



EKM Protocol Interface Control Document

Retrieve all measurements from the EKM OmniMeter v4 A and B requests. [Protocol Revision B] - OMNIV4AB_READ_REVB - 26006

Phillip Dillinger <pdillinger@ekmmetering.com>

01 May 2015

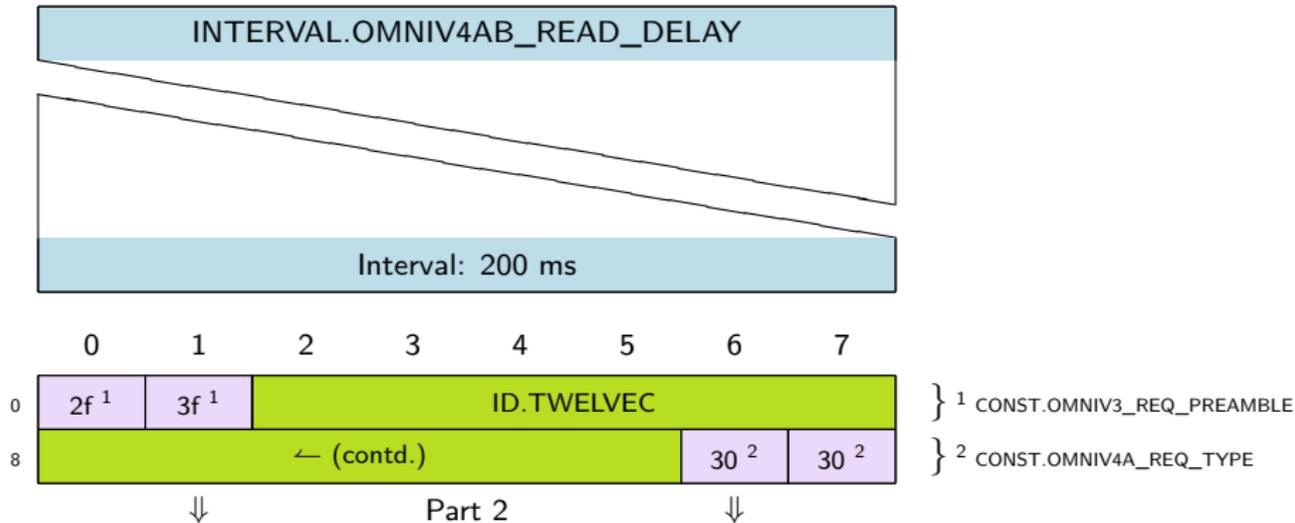
This document describes the EKM OMNIV4AB_READ_REVB [26006] protocol, which is titled "Retrieve all measurements from the EKM OmniMeter v4 A and B requests. [Protocol Revision B]" This document consists of three sections:

- ▶ **Protocol Map**
- ▶ **Protocol Fields**
- ▶ **Schema**

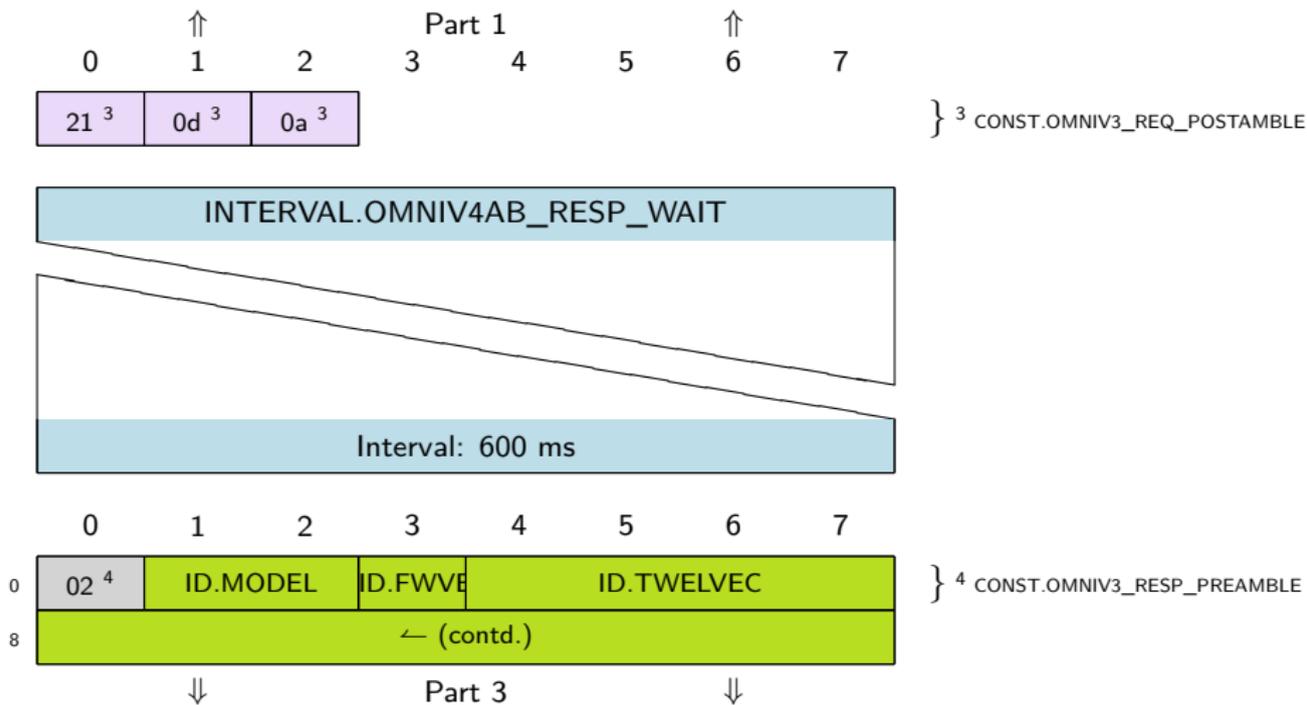
- ▶ **Protocol Map** The protocol map shows the protocol as a diagram. Each field name in the protocol map is a link to corresponding detailed information pages in the Protocol Fields section. The protocol map is color-coded, and the key to these color codes follows. The protocol map is arranged horizontally by words and vertically by groups of an equal number of words. Each word corresponds to a character transmitted over the serial interface. Some fields span vertical groups of words, and these fields are linked together by arrows and "(contd.)" text. The protocol map is read left-to-right and up-to-down, and may be split across multiple pages.

- ▶ **Protocol Fields** The protocol fields section describes each protocol field in detail, and contains information about these fields as returned by the meter in the "raw" format, to the information about these fields as transmitted to the data collection servers, and to the information displayed to users via the web interface. The protocol field documentation provides an overall view of this protocol's data as collected and as viewed.

- ▶ **Schema** The schema section provides schema snippets as generated by the datamodel. These schema snippets are provided as reference information to developers using the data as stored on the data collection servers. When data is retrieved from the data collection servers directly, these schema snippets correspond directly to the data format and characteristics which are to be expected. This schema information is used to auto-generate parts of the device firmware and data collection server software, and as such this schema information is the normative reference of protocol operations and of the data collected.



Protocol Map OMNIV4AB_READ_REVB Part 2

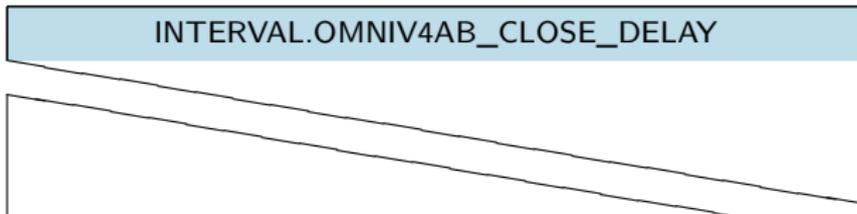
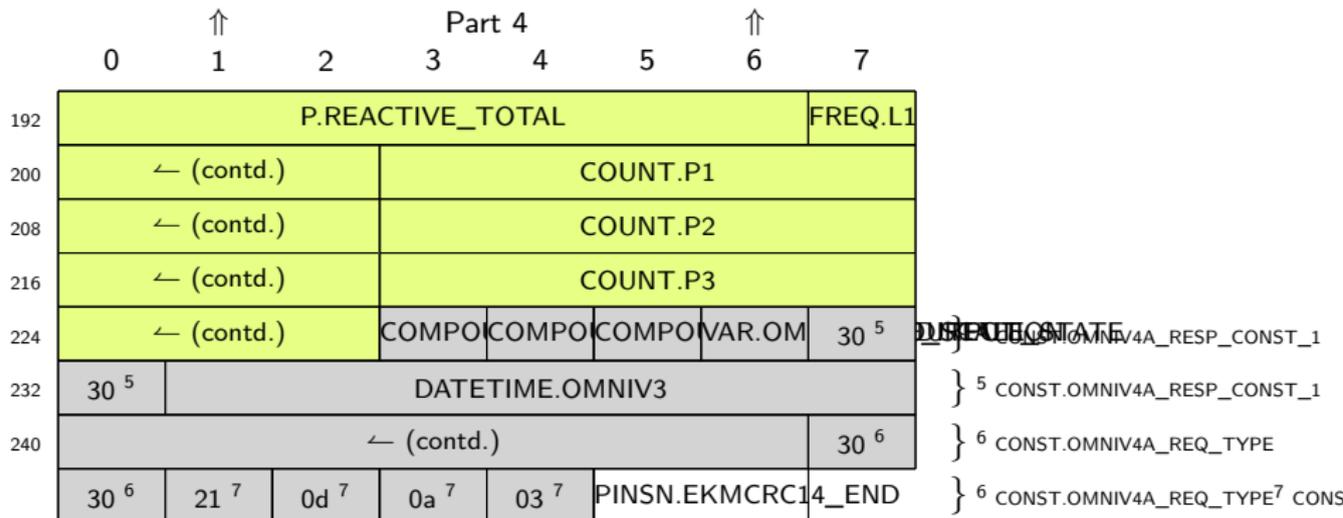


	0	1	2	3	4	5	6	7
16	E.TOTAL							
24	E.REACTIVE_TOTAL							
32	E.UPSTREAM_TOTAL							
40	E.L1							
48	E.L2							
56	E.L3							
64	E.UPSTREAM_L1							
72	E.UPSTREAM_L2							
80	E.UPSTREAM_L3							
88	E.SINCE_RESET							
96	E.UPSTREAM_SINCE_RESET							

