 = auto negotiation not attempted) Bit 5 - not supported Bit 6 - not supported Bit 7...31 - reserved
3	Physical address	Get	ARRAY of 6 USINTs	Interface specific counters Details are defined in "THE CIP NETWORKS LIBRARY Volume 2 EtherNet/IP Adaptation of CIP"
5	Media counters	Get	Structure of 12 UINTs	Media specific counters "THE CIP NETWORKS LIBRARY Volume 2 EtherNet/IP Adaptation of CIP"

Attribute	Name	Access	Data type	Value / description
6	Interface control	Get, Set	Structure of WORD	Interface control bits Bit 0 - auto negotiation (0 = active; 1 = inactive) Bit 1 - half/full duplex (0 = half duplex; 1 = full duplex) Bit 2...15 - reserved Data rate (10 = 10 Mbps; 100 = 100 Mps)
7	Interface type	Get	USINT	Type of interface: 0 = unknown interface type 1 = the interface is internal to the device 2 = twisted pair 3 = optical fibre 4..255 = reserved
8	Interface state	Get	USINT	Current operational state of the interface: 0 = unknown interface state 1 = the interface is enabled and is ready to send and receive data 2 = the interface is disabled 3 = the interface is testing 4...255 = reserved
9	Admin state	Get	USINT	Settings of the interface state: 0 = reserved 1 = enable the interface 2 = disable the interface 3...255 = reserved
10	Interface label	Get	SHORT STRING	Interface name
11	Interface capability	Get	Structure of DWORD Structure of USINT ARRAY of structure of UINT USINT	Interface capabilities, other than speed/duplex Speed/duplex array count speed/duplex array Interface speed Interface duplex mode

Common services

Service code	Class	Instance	Service name
0x0E	Yes	Yes	Get_Attribute_Single
0x10	no	Yes	Set_Attribute_Single

7.5 EtherNet/IP assembly instances

The device has several assembly instances for input and output processes with different data lengths.

Input process data

Object class	Assembly instance no.	Byte	Description
0x04	100	0...15	Input assembly 16 bytes

Object class	Assembly instance no.	Byte	Description
0x04	101	0...31	Input assembly 32 bytes
0x04	102	0...63	Input assembly 64 bytes
0x04	103	0...127	Input assembly 128 bytes
0x04	104	0...254	Input assembly 255 bytes_A
0x04	105	0...254	Input assembly 255 bytes_B
0x04	106	0...254	Input assembly 255 bytes_C
0x04	107	0...254	Input assembly 255 bytes_D

Output process data

Object class	Assembly instance no.	Byte	Description
0x04	150	0...9	Output assembly 16 bytes
0x04	151	0...31	Output assembly 32 bytes
0x04	152	0...63	Output assembly 64 bytes
0x04	153	0...127	Output assembly 128 bytes
0x04	154	0...254	Output assembly 255 bytes

7.6 EtherNet/IP connection types

EtherNet/IP connection types

Connection type	Is supported by VSE951	Description
Exclusive owner	Yes	Connections can be configured as multicast or as point-to-point connection in target or originator direction from the scanner.
Input only	Yes	Connections can be configured as multicast or as point-to-point connection in target or originator direction from the scanner.
Listen only	Yes	Connections can only be used as multicast.

7.6.1 EtherNet/IP - defined connections in the standard EDS file

EtherNet/IP connection types

Connection no.	Connection type	Input	Connection no.	Connection type
1	Exclusive owner	100	150	Connection with 16 byte input and 16 byte output data
2	Exclusive owner	101	151	Connection with 32 byte input and 32 byte output data
3	Exclusive owner	102	152	Connection with 64 byte input and 64 byte output data
4	Exclusive owner	103	153	Connection with 128 byte input and 128 byte output data
5	Exclusive owner	104	154	Connection with 255 byte input and 255 byte output data
6	Input only	100	-	Connection with 16 byte input
7	Input only	101	-	Connection with 32 byte input
8	Input only	102	-	Connection with 64 byte input
9	Input only	103	-	Connection with 128 byte input

Connection no.	Connection type	Input	Connection no.	Connection type
10	Input only	104	-	Connection with 255 byte input
11	Input only	105	-	Connection with 255 byte input
12	Input only	106	-	Connection with 255 byte input
13	Input only	107	-	Connection with 255 byte input
14	Listen only	100	-	Connection with 16 byte input
15	Listen only	101	-	Connection with 32 byte input
16	Listen only	102	-	Connection with 64 byte input
17	Listen only	103	-	Connection with 128 byte input
18	Listen only	104	-	Connection with 255 byte input
19	Listen only	105	-	Connection with 255 byte input
20	Listen only	106	-	Connection with 255 byte input
21	Listen only	107	-	Connection with 255 byte input

7.7 EtherNet/IP functions

The following EtherNet/IP functions are supported. All functions that are not listed here are explicitly NOT supported.