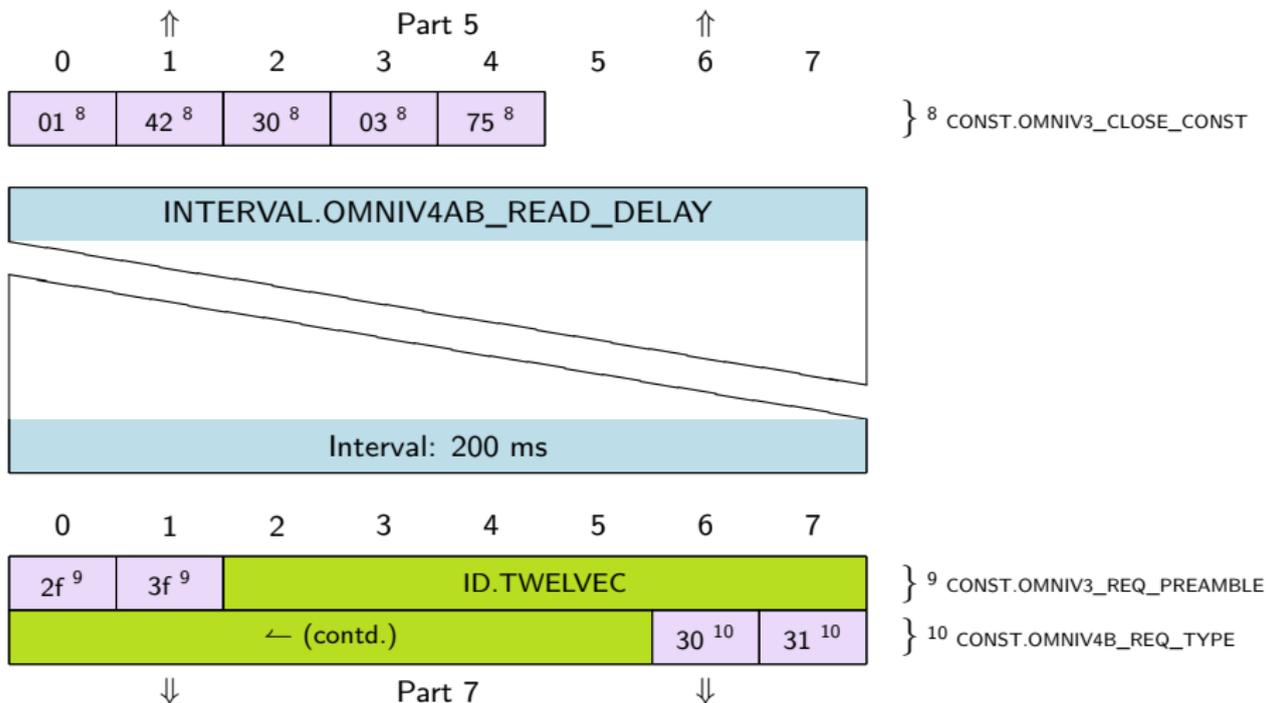
↑
Part 2
↑

↓
Part 4
↓

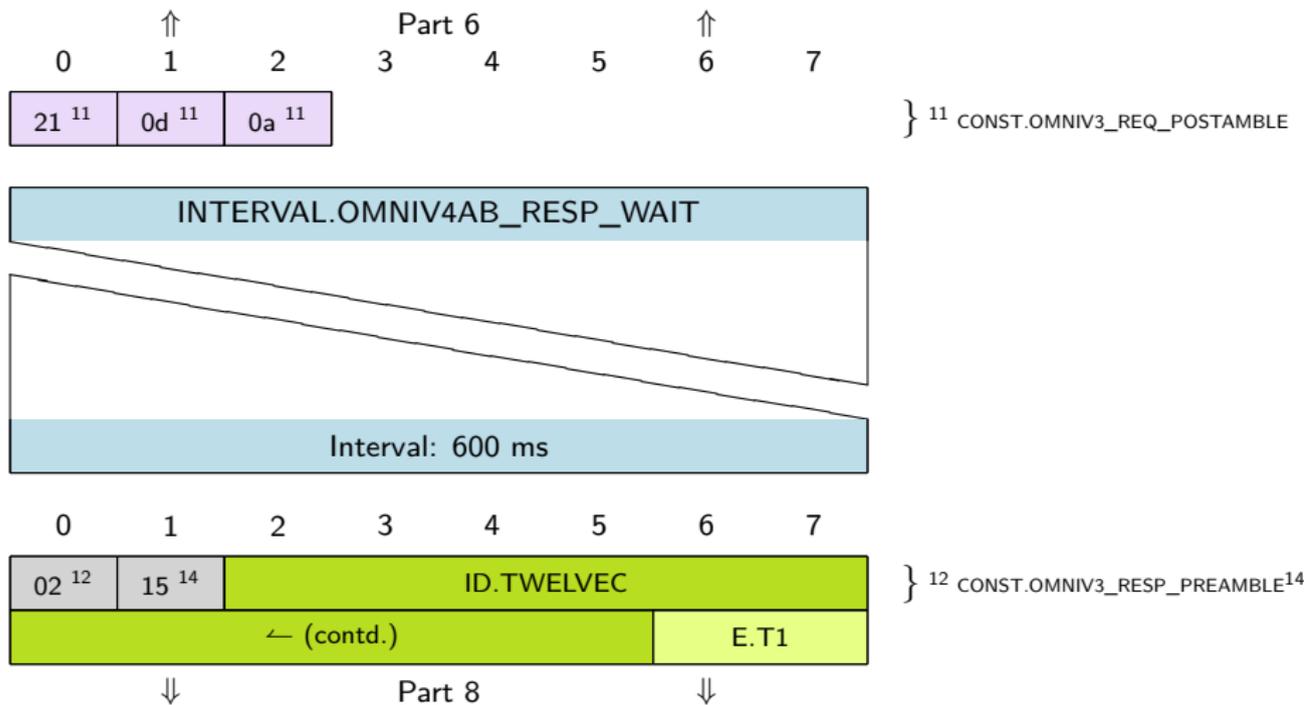
Protocol Map OMNIV4AB_READ_REVB Part 5



Protocol Map OMNIV4AB_READ_REVB Part 6

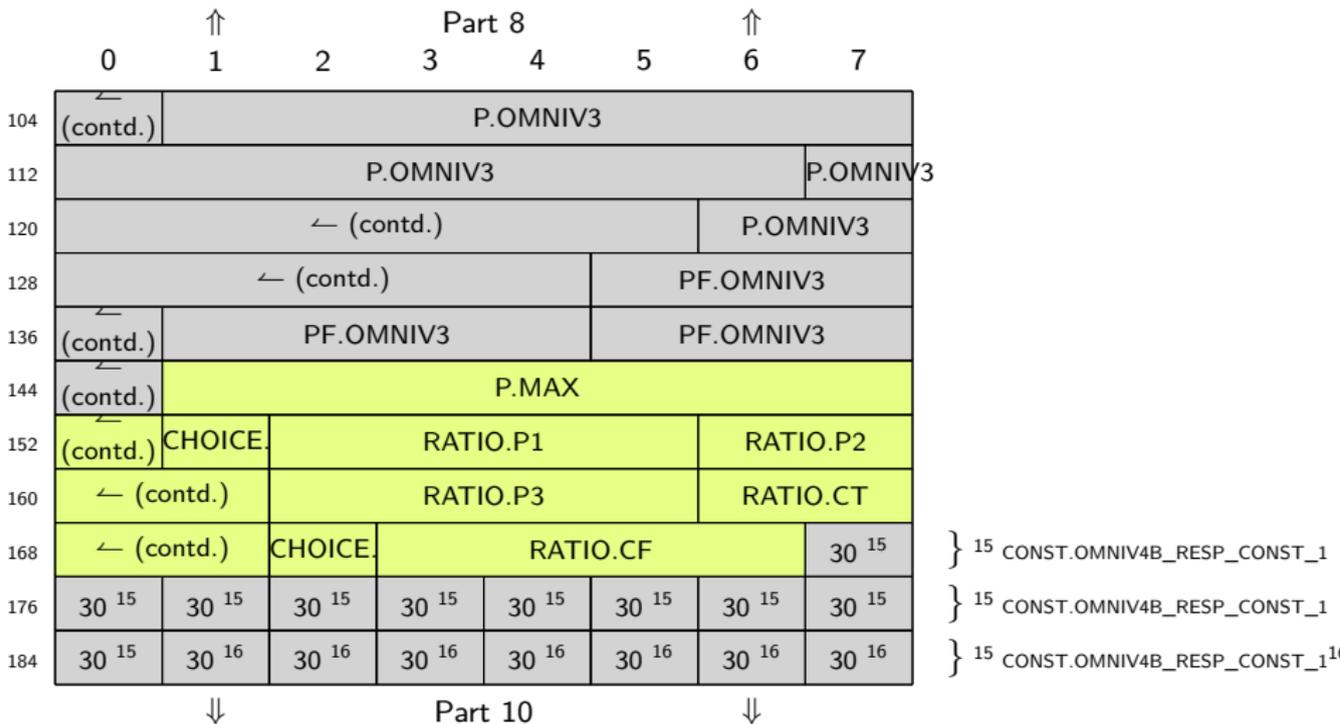


Protocol Map OMNIV4AB_READ_REVB Part 7

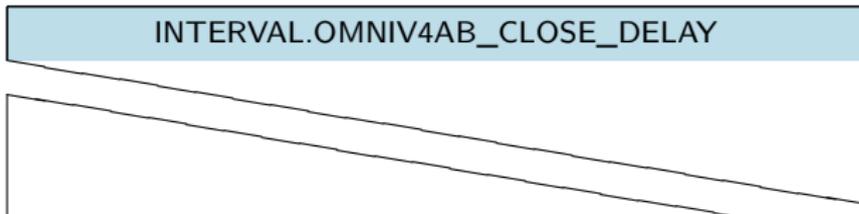


	0	1	2	3	4	5	6	7
16		↑	← (contd.)				E.T2	
24			← (contd.)				E.T3	
32			← (contd.)				E.T4	
40			← (contd.)				E.UPSTREAM_T1	
48			← (contd.)				E.UPSTREAM_T2	
56			← (contd.)				E.UPSTREAM_T3	
64			← (contd.)				E.UPSTREAM_T4	
72			← (contd.)				V.OMNIV3	
80	← (contd.)		V.OMNIV3				V.OMNIV3	
88	← (contd.)		I.OMNIV3					I.OMNIV3
96	← (contd.)			I.OMNIV3				
		↓	Part 9				↓	

Protocol Map OMNIV4AB_READ_REVB Part 9



	0	↑ 1	2	Part 9		5	↑ 6	7		
192	30 ¹⁶	30 ¹⁶	30 ¹⁶	30 ¹⁷	30 ¹⁷	30 ¹⁷	30 ¹⁷	30 ¹⁷	} 16 CONST.OMNIV4B_RESP_CONST_2 ¹⁷	
200	30 ¹⁷	30 ¹⁸	30 ¹⁸	30 ¹⁸	} 17 CONST.OMNIV4B_RESP_CONST_3 ¹⁸					
208	30 ¹⁸	30 ¹⁸	30 ¹⁸	30 ¹⁹	} 18 CONST.OMNIV4B_RESP_CONST_4 ¹⁹					
216	30 ¹⁹	30 ¹⁹	30 ¹⁹	30 ¹⁹	} 19 CONST.OMNIV4B_RESP_CONST_5					
224	30 ¹⁹	30 ²⁰	30 ²⁰	30 ²⁰	30 ²⁰	30 ²⁰	30 ²⁰	DATE TIME) OMNIV3	} 20 CONST.OMNIV4B_RESP_CONST_5 ²⁰	
232	← (contd.) →									
240	← (contd.)						30 ²¹	31 ²¹		} 21 CONST.OMNIV4B_REQ_TYPE
	21 ²²	0d ²²	0a ²²	03 ²²	PINSN.EKMCRC14_END				} 22 CONST.OMNIV3_RESP_POSTAMBLE	





Protocol Field | **EKM OmniMeter v4 delay between reads - INTERVAL.OMNIV4AB_READ_DELAY** - Field ID 27076

EKM OmniMeter v4 delay between reads - INTERVAL.OMNIV4AB_READ_DELAY - Field ID 27076

Technical Description of *EKM OmniMeter v4 delay between reads*

- ▶ **Type of field** *EKM OmniMeter v4 delay between reads* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field does not read from or write to the meter. [NONE]
- ▶ **Format of field** A delay of length 200. [DELAY]

Protocol Field | **EKM OmniMeter v3 and above request preamble -
CONST.OMNIV3_REQ_PREAMBLE - Field ID 26944**

EKM OmniMeter v3 and above request preamble - CONST.OMNIV3_REQ_PREAMBLE - Field ID 26944

Technical Description of *EKM OmniMeter v3 and above request preamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above request preamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =2f=3f

TWELVEC address - ID.TWELVEC - Field ID 6929

Technical Description of *TWELVEC address*

This is a twelve character address of a serial device.

- ▶ **Type of field** *TWELVEC address* is a An identifier. (Kind "ID")
- ▶ **Direction of field** This field is written to the meter from the input message. [GW_THEN_OUT]
- ▶ **Format of field** A sequence of verbatim words of length 12. [WORDS]

Protocol Field | **EKM OmniMeter v4 request type A identifier -
CONST.OMNIV4A_REQ_TYPE** - Field ID 27024

EKM OmniMeter v4 request type A identifier - CONST.OMNIV4A_REQ_TYPE - Field ID 27024

Technical Description of *EKM OmniMeter v4 request type A identifier*

- ▶ **Type of field** *EKM OmniMeter v4 request type A identifier* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =30=30

Protocol Field | **EKM OmniMeter v3 and above request postamble -
CONST.OMNIV3_REQ_POSTAMBLE - Field ID 26945**

**EKM OmniMeter v3 and above request postamble -
CONST.OMNIV3_REQ_POSTAMBLE - Field ID 26945**

Technical Description of *EKM OmniMeter v3 and above request postamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above request postamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 3. [WORDS]
- ▶ **Constant** =21=0d=0a

Protocol Field | **EKM OmniMeter v4 response wait time - INTERVAL.OMNIV4AB_RESP_WAIT** - Field ID 27105

EKM OmniMeter v4 response wait time - INTERVAL.OMNIV4AB_RESP_WAIT - Field ID 27105

Technical Description of *EKM OmniMeter v4 response wait time*

- ▶ **Type of field** *EKM OmniMeter v4 response wait time* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An interval to wait until input is available to read of length 600. [WAIT]

Protocol Field | **EKM OmniMeter v3 and above response preamble -
CONST.OMNIV3_RESP_PREAMBLE - Field ID 26946**

**EKM OmniMeter v3 and above response preamble -
CONST.OMNIV3_RESP_PREAMBLE - Field ID 26946**

Technical Description of *EKM OmniMeter v3 and above response preamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above response preamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 1. [IGNORE]
- ▶ **Constant** =02

Beginning of EKM CRC14 calculation - PINSN.EKMCRC14_BEGIN - Field ID 26948

Technical Description of *Beginning of EKM CRC14 calculation*

- ▶ **Type of field** *Beginning of EKM CRC14 calculation* is a Protocol instruction (Kind "PINSN")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A marker indicating the beginning of the EKM CRC14 calculation of length 0. [EKMCRC14_BEGIN]

Meter model - ID.MODEL - Field ID 2

Technical Description of *Meter model*

The model number as reported by the EKM OmniMeter v3 and above.

- ▶ **Type of field** *Meter model* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant**

Display Description

- ▶ **Display field name** Model

Meter firmware version - ID.FWVER - Field ID 1

Technical Description of *Meter firmware version*

The firmware version as reported by the EKM OmniMeter v3 and above.

- ▶ **Type of field** *Meter firmware version* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** A sequence of verbatim words of length 1. [WORDS]
- ▶ **Constant** =15

Display Description

- ▶ **Display field name** Firmware

TWELVEC address - ID.TWELVEC - Field ID 6929

Technical Description of *TWELVEC address*

This is a twelve character address of a serial device.

- ▶ **Type of field** *TWELVEC address* is a An identifier. (Kind "ID")
- ▶ **Direction of field** This field is read from the meter and checked against a field in the input message. [IN_THEN_CHECK]
- ▶ **Format of field** A sequence of verbatim words of length 12. [WORDS]

Total energy - E.TOTAL - Field ID 3

Technical Description of *Total energy*

Cumulative energy flowing in either direction.

- ▶ **Type of field** *Total energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Total energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Total energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Total energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Total kWh. This value is returned by the meter. It is calculated the sum of all Total kWh 4 tariffs. Total kWh is Forward kWh + Reverse kWh. It is the measure of how much energy has been measured (both Forward and Reverse) since the meter was installed.

- ▶ Display field name kWh_Tot
- ▶ Display scale

Reactive energy - E.REACTIVE_TOTAL - Field ID 38

Technical Description of *Reactive energy*

Cumulative reactive energy measured in either direction through the meter.

- ▶ **Type of field** *Reactive energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Reactive energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Reactive energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Reactive energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Reactive_Energy_Tot
- ▶ **Display scale**

Total upstream energy - E.UPSTREAM_TOTAL - Field ID 8

Technical Description of *Total upstream energy*

Cumulative energy flowing upstream to the grid.

- ▶ **Type of field** *Total upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Total upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Total upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Total upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Reverse kWh. This value is returned by the meter. It is calculated the sum of all four Reverse kWh tariffs. Reverse kWh is the measure of how much energy has been measured going toward the grid since the meter was installed.

- ▶ **Display field name** Rev_kWh_Tot
- ▶ **Display scale**

Line 1 energy - E.L1 - Field ID 51

Technical Description of *Line 1 energy*

Cumulative energy flowing in either direction on line 1.

- ▶ **Type of field** *Line 1 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 1 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 1 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 1 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Total kWh on each line. This is the Total kWh (Forward kWh + Reverse kWh) measured on Line 1 since the meter was installed.

- ▶ **Display field name** kWh_Ln_1
- ▶ **Display scale**

Line 2 energy - E.L2 - Field ID 52

Technical Description of *Line 2 energy*

Cumulative energy flowing in either direction on line 2.

- ▶ **Type of field** *Line 2 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 2 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 2 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 2 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Total kWh on each line. This is the Total kWh (Forward kWh + Reverse kWh) measured on Line 2 since the meter was installed.

- ▶ **Display field name** kWh_Ln_2
- ▶ **Display scale**

Line 3 energy - E.L3 - Field ID 53

Technical Description of *Line 3 energy*

Cumulative energy flowing in either direction on line 3.

- ▶ **Type of field** *Line 3 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 3 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 3 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 3 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Total kWh on each line. This is the Total kWh (Forward kWh + Reverse kWh) measured on Line 3 since the meter was installed.

- ▶ **Display field name** kWh_Ln_3
- ▶ **Display scale**

Line 1 upstream energy - E.UPSTREAM_L1 - Field ID 54

Technical Description of *Line 1 upstream energy*

Cumulative energy flowing upstream to the grid on line 1.

- ▶ **Type of field** *Line 1 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 1 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 1 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 1 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Reverse kWh on each line. This is the Reverse kWh measured on Line 1 since the meter was installed.

- ▶ **Display field name** Rev_kWh_Ln_1
- ▶ **Display scale**

Line 2 upstream energy - E.UPSTREAM_L2 - Field ID 55

Technical Description of *Line 2 upstream energy*

Cumulative energy flowing upstream to the grid on line 2.

- ▶ **Type of field** *Line 2 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 2 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 2 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 2 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Reverse kWh on each line. This is the Reverse kWh measured on Line 2 since the meter was installed.

- ▶ **Display field name** Rev_kWh_Ln_2
- ▶ **Display scale**

Line 3 upstream energy - E.UPSTREAM_L3 - Field ID 56

Technical Description of *Line 3 upstream energy*

Cumulative energy flowing upstream to the grid on line 3.

- ▶ **Type of field** *Line 3 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Line 3 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Line 3 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Line 3 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

The v4 Meter keeps track of Reverse kWh on each line. This is the Reverse kWh measured on Line 3 since the meter was installed.

- ▶ **Display field name** Rev_kWh_Ln_3
- ▶ **Display scale**

Energy since reset - E.SINCE_RESET - Field ID 39

Technical Description of *Energy since reset*

Cumulative energy flowing in either direction since last reset of this field.

- ▶ **Type of field** *Energy since reset* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Energy since reset* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Energy since reset* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Energy since reset* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

This is Resettable Total kWh. This is similar to the Total kWh register, except that this one can be reset to 0 over RS485 using our EKM Dash software and a USB to RS485 converter. Similar to a trip odometer on your car.

- ▶ Display field name kWh_Rst
- ▶ Display scale

Upstream energy since reset - E.UPSTREAM_SINCE_RESET - Field ID 40

Technical Description of *Upstream energy since reset*

Cumulative energy flowing upstream to the grid since last reset of this field.