7.7.1 Quality of service (QoS)

Request	Parameter
Quality of service	Yes
Description	Quality of service (QoS) affects the forwarding and handling of data streams and results in individual data streams being given differential treatment (usually preferential). QoS can be used to ensure a transmission bandwidth for separate data flows. The device uses QoS in connection with prioritisation.

7.7.2 Device level ring (DLR)

Request	Parameter
Device level ring	Yes
Description	The device supports the development of a redundant 1-ring topology by using the DLR protocol.

7.7.3 Address conflict detection (ACD)

Request	Parameter
Address conflict detection (ACD)	Yes
Description	The device supports the recognition of IPv4 address conflicts.

7.8 EtherNet/IP properties

Request	Parameter
Device type	EtherNet/IP adapter
Transmission rate	10/100 MBit/s (with auto-negotiation)

Request	Parameter
Minimum cycle time	5 ms (RPI)
EtherNet/IP protocols	ACD, DLR, IGMP v2
Additional protocols	DHCP
EtherNet/IP transmission format	Little Endian
Max. I/O connections	10 (1 input and output assembly per connection)
Max. data size	1024 bytes (max. 255 bytes per assembly)
Device description file	ifm_VSE951.eds
Specification	CIP Edition 3.20 EIP adaption of CIP 1.21

7.9 Behaviour if parameter set is changed



Writing of the parameter set (even without changes) or changing the system mode of the diagnostic unit to “set-up” triggers an initialisation (reboot) of the fieldbus module.

The connection of the PLC (master / controller / supervisor) to the diagnostic unit is interrupted. It depends on the programming of the PLC how a connection loss is handled. The LED behaviour is described in the chapter → Operating states of the network (NET) and mode (MOD) status LED.

8 Factory setting

8.1 General factory setting

Request	Parameter
Parameter set	none
Host name	no name assigned
IP address	192.168.0.1
TCP/IP port	3321
Subnet mask	255.255.255.0
Default gateway	192.168.0.244
MAC address	is defined in the production process

8.2 Factory setting VSE951 - EtherNet/IP

There is read and write access to the device settings. The following default values are set by the factory:

Request	Parameter
IP address	192.168.0.100
Subnet mask	255.255.255.0
Gateway	192.168.0.244
Port	502

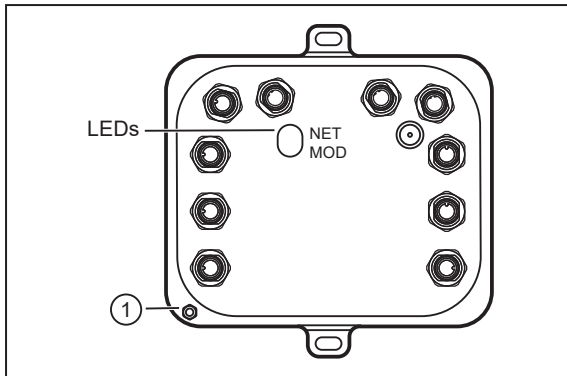
9 Parameter setting

The device parameters are set exclusively via the VES004 PC software. All parameters of the configured application are bundled in a parameter set and transferred to the device.

For a detailed description of all parameters and possible configurations we refer you to the VES004 software manual.

10 Operating and display elements

For quick identification of error states, the device has two diagnostic LEDs on the device front.



1: Earthing screw

10.1 Operating states of the network (NET) and mode (MOD) status LED

Designation	Meaning	Colour	State	Description
LED NET	Network status	n.a.	Off	The device is switched off
		Green	Flashing (approx. 2 Hz)	No connection has been established, an IP address was assigned
		Green	On	Device connected to the network
		Red	On	Error on the fieldbus
LED MOD	EtherNet/IP status	n.a.	Off	Device is switched off (no voltage supply)
		Green	On	Device functions reliably (normal operation)
		Red	On	Device error
		Orange	Flashes	Firmware image is loaded to the RAM
		Orange	Flashes	Firmware image is loaded to the flash
		Green	Flashing for 2 s (approx. 2 Hz)	Firmware image has been written correctly to the flash
		Orange	Flashing for 2 s (approx. 2 Hz)	Parameter set was successfully transferred

11 Maintenance, repair and disposal

The operation of the unit is maintenance-free.

Only the manufacturer is allowed to repair the unit.

- ▶ After use dispose of the device in an environmentally friendly way in accordance with the applicable national regulations.

Cleaning the unit:

- ▶ Disconnect the unit from the voltage supply.
- ▶ Clean the unit from dirt using a soft, chemically untreated and dry micro-fibre cloth.