- ▶ **Type of field** *Upstream energy since reset* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Upstream energy since reset* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Upstream energy since reset* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Upstream energy since reset* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

This Is Resettable Reverse kWh. This is similar to the Reverse kWh register, except that this one can be reset to 0 over RS485 using our EKM Dash software and a USB to RS485 converter. Similar to a trip odometer on your car.

- ▶ **Display field name** Rev_kWh_Rst
- ▶ **Display scale**

Line 1 voltage - V.L1 - Field ID 13

Technical Description of *Line 1 voltage*

The RMS voltage across line 1 and neutral.

- ▶ **Type of field** *Line 1 voltage* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Input Scale** *Line 1 voltage* is read from the meter in units of 0.1 V
- ▶ **Output Scale** *Line 1 voltage* is displayed and stored in units of 1 V
- ▶ **Field Range** *Line 1 voltage* as displayed and stored may have a value of 0 to 999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

This is the measure of the voltage from Line 1 to Neutral. Voltage is calculated by the meter as RMS Volts. This is a real-time value.

- ▶ **Display field name** RMS_Volts_Ln_1
- ▶ **Display scale**

Line 2 voltage - V.L2 - Field ID 14

Technical Description of *Line 2 voltage*

The RMS voltage across line 2 and neutral.

- ▶ **Type of field** *Line 2 voltage* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Input Scale** *Line 2 voltage* is read from the meter in units of 0.1 V
- ▶ **Output Scale** *Line 2 voltage* is displayed and stored in units of 1 V
- ▶ **Field Range** *Line 2 voltage* as displayed and stored may have a value of 0 to 999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

This is the measure of the voltage from Line 2 to Neutral. Voltage is calculated by the meter as RMS Volts. This is a real-time value.

- ▶ **Display field name** RMS_Volts_Ln_2
- ▶ **Display scale**

Line 3 voltage - V.L3 - Field ID 15

Technical Description of *Line 3 voltage*

The RMS voltage across line 3 and neutral.

- ▶ **Type of field** *Line 3 voltage* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Input Scale** *Line 3 voltage* is read from the meter in units of 0.1 V
- ▶ **Output Scale** *Line 3 voltage* is displayed and stored in units of 1 V
- ▶ **Field Range** *Line 3 voltage* as displayed and stored may have a value of 0 to 999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

This is the measure of the voltage from Line 3 to Neutral. Voltage is calculated by the meter as RMS Volts. This is a real-time value.

- ▶ **Display field name** RMS_Volts_Ln_3
- ▶ **Display scale**

Line 1 current - I.L1 - Field ID 16

Technical Description of *Line 1 current*

The RMS current flowing in either direction through line 1.

- ▶ **Type of field** *Line 1 current* is a Measurement of RMS current. (Kind "I")
- ▶ **Input Scale** *Line 1 current* is read from the meter in units of 0.1 A
- ▶ **Output Scale** *Line 1 current* is displayed and stored in units of 1 A
- ▶ **Field Range** *Line 1 current* as displayed and stored may have a value of 0 to 9999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

Display Description

This is the measure of Amps or Current on Line 1. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name Amps_Ln_1scale=-1
- ▶ Display scale

Line 2 current - I.L2 - Field ID 17

Technical Description of *Line 2 current*

The RMS current flowing in either direction through line 2.

- ▶ **Type of field** *Line 2 current* is a Measurement of RMS current. (Kind "I")
- ▶ **Input Scale** *Line 2 current* is read from the meter in units of 0.1 A
- ▶ **Output Scale** *Line 2 current* is displayed and stored in units of 1 A
- ▶ **Field Range** *Line 2 current* as displayed and stored may have a value of 0 to 9999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

Display Description

This is the measure of Amps or Current on Line 2. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name Amps_Ln_2
- ▶ Display scale

Line 3 current - I.L3 - Field ID 18

Technical Description of *Line 3 current*

The RMS current flowing in either direction through line 3.

- ▶ **Type of field** *Line 3 current* is a Measurement of RMS current. (Kind "I")
- ▶ **Input Scale** *Line 3 current* is read from the meter in units of 0.1 A
- ▶ **Output Scale** *Line 3 current* is displayed and stored in units of 1 A
- ▶ **Field Range** *Line 3 current* as displayed and stored may have a value of 0 to 9999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

Display Description

This is the measure of Amps or Current on Line 3. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name Amps_Ln_3
- ▶ Display scale

Line 1 power - P.L1 - Field ID 19

Technical Description of *Line 1 power*

The RMS power flowing in either direction through line 1.

- ▶ **Type of field** *Line 1 power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 1 power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 1 power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 1 power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

This is the measure of Watts or Power on Line 1. This is a value that is calculated by the meter. $\text{Watts} = \text{Volts} \times \text{Amps} \times \text{Power Factor}$ on Line 1. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name RMS_Watts_Ln_1
- ▶ Display scale

Line 2 power - P.L2 - Field ID 20

Technical Description of *Line 2 power*

The RMS power flowing in either direction through line 2.

- ▶ **Type of field** *Line 2 power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 2 power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 2 power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 2 power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

This is the measure of Watts or Power on Line 2. This is a value that is calculated by the meter. $\text{Watts} = \text{Volts} \times \text{Amps} \times \text{Power Factor}$ on Line 2. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name RMS_Watts_Ln_2
- ▶ Display scale

Line 3 power - P.L3 - Field ID 21

Technical Description of *Line 3 power*

The RMS power flowing in either direction through line 3.

- ▶ **Type of field** *Line 3 power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 3 power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 3 power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 3 power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

This is the measure of Watts or Power on Line 3. This is a value that is calculated by the meter. $\text{Watts} = \text{Volts} \times \text{Amps} \times \text{Power Factor}$ on Line 3. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name RMS_Watts_Ln_3
- ▶ Display scale

Total power - P.TOTAL - Field ID 22

Technical Description of *Total power*

The RMS power flowing in either direction.

- ▶ **Type of field** *Total power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Total power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Total power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Total power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

This is the measure of Total Watts or Power. This is a value that is calculated by the meter. Total Watts = (Volts x Amps x Power Factor) on all Lines. This value is calculated by the meter. This is a real-time value.

- ▶ Display field name RMS_Watts_Tot
- ▶ Display scale

Line 1 power factor - PF.L1 - Field ID 23

Technical Description of *Line 1 power factor*

The current load power factor measurement for line 1.

- ▶ **Type of field** *Line 1 power factor* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Input Scale** *Line 1 power factor* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Line 1 power factor* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Line 1 power factor* as displayed and stored may have a value of 0 to 200
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4.
[OMNIV3_PF]

Display Description

Power Factor is a measure of power quality on Line 1. This is Cosine Theta. A perfect Power Factor (1.00) is when the voltage sine wave and the amperage sine wave are in perfect alignment on top of each other. When the voltage and amperage sine waves are off of each other then you have less than perfect Power Factor. When voltage is leading amps it is considered leading (Inductive or L), when voltage is lagging amps it is considered lagging (Capacitive or C)

- ▶ **Display field name** Power_Factor_Ln_1
- ▶ **Display scale**

Line 2 power factor - PF.L2 - Field ID 24

Technical Description of *Line 2 power factor*

The current load power factor measurement for line 2.

- ▶ **Type of field** *Line 2 power factor* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Input Scale** *Line 2 power factor* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Line 2 power factor* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Line 2 power factor* as displayed and stored may have a value of 0 to 200
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4.
[OMNIV3_PF]

Display Description

Power Factor is a measure of power quality on Line 2. This is Cosine Theta. A perfect Power Factor (1.00) is when the voltage sine wave and the amperage sine wave are in perfect alignment on top of each other. When the voltage and amperage sine waves are off of each other then you have less than perfect Power Factor. When voltage is leading amps it is considered leading (Inductive or L), when voltage is lagging amps it is considered lagging (Capacitive or C)

- ▶ **Display field name** Power_Factor_Ln_2
- ▶ **Display scale**

Line 3 power factor - PF.L3 - Field ID 25

Technical Description of *Line 3 power factor*

The current load power factor measurement for line 3.

- ▶ **Type of field** *Line 3 power factor* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Input Scale** *Line 3 power factor* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Line 3 power factor* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Line 3 power factor* as displayed and stored may have a value of 0 to 200
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4.
[OMNIV3_PF]

Display Description

Power Factor is a measure of power quality on Line 3. This is Cosine Theta. A perfect Power Factor (1.00) is when the voltage sine wave and the amperage sine wave are in perfect alignment on top of each other. When the voltage and amperage sine waves are off of each other then you have less than perfect Power Factor. When voltage is leading amps it is considered leading (Inductive or L), when voltage is lagging amps it is considered lagging (Capacitive or C)

- ▶ **Display field name** Power_Factor_Ln_3
- ▶ **Display scale**

Line 1 reactive power - P.REACTIVE_L1 - Field ID 41

Technical Description of *Line 1 reactive power*

The RMS reactive power flowing in either direction through line 1.

- ▶ **Type of field** *Line 1 reactive power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 1 reactive power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 1 reactive power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 1 reactive power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Reactive_Pwr_Ln_1
- ▶ **Display scale**

Line 2 reactive power - P.REACTIVE_L2 - Field ID 42

Technical Description of *Line 2 reactive power*

The RMS reactive power flowing in either direction through line 2.

- ▶ **Type of field** *Line 2 reactive power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 2 reactive power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 2 reactive power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 2 reactive power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Reactive_Pwr_Ln_2
- ▶ **Display scale**

Line 3 reactive power - P.REACTIVE_L3 - Field ID 43

Technical Description of *Line 3 reactive power*

The RMS reactive power flowing in either direction through line 3.

- ▶ **Type of field** *Line 3 reactive power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Line 3 reactive power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Line 3 reactive power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Line 3 reactive power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Reactive_Pwr_Ln_3
- ▶ **Display scale**

Total reactive power - P.REACTIVE_TOTAL - Field ID 44

Technical Description of *Total reactive power*

The RMS reactive power currently flowing in either direction.

- ▶ **Type of field** *Total reactive power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Total reactive power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Total reactive power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Total reactive power* as displayed and stored may have a value of 0 to 9999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Reactive_Pwr_Tot
- ▶ **Display scale**

Line 1 frequency - FREQ.L1 - Field ID 47

Technical Description of *Line 1 frequency*

The frequency of AC power flowing through line 1.

- ▶ **Type of field** *Line 1 frequency* is a Measurement of frequency. (Kind "FREQ")
- ▶ **Input Scale** *Line 1 frequency* is read from the meter in units of 0.1 Hz
- ▶ **Output Scale** *Line 1 frequency* is displayed and stored in units of 1 Hz
- ▶ **Field Range** *Line 1 frequency* as displayed and stored may have a value of 0 to 999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Line_Freq
- ▶ **Display scale**

Pulse input 1 pulse count - COUNT.P1 - Field ID 31

Technical Description of *Pulse input 1 pulse count*

Cumulative pulse count for pulse input 1. The reported pulse count increments once per *RATIO.P1* actual counted pulses.

- ▶ **Type of field** *Pulse input 1 pulse count* is a Cumulative count of events. (Kind "COUNT")
- ▶ **Input Scale** *Pulse input 1 pulse count* is read from the meter in units of 1 NONE
- ▶ **Output Scale** *Pulse input 1 pulse count* is displayed and stored in units of NONE
- ▶ **Field Range** *Pulse input 1 pulse count* as displayed and stored may have a value of 0 to 99999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Pulse Count 1 is the number of contact closures or Pulse Counts the meter has measured divided by the Pulse Count Ratio. If the Pulse Count Ratio is 1:1000, it will take 1000 inputs to increment this register 1. This can be used to count pulses from Water Meters or Gas Meters, or other pulse output devices.

- ▶ **Display field name** Pulse_Cnt_1
- ▶ **Display scale**

Pulse input 2 pulse count - COUNT.P2 - Field ID 32

Technical Description of *Pulse input 2 pulse count*

Cumulative pulse count for pulse input 2. The reported pulse count increments once per RATIO.P2 actual counted pulses.

- ▶ **Type of field** *Pulse input 2 pulse count* is a Cumulative count of events. (Kind "COUNT")
- ▶ **Input Scale** *Pulse input 2 pulse count* is read from the meter in units of 1 NONE
- ▶ **Output Scale** *Pulse input 2 pulse count* is displayed and stored in units of NONE
- ▶ **Field Range** *Pulse input 2 pulse count* as displayed and stored may have a value of 0 to 99999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Pulse Count 2 is the number of contact closures or Pulse Counts the meter has measured divided by the Pulse Count Ratio. If the Pulse Count Ratio is 1:1000, it will take 1000 inputs to increment this register 1. This can be used to count pulses from Water Meters or Gas Meters, or other pulse output devices.

- ▶ Display field name Pulse_Cnt_2
- ▶ Display scale

Pulse input 3 pulse count - COUNT.P3 - Field ID 33

Technical Description of *Pulse input 3 pulse count*

Cumulative pulse count for pulse input 3. The reported pulse count increments once per `RATIO.P3` actual counted pulses.

- ▶ **Type of field** *Pulse input 3 pulse count* is a Cumulative count of events. (Kind "COUNT")
- ▶ **Input Scale** *Pulse input 3 pulse count* is read from the meter in units of 1 NONE
- ▶ **Output Scale** *Pulse input 3 pulse count* is displayed and stored in units of NONE
- ▶ **Field Range** *Pulse input 3 pulse count* as displayed and stored may have a value of 0 to 99999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Pulse Count 3 is the number of contact closures or Pulse Counts the meter has measured divided by the Pulse Count Ratio. If the Pulse Count Ratio is 1:1000, it will take 1000 inputs to increment this register 1. This can be used to count pulses from Water Meters or Gas Meters, or other pulse output devices.

- ▶ **Display field name** Pulse_Cnt_3
- ▶ **Display scale**

Protocol Field | **EKM OmniMeter v4 input states -
COMPOUND.OMNIV4A_INPUT_STATE - Field ID 27035**

**EKM OmniMeter v4 input states -
COMPOUND.OMNIV4A_INPUT_STATE - Field ID 27035**

Protocol Field II EKM OmniMeter v4 input states - COMPOUND.OMNIV4A_INPUT_STATE - Field ID 27035

Technical Description of *EKM OmniMeter v4 input states*

- ▶ **Type of field** *EKM OmniMeter v4 input states* is a Single value representing multiple values across multiple fields, which is expanded into multiple fields and values. (Kind "COMPOUND")
- ▶ **Direction of field** This field is read from the meter and inserted as several fields in the output message. [IN_THEN_GW_COMPOUND]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Field Value	CHOICE.STATE_P1	CHOICE.STATE_P2	CHOICE.STATE_P3
0	1	1	1
1	1	1	0
2	1	0	1
3	1	0	0
4	0	1	1
5	0	1	0
6	0	0	1
7	0	0	0

Protocol Field | EKM OmniMeter v4 power directions -

COMPOUND.OMNIV4A_POWER_DIRECTION - Field ID 27036

EKM OmniMeter v4 power directions -

COMPOUND.OMNIV4A_POWER_DIRECTION - Field ID 27036

Protocol Field II EKM OmniMeter v4 power directions - COMPOUND.OMNIV4A_POWER_DIRECTION - Field ID 27036

Technical Description of *EKM OmniMeter v4 power directions*

- ▶ **Type of field** *EKM OmniMeter v4 power directions* is a Single value representing multiple values across multiple fields, which is expanded into multiple fields and values. (Kind "COMPOUND")
- ▶ **Direction of field** This field is read from the meter and inserted as several fields in the output message. [IN_THEN_GW_COMPOUND]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Field Value	CHOICE.DIRECTION_L1	CHOICE.DIRECTION_L2	CHOICE.DIRECTION_L3
1	1	1	1
2	1	1	0
3	1	0	1
4	0	1	1
5	1	0	0
6	0	1	0
7	0	0	1
8	0	0	0

Protocol Field | EKM OmniMeter v4 output states - COMPOUND.OMNIV4A_OUTPUT_STATE - Field ID 27037

EKM OmniMeter v4 output states - COMPOUND.OMNIV4A_OUTPUT_STATE - Field ID 27037

Technical Description of *EKM OmniMeter v4 output states*

- ▶ **Type of field** *EKM OmniMeter v4 output states* is a Single value representing multiple values across multiple fields, which is expanded into multiple fields and values. (Kind "COMPOUND")
- ▶ **Direction of field** This field is read from the meter and inserted as several fields in the output message. [IN_THEN_GW_COMPOUND]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Field Value	CHOICE.STATE_SW1	CHOICE.STATE_SW2
1	0	0
2	0	1
3	1	0
4	1	1

EKM OmniMeter v4 energy scale field - VAR.OMNIV4A_E_SCALE - Field ID 27047

Technical Description of *EKM OmniMeter v4 energy scale field*

- ▶ **Type of field** *EKM OmniMeter v4 energy scale field* is a A variable (Kind "VAR")
- ▶ **Direction of field** This field is read from the meter and stored as a variable for future use. [IN_THEN_STORE]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Protocol Field | **EKM OmniMeter v4 response type A constant 1 -
CONST.OMNIV4A_RESP_CONST_1 - Field ID 27033**

EKM OmniMeter v4 response type A constant 1 - CONST.OMNIV4A_RESP_CONST_1 - Field ID 27033

Technical Description of *EKM OmniMeter v4 response type A constant 1*

- ▶ **Type of field** *EKM OmniMeter v4 response type A constant 1* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 2. [IGNORE]
- ▶ **Constant** =30=30

Protocol Field | **EKM OmniMeter v3 and above date and time field - DATETIME.OMNIV3 - Field ID 26937**

EKM OmniMeter v3 and above date and time field - DATETIME.OMNIV3 - Field ID 26937

Technical Description of *EKM OmniMeter v3 and above date and time field*

- ▶ **Type of field** *EKM OmniMeter v3 and above date and time field* is a Date and time measured since 1Jan1970 UTC, without leap seconds (Kind "DATETIME")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A date as formatted by the EKM OmniMeter v3 and later meters of length 14. [OMNIV3_DATE]

Protocol Field | EKM OmniMeter v4 request type A identifier -
CONST.OMNIV4A_REQ_TYPE - Field ID 27024

EKM OmniMeter v4 request type A identifier - CONST.OMNIV4A_REQ_TYPE - Field ID 27024

Technical Description of *EKM OmniMeter v4 request type A identifier*

- ▶ **Type of field** *EKM OmniMeter v4 request type A identifier* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =30=30

Protocol Field | **EKM OmniMeter v3 and above response postamble -
CONST.OMNIV3_RESP_POSTAMBLE - Field ID 26947**

**EKM OmniMeter v3 and above response postamble -
CONST.OMNIV3_RESP_POSTAMBLE - Field ID 26947**

Technical Description of *EKM OmniMeter v3 and above response postamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above response postamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 4. [IGNORE]
- ▶ **Constant** =21=0d=0a=03

Protocol Field | End of EKM CRC14 calculation and check against computed value or generate computed value - PINSN.EKMCRC14_END - Field ID 26949

End of EKM CRC14 calculation and check against computed value or generate computed value - PINSN.EKMCRC14_END - Field ID 26949

Technical Description of *End of EKM CRC14 calculation and check against computed value or generate computed value*

- ▶ **Type of field** *End of EKM CRC14 calculation and check against computed value or generate computed value* is a Protocol instruction (Kind "PINSN")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A two word EKM CRC14 value and the EKM CRC14 check of the calculated value against the field as read from the meter of length 2. [EKMCRC14_END]

Protocol Field | **EKM OmniMeter v4 delay before sending close string - INTERVAL.OMNIV4AB_CLOSE_DELAY - Field ID 27081**

EKM OmniMeter v4 delay before sending close string - INTERVAL.OMNIV4AB_CLOSE_DELAY - Field ID 27081

Technical Description of *EKM OmniMeter v4 delay before sending close string*

- ▶ **Type of field** *EKM OmniMeter v4 delay before sending close string* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field does not read from or write to the meter. [NONE]
- ▶ **Format of field** A delay of length 200. [DELAY]

Protocol Field | **EKM OmniMeter v3 and above close string -
CONST.OMNIV3_CLOSE_CONST - Field ID 26992**

EKM OmniMeter v3 and above close string - CONST.OMNIV3_CLOSE_CONST - Field ID 26992

Technical Description of *EKM OmniMeter v3 and above close string*

- ▶ **Type of field** *EKM OmniMeter v3 and above close string* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of input words to ignore of length 5. [IGNORE]
- ▶ **Constant** =01=42=30=03=75

Protocol Field | **EKM OmniMeter v4 delay between reads - INTERVAL.OMNIV4AB_READ_DELAY** - Field ID 27076

EKM OmniMeter v4 delay between reads - INTERVAL.OMNIV4AB_READ_DELAY - Field ID 27076

Technical Description of *EKM OmniMeter v4 delay between reads*

- ▶ **Type of field** *EKM OmniMeter v4 delay between reads* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field does not read from or write to the meter. [NONE]
- ▶ **Format of field** A delay of length 200. [DELAY]

Protocol Field | **EKM OmniMeter v3 and above request preamble -**
CONST.OMNIV3_REQ_PREAMBLE - Field ID 26944

EKM OmniMeter v3 and above request preamble - **CONST.OMNIV3_REQ_PREAMBLE - Field ID 26944**

Technical Description of *EKM OmniMeter v3 and above request preamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above request preamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =2f=3f

TWELVEC address - ID.TWELVEC - Field ID 6929

Technical Description of *TWELVEC address*

This is a twelve character address of a serial device.

- ▶ **Type of field** *TWELVEC address* is a An identifier. (Kind "ID")
- ▶ **Direction of field** This field is written to the meter from the input message. [GW_THEN_OUT]
- ▶ **Format of field** A sequence of verbatim words of length 12. [WORDS]

Protocol Field | EKM OmniMeter v4 request type B identifier -
CONST.OMNIV4B_REQ_TYPE - Field ID 27054

EKM OmniMeter v4 request type B identifier - CONST.OMNIV4B_REQ_TYPE - Field ID 27054

Technical Description of *EKM OmniMeter v4 request type B identifier*

- ▶ **Type of field** *EKM OmniMeter v4 request type B identifier* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =30=31

Protocol Field | **EKM OmniMeter v3 and above request postamble -
CONST.OMNIV3_REQ_POSTAMBLE - Field ID 26945**

**EKM OmniMeter v3 and above request postamble -
CONST.OMNIV3_REQ_POSTAMBLE - Field ID 26945**

Technical Description of *EKM OmniMeter v3 and above request postamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above request postamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of verbatim words of length 3. [WORDS]
- ▶ **Constant** =21=0d=0a

Protocol Field | **EKM OmniMeter v4 response wait time - INTERVAL.OMNIV4AB_RESP_WAIT** - Field ID 27105

EKM OmniMeter v4 response wait time - INTERVAL.OMNIV4AB_RESP_WAIT - Field ID 27105

Technical Description of *EKM OmniMeter v4 response wait time*

- ▶ **Type of field** *EKM OmniMeter v4 response wait time* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An interval to wait until input is available to read of length 600. [WAIT]

Protocol Field | **EKM OmniMeter v3 and above response preamble -
CONST.OMNIV3_RESP_PREAMBLE - Field ID 26946**

**EKM OmniMeter v3 and above response preamble -
CONST.OMNIV3_RESP_PREAMBLE - Field ID 26946**

Technical Description of *EKM OmniMeter v3 and above response preamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above response preamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 1. [IGNORE]
- ▶ **Constant** =02

Beginning of EKM CRC14 calculation - PINSN.EKMCRC14_BEGIN - Field ID 26948

Technical Description of *Beginning of EKM CRC14 calculation*

- ▶ **Type of field** *Beginning of EKM CRC14 calculation* is a Protocol instruction (Kind "PINSN")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A marker indicating the beginning of the EKM CRC14 calculation of length 0. [EKMCRC14_BEGIN]

Protocol Field | **EKM OmniMeter v3 and above model number -
CONST.OMNIV3_MODEL - Field ID 26940**

EKM OmniMeter v3 and above model number - CONST.OMNIV3_MODEL - Field ID 26940

Technical Description of *EKM OmniMeter v3 and above model number*

- ▶ **Type of field** *EKM OmniMeter v3 and above model number* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant**

EKM OmniMeter v4 firmware version - CONST.OMNIV4_FWVER - Field ID 27104

Technical Description of *EKM OmniMeter v4 firmware version*

- ▶ **Type of field** *EKM OmniMeter v4 firmware version* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of verbatim words of length 1. [WORDS]
- ▶ **Constant** =15

TWELVEC address - ID.TWELVEC - Field ID 6929

Technical Description of *TWELVEC address*

This is a twelve character address of a serial device.

- ▶ **Type of field** *TWELVEC address* is a An identifier. (Kind "ID")
- ▶ **Direction of field** This field is read from the meter and checked against a field in the input message. [IN_THEN_CHECK]
- ▶ **Format of field** A sequence of verbatim words of length 12. [WORDS]

Tariff 1 energy - E.T1 - Field ID 4

Technical Description of *Tariff 1 energy*

Cumulative energy flowing in either direction during the Tariff 1 time period.

- ▶ **Type of field** *Tariff 1 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 1 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 1 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 1 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Total kWh Tariff 1. This value is returned by the meter. It is the measure of how much energy has been measured (both Forward and Reverse) since the meter was installed within the Tariff 1 period. The Tariff 1 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ Display field name kWh_Tariff_1
- ▶ Display scale

Tariff 2 energy - E.T2 - Field ID 5

Technical Description of *Tariff 2 energy*

Cumulative energy flowing in either direction during the Tariff 2 time period.

- ▶ **Type of field** *Tariff 2 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 2 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 2 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 2 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Total kWh Tariff 2. This value is returned by the meter. It is the measure of how much energy has been measured (both Forward and Reverse) since the meter was installed within the Tariff 2 period. The Tariff 2 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** kWh_Tariff_2
- ▶ **Display scale**

Tariff 3 energy - E.T3 - Field ID 6

Technical Description of *Tariff 3 energy*

Cumulative energy flowing in either direction during the Tariff 3 time period.

- ▶ **Type of field** *Tariff 3 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 3 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 3 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 3 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Total kWh Tariff 3. This value is returned by the meter. It is the measure of how much energy has been measured (both Forward and Reverse) since the meter was installed within the Tariff 3 period. The Tariff 3 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ Display field name kWh_Tariff_3
- ▶ Display scale

Tariff 4 energy - E.T4 - Field ID 7

Technical Description of *Tariff 4 energy*

Cumulative energy flowing in either direction during the Tariff 4 time period.

- ▶ **Type of field** *Tariff 4 energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 4 energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 4 energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 4 energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Total kWh Tariff 4. This value is returned by the meter. It is the measure of how much energy has been measured (both Forward and Reverse) since the meter was installed within the Tariff 4 period. The Tariff 4 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ Display field name kWh_Tariff_4
- ▶ Display scale

Tariff 1 upstream energy - E.UPSTREAM_T1 - Field ID 9

Technical Description of *Tariff 1 upstream energy*

Cumulative energy flowing upstream to the grid during the Tariff 1 time period.

- ▶ **Type of field** *Tariff 1 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 1 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 1 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 1 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Reverse kWh Tariff 1. This value is returned by the meter. It is the measure of how much energy has been measured going toward the grid since the meter was installed within the Tariff 1 period. The Tariff 1 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** Rev_kWh_Tariff_1
- ▶ **Display scale**

Tariff 2 upstream energy - E.UPSTREAM_T2 - Field ID 10

Technical Description of *Tariff 2 upstream energy*

Cumulative energy flowing upstream to the grid during the Tariff 2 time period.

- ▶ **Type of field** *Tariff 2 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 2 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 2 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 2 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Reverse kWh Tariff 2. This value is returned by the meter. It is the measure of how much energy has been measured going toward the grid since the meter was installed within the Tariff 2 period. The Tariff 2 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** Rev_kWh_Tariff_2
- ▶ **Display scale**

Tariff 3 upstream energy - E.UPSTREAM_T3 - Field ID 11

Technical Description of *Tariff 3 upstream energy*

Cumulative energy flowing upstream to the grid during the Tariff 3 time period.

- ▶ **Type of field** *Tariff 3 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 3 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 3 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 3 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Reverse kWh Tariff 3. This value is returned by the meter. It is the measure of how much energy has been measured going toward the grid since the meter was installed within the Tariff 3 period. The Tariff 3 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** Rev_kWh_Tariff_3
- ▶ **Display scale**

Tariff 4 upstream energy - E.UPSTREAM_T4 - Field ID 12

Technical Description of *Tariff 4 upstream energy*

Cumulative energy flowing upstream to the grid during the Tariff 4 time period.

- ▶ **Type of field** *Tariff 4 upstream energy* is a Measurement of cumulative energy. (Kind "E")
- ▶ **Input Scale** *Tariff 4 upstream energy* is read from the meter in units of 0.1 KW-hr
- ▶ **Output Scale** *Tariff 4 upstream energy* is displayed and stored in units of 1 KW-hr
- ▶ **Field Range** *Tariff 4 upstream energy* as displayed and stored may have a value of 0 to 9999999.9
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Reverse kWh Tariff 4. This value is returned by the meter. It is the measure of how much energy has been measured going toward the grid since the meter was installed within the Tariff 4 period. The Tariff 4 period is set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** Rev_kWh_Tariff_4
- ▶ **Display scale**

EKM OmniMeter v3 and above voltage field - V.OMNIV3 - Field ID 26925

Technical Description of *EKM OmniMeter v3 and above voltage field*

- ▶ **Type of field** *EKM OmniMeter v3 and above voltage field* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

EKM OmniMeter v3 and above voltage field - V.OMNIV3 - Field ID 26925

Technical Description of *EKM OmniMeter v3 and above voltage field*

- ▶ **Type of field** *EKM OmniMeter v3 and above voltage field* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

EKM OmniMeter v3 and above voltage field - V.OMNIV3 - Field ID 26925

Technical Description of *EKM OmniMeter v3 and above voltage field*

- ▶ **Type of field** *EKM OmniMeter v3 and above voltage field* is a Measurement of RMS voltage. (Kind "V")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

EKM OmniMeter v3 and above current field - I.OMNIV3 - Field ID 26926

Technical Description of *EKM OmniMeter v3 and above current field*

- ▶ **Type of field** *EKM OmniMeter v3 and above current field* is a Measurement of RMS current. (Kind "I")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

EKM OmniMeter v3 and above current field - I.OMNIV3 - Field ID 26926

Technical Description of *EKM OmniMeter v3 and above current field*

- ▶ **Type of field** *EKM OmniMeter v3 and above current field* is a Measurement of RMS current. (Kind "I")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

EKM OmniMeter v3 and above current field - I.OMNIV3 - Field ID 26926

Technical Description of *EKM OmniMeter v3 and above current field*

- ▶ **Type of field** *EKM OmniMeter v3 and above current field* is a Measurement of RMS current. (Kind "I")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 5. [ASCII_NUMBER]

EKM OmniMeter v3 7 character power field - P.OMNIV3 - Field ID 26929

Technical Description of *EKM OmniMeter v3 7 character power field*

- ▶ **Type of field** *EKM OmniMeter v3 7 character power field* is a Measurement of RMS power. (Kind "P")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

EKM OmniMeter v3 7 character power field - P.OMNIV3 - Field ID 26929

Technical Description of *EKM OmniMeter v3 7 character power field*

- ▶ **Type of field** *EKM OmniMeter v3 7 character power field* is a Measurement of RMS power. (Kind "P")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

EKM OmniMeter v3 7 character power field - P.OMNIV3 - Field ID 26929

Technical Description of *EKM OmniMeter v3 7 character power field*

- ▶ **Type of field** *EKM OmniMeter v3 7 character power field* is a Measurement of RMS power. (Kind "P")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

EKM OmniMeter v3 7 character power field - P.OMNIV3 - Field ID 26929

Technical Description of *EKM OmniMeter v3 7 character power field*

- ▶ **Type of field** *EKM OmniMeter v3 7 character power field* is a Measurement of RMS power. (Kind "P")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 7. [ASCII_NUMBER]

EKM OmniMeter v3 and above power factor field - PF.OMNIV3 - Field ID 26931

Technical Description of *EKM OmniMeter v3 and above power factor field*

- ▶ **Type of field** *EKM OmniMeter v3 and above power factor field* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4. [OMNIV3_PF]

EKM OmniMeter v3 and above power factor field - PF.OMNIV3 - Field ID 26931

Technical Description of *EKM OmniMeter v3 and above power factor field*

- ▶ **Type of field** *EKM OmniMeter v3 and above power factor field* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4. [OMNIV3_PF]

EKM OmniMeter v3 and above power factor field - PF.OMNIV3 - Field ID 26931

Technical Description of *EKM OmniMeter v3 and above power factor field*

- ▶ **Type of field** *EKM OmniMeter v3 and above power factor field* is a Measurement of current load power factor. The load power factor is represented by a number between zero and 200. This number indicates an inductive load if less than 100, a capacitive load if greater than 100, and a purely resistive load if equal to 100. (Kind "PF")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A power factor as formatted by the EKM OmniMeter v3 and later meters of length 4. [OMNIV3_PF]

Maximum power - P.MAX - Field ID 26

Technical Description of *Maximum power*

The maximum RMS power flowing in either direction during the time period indicated by CHOICE.DEMAND_PERIOD and as reset at a period indicated by CHOICE.DEMAND_RESET_PERIOD.

- ▶ **Type of field** *Maximum power* is a Measurement of RMS power. (Kind "P")
- ▶ **Input Scale** *Maximum power* is read from the meter in units of 1 W
- ▶ **Output Scale** *Maximum power* is displayed and stored in units of 1 W
- ▶ **Field Range** *Maximum power* as displayed and stored may have a value of 0 to 99999999
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 8. [ASCII_NUMBER]

Display Description

Max Demand is a measure of the highest average peak watts. It is calculated on 15, 30, or 60 minute time periods by the meter. This time period can be set internally in the meter over RS485 using our EKM Dash software and a USB to RS485 converter. The Max Demand is cumulative. It can be reset.

- ▶ **Display field name** RMS_Watts_Max_Demand
- ▶ **Display scale**

Maximum Demand Period - CHOICE.DEMAND_PERIOD - Field ID 27

Technical Description of *Maximum Demand Period*

The time range which the reported maximum demand (P.MAX) was measured within.

- ▶ **Type of field** *Maximum Demand Period* is a A value which must be one of several choices. (Kind "CHOICE")
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Choice ID	Choice Name	Choice Description
0	OFF	The maximum demand (P.MAX) field contains the maximum power usage since
1	MONTHLY	The maximum demand (P.MAX) field contains the maximum power usage over
2	WEEKLY	The maximum demand (P.MAX) field contains the maximum power usage over
3	DAILY	The maximum demand (P.MAX) field contains the maximum power usage over
4	HOURLY	The maximum demand (P.MAX) field contains the maximum power usage over

Protocol Field || Maximum Demand Period - CHOICE.DEMAND_PERIOD - Field ID 27

Display Description

- ▶ Display field name Max_Demand_Period

Pulse input 1 ratio - RATIO.P1 - Field ID 34

Technical Description of *Pulse input 1 ratio*

The ratio of pulses counted by pulse input 1 to the pulse count reported in COUNT.P1 .

- ▶ **Type of field** *Pulse input 1 ratio* is a A ratio between two values of the same dimension (Kind "RATIO")
- ▶ **Input Scale** *Pulse input 1 ratio* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Pulse input 1 ratio* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Pulse input 1 ratio* as displayed and stored may have a value of 0 to
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Pulse_Ratio_1
- ▶ **Display scale**

Pulse input 2 ratio - RATIO.P2 - Field ID 35

Technical Description of *Pulse input 2 ratio*

The ratio of pulses counted by pulse input 2 to the pulse count reported in COUNT.P2 .

- ▶ **Type of field** *Pulse input 2 ratio* is a A ratio between two values of the same dimension (Kind "RATIO")
- ▶ **Input Scale** *Pulse input 2 ratio* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Pulse input 2 ratio* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Pulse input 2 ratio* as displayed and stored may have a value of 0 to
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Pulse_Ratio_2
- ▶ **Display scale**

Pulse input 3 ratio - RATIO.P3 - Field ID 36

Technical Description of *Pulse input 3 ratio*

The ratio of pulses counted by pulse input 3 to the pulse count reported in COUNT.P3 .

- ▶ **Type of field** *Pulse input 3 ratio* is a A ratio between two values of the same dimension (Kind "RATIO")
- ▶ **Input Scale** *Pulse input 3 ratio* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Pulse input 3 ratio* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Pulse input 3 ratio* as displayed and stored may have a value of 0 to
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ **Display field name** Pulse_Ratio_3
- ▶ **Display scale**

Current transformer ratio - RATIO.CT - Field ID 29

Technical Description of *Current transformer ratio*

The ratio of current measured by the current transformer to CT current output in A / 26.6 mA.

- ▶ **Type of field** *Current transformer ratio* is a A ratio between two values of the same dimension (Kind "RATIO")
- ▶ **Input Scale** *Current transformer ratio* is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** *Current transformer ratio* is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** *Current transformer ratio* as displayed and stored may have a value of 0 to
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ Display field name CT_Ratio
- ▶ Display scale

Protocol Field | Maximum demand reset period -
CHOICE.DEMAND_RESET_PERIOD - Field ID 57

**Maximum demand reset period - CHOICE.DEMAND_RESET_PERIOD -
Field ID 57**

Protocol Field II Maximum demand reset period - CHOICE.DEMAND_RESET_PERIOD - Field ID 57

Technical Description of *Maximum demand reset period*

The time period at which the reported maximum demand (P.MAX) is reset.

- ▶ **Type of field** *Maximum demand reset period* is a A value which must be one of several choices. (Kind "CHOICE")
- ▶ **Direction of field** This field is read from the meter and inserted into the output message. [IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 1. [ASCII_NUMBER]

Choice ID	Choice Name	Choice Description
0	OFF	The maximum demand (P.MAX) is never reset.
1	MONTHLY	The maximum demand (P.MAX) is reset monthly.
2	WEEKLY	The maximum demand (P.MAX) is reset weekly.
3	DAILY	The maximum demand (P.MAX) is reset daily.
4	HOURLY	The maximum demand (P.MAX) is reset hourly.

Protocol Field III Maximum demand reset period - CHOICE.DEMAND_RESET_PERIOD - Field ID 57

Display Description

The v4 Meter can be set to Reset the Max Demand register on a schedule. You can make this setting in the meter over RS485 using our EKM Dash software and a USB to RS485 converter.

- ▶ **Display field name** Max_Demand_Rst

- **RATIO.CF** - Field ID 67

Technical Description of

- ▶ **Type of field** is a A ratio between two values of the same dimension (Kind "RATIO")
- ▶ **Input Scale** is read from the meter in units of DIMENSIONLESS
- ▶ **Output Scale** is displayed and stored in units of DIMENSIONLESS
- ▶ **Field Range** as displayed and stored may have a value of 0 to
- ▶ **Direction of field** This field is read from the meter and inserted into the output message.
[IN_THEN_GW]
- ▶ **Format of field** An ASCII-encoded number preceded by leading zeroes of length 4. [ASCII_NUMBER]

Display Description

- ▶ Display field name CF_Ratio
- ▶ Display scale

Protocol Field | **EKM OmniMeter v4 response type B constant 1 -**
CONST.OMNIV4B_RESP_CONST_1 - Field ID 27048

EKM OmniMeter v4 response type B constant 1 - **CONST.OMNIV4B_RESP_CONST_1 - Field ID 27048**

Technical Description of *EKM OmniMeter v4 response type B constant 1*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 1* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 10. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30=30=30=30=30

Protocol Field I **EKM OmniMeter v4 response type B constant 2 -**
CONST.OMNIV4B_RESP_CONST_2 - Field ID 27049

EKM OmniMeter v4 response type B constant 2 - **CONST.OMNIV4B_RESP_CONST_2 - Field ID 27049**

Technical Description of *EKM OmniMeter v4 response type B constant 2*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 2* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 10. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30=30=30=30=30

Protocol Field | **EKM OmniMeter v4 response type B constant 3 -**
CONST.OMNIV4B_RESP_CONST_3 - Field ID 27050

EKM OmniMeter v4 response type B constant 3 - **CONST.OMNIV4B_RESP_CONST_3 - Field ID 27050**

Technical Description of *EKM OmniMeter v4 response type B constant 3*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 3* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 10. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30=30=30=30=30

Protocol Field | **EKM OmniMeter v4 response type B constant 4 -
CONST.OMNIV4B_RESP_CONST_4 - Field ID 27051**

EKM OmniMeter v4 response type B constant 4 - CONST.OMNIV4B_RESP_CONST_4 - Field ID 27051

Technical Description of *EKM OmniMeter v4 response type B constant 4*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 4* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 10. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30=30=30=30=30

Protocol Field | **EKM OmniMeter v4 response type B constant 5 -**
CONST.OMNIV4B_RESP_CONST_5 - Field ID 27052

EKM OmniMeter v4 response type B constant 5 - **CONST.OMNIV4B_RESP_CONST_5 - Field ID 27052**

Technical Description of *EKM OmniMeter v4 response type B constant 5*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 5* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 10. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30=30=30=30=30

Protocol Field | **EKM OmniMeter v4 response type B constant 6 -
CONST.OMNIV4B_RESP_CONST_6** - Field ID 27053

EKM OmniMeter v4 response type B constant 6 - CONST.OMNIV4B_RESP_CONST_6 - Field ID 27053

Technical Description of *EKM OmniMeter v4 response type B constant 6*

- ▶ **Type of field** *EKM OmniMeter v4 response type B constant 6* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 6. [IGNORE]
- ▶ **Constant** =30=30=30=30=30=30

Protocol Field | **EKM OmniMeter v3 and above date and time field - DATETIME.OMNIV3 - Field ID 26937**

EKM OmniMeter v3 and above date and time field - DATETIME.OMNIV3 - Field ID 26937

Technical Description of *EKM OmniMeter v3 and above date and time field*

- ▶ **Type of field** *EKM OmniMeter v3 and above date and time field* is a Date and time measured since 1Jan1970 UTC, without leap seconds (Kind "DATETIME")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A date as formatted by the EKM OmniMeter v3 and later meters of length 14. [OMNIV3_DATE]

Protocol Field | EKM OmniMeter v4 request type B identifier -
CONST.OMNIV4B_REQ_TYPE - Field ID 27054

EKM OmniMeter v4 request type B identifier - CONST.OMNIV4B_REQ_TYPE - Field ID 27054

Technical Description of *EKM OmniMeter v4 request type B identifier*

- ▶ **Type of field** *EKM OmniMeter v4 request type B identifier* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of verbatim words of length 2. [WORDS]
- ▶ **Constant** =30=31

Protocol Field | **EKM OmniMeter v3 and above response postamble -
CONST.OMNIV3_RESP_POSTAMBLE - Field ID 26947**

**EKM OmniMeter v3 and above response postamble -
CONST.OMNIV3_RESP_POSTAMBLE - Field ID 26947**

Technical Description of *EKM OmniMeter v3 and above response postamble*

- ▶ **Type of field** *EKM OmniMeter v3 and above response postamble* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A sequence of input words to ignore of length 4. [IGNORE]
- ▶ **Constant** =21=0d=0a=03

Protocol Field | End of EKM CRC14 calculation and check against computed value or generate computed value - PINSN.EKMCRC14_END - Field ID 26949

End of EKM CRC14 calculation and check against computed value or generate computed value - PINSN.EKMCRC14_END - Field ID 26949

Technical Description of *End of EKM CRC14 calculation and check against computed value or generate computed value*

- ▶ **Type of field** *End of EKM CRC14 calculation and check against computed value or generate computed value* is a Protocol instruction (Kind "PINSN")
- ▶ **Direction of field** This field is read from the meter and not inserted into the output message. [IN]
- ▶ **Format of field** A two word EKM CRC14 value and the EKM CRC14 check of the calculated value against the field as read from the meter of length 2. [EKMCRC14_END]

Protocol Field | **EKM OmniMeter v4 delay before sending close string - INTERVAL.OMNIV4AB_CLOSE_DELAY - Field ID 27081**

EKM OmniMeter v4 delay before sending close string - INTERVAL.OMNIV4AB_CLOSE_DELAY - Field ID 27081

Technical Description of *EKM OmniMeter v4 delay before sending close string*

- ▶ **Type of field** *EKM OmniMeter v4 delay before sending close string* is a An interval of time (Kind "INTERVAL")
- ▶ **Direction of field** This field does not read from or write to the meter. [NONE]
- ▶ **Format of field** A delay of length 200. [DELAY]

Protocol Field | **EKM OmniMeter v3 and above close string -
CONST.OMNIV3_CLOSE_CONST - Field ID 26992**

EKM OmniMeter v3 and above close string - CONST.OMNIV3_CLOSE_CONST - Field ID 26992

Technical Description of *EKM OmniMeter v3 and above close string*

- ▶ **Type of field** *EKM OmniMeter v3 and above close string* is a A constant or a list of constants. (Kind "CONST")
- ▶ **Direction of field** This field is written to the meter. [OUT]
- ▶ **Format of field** A sequence of input words to ignore of length 5. [IGNORE]
- ▶ **Constant** =01=42=30=03=75

JSON Schema

```
{
  "~id": 2,
  "title": "Meter model",
  "description": "The model number as reported by the EKM OmniMeter v3 and above. (Kind \u0022ID\u0022)",
  "type": "string",
  "minLength": 2,
  "maxLength": 2,
  "media": {
    "binaryEncoding": "quoted-printable"
  }
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "WORDS",
  "CHOICE.PINSN_FIELD": "ID.MODEL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "BINARY",
  "CONST.PINSN_LENGTH": 2
}
```

JSON Schema

```
{
  "~id": 1,
  "title": "Meter firmware version",
  "description": "The firmware version as reported by the EKM OmniMeter v3 and above. (Kind \u0022ID\u0022)",
  "type": "string",
  "minLength": 1,
  "maxLength": 1,
  "media": {
    "binaryEncoding": "quoted-printable"
  }
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "WORDS",
  "CHOICE.PINSN_FIELD": "ID.FWVER",
  "CHOICE.PINSN_OUTPUT_FORMAT": "BINARY",
  "CONST.PINSN_LENGTH": 1,
  "CONST.PINSN_CONST": ["=15"]
}
```

JSON Schema

```
{
  "~id": 3,
  "title": "Total energy",
  "description": "Cumulative energy flowing in either direction. (Kind \u0022E\u0022 for \u0022Energy)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.TOTAL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 38,
  "title": "Reactive energy",
  "description": "Cumulative reactive energy measured in either direction through the meter. (Kind \u0026 u",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.REACTIVE_TOTAL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 8,
  "title": "Total upstream energy",
  "description": "Cumulative energy flowing upstream to the grid. (Kind \u002E\u0022 for \u0022Energy",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_TOTAL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 51,
  "title": "Line 1 energy",
  "description": "Cumulative energy flowing in either direction on line 1. (Kind \u0022E\u0022 for \u0022U\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{  
  "~id": 52,  
  "title": "Line 2 energy",  
  "description": "Cumulative energy flowing in either direction on line 2. (Kind \u0022E\u0022 for \u0022U\u0022)",  
  "type": "number",  
  "minimum": 0,  
  "maximum": 9999999.9  
}
```

Protocol instruction

```
{  
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",  
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",  
  "CHOICE.PINSN_FIELD": "E.L2",  
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",  
  "CONST.PINSN_LENGTH": 8,  
  "CHOICE.PINSN_TRANSFORM": "SCALE",  
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",  
  "CONST.PINSN_TRANSFORM_ARG": 1  
}
```

JSON Schema

```
{
  "~id": 53,
  "title": "Line 3 energy",
  "description": "Cumulative energy flowing in either direction on line 3. (Kind \u0022E\u0022 for \u0022U\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 54,
  "title": "Line 1 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid on line 1. (Kind \u0022E\u0022 for \u0022E\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 55,
  "title": "Line 2 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid on line 2. (Kind \u0022E\u0022 for \u0022E\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 56,
  "title": "Line 3 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid on line 3. (Kind \u0022E\u0022 for \u0022E\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 39,
  "title": "Energy since reset",
  "description": "Cumulative energy flowing in either direction since last reset of this field. (Kind",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.SINCE_RESET",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 40,
  "title": "Upstream energy since reset",
  "description": "Cumulative energy flowing upstream to the grid since last reset of this field. (Kind",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_SINCE_RESET",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 13,
  "title": "Line 1 voltage",
  "description": "The RMS voltage across line 1 and neutral. (Kind \u0022V\u0022 for \u0022Voltage\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "V.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 14,
  "title": "Line 2 voltage",
  "description": "The RMS voltage across line 2 and neutral. (Kind \u0022V\u0022 for \u0022Voltage\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "V.L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 15,
  "title": "Line 3 voltage",
  "description": "The RMS voltage across line 3 and neutral. (Kind \u0022V\u0022 for \u0022Voltage\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "V.L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 16,
  "title": "Line 1 current",
  "description": "The RMS current flowing in either direction through line 1. (Kind \u0022I\u0022 for
  "type": "number",
  "minimum": 0,
  "maximum": 9999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "I.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 5,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 17,
  "title": "Line 2 current",
  "description": "The RMS current flowing in either direction through line 2. (Kind \u0022I\u0022 for",
  "type": "number",
  "minimum": 0,
  "maximum": 9999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "I.L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 5,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 18,
  "title": "Line 3 current",
  "description": "The RMS current flowing in either direction through line 3. (Kind \u0022I\u0022 for",
  "type": "number",
  "minimum": 0,
  "maximum": 9999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "I.L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 5,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 19,
  "title": "Line 1 power",
  "description": "The RMS power flowing in either direction through line 1. (Kind \u0022P\u0022 for \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 20,
  "title": "Line 2 power",
  "description": "The RMS power flowing in either direction through line 2. (Kind \u0022P\u0022 for \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 21,
  "title": "Line 3 power",
  "description": "The RMS power flowing in either direction through line 3. (Kind \u0022P\u0022 for \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 22,
  "title": "Total power",
  "description": "The RMS power flowing in either direction. (Kind \u0022P\u0022 for \u0022Instantaneous Power\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.TOTAL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 23,
  "title": "Line 1 power factor",
  "description": "The current load power factor measurement for line 1. (Kind \u0022PF\u0022 for \u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 200
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "OMNIV3_PF",
  "CHOICE.PINSN_FIELD": "PF.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 24,
  "title": "Line 2 power factor",
  "description": "The current load power factor measurement for line 2. (Kind \u0022PF\u0022 for \u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 200
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "OMNIV3_PF",
  "CHOICE.PINSN_FIELD": "PF.L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 25,
  "title": "Line 3 power factor",
  "description": "The current load power factor measurement for line 3. (Kind \u0022PF\u0022 for \u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 200
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "OMNIV3_PF",
  "CHOICE.PINSN_FIELD": "PF.L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 41,
  "title": "Line 1 reactive power",
  "description": "The RMS reactive power flowing in either direction through line 1. (Kind \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.REACTIVE_L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 42,
  "title": "Line 2 reactive power",
  "description": "The RMS reactive power flowing in either direction through line 2. (Kind \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.REACTIVE_L2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 43,
  "title": "Line 3 reactive power",
  "description": "The RMS reactive power flowing in either direction through line 3. (Kind \u0022P\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.REACTIVE_L3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 44,
  "title": "Total reactive power",
  "description": "The RMS reactive power currently flowing in either direction. (Kind \u0022P\u0022 f",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.REACTIVE_TOTAL",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 7
}
```

JSON Schema

```
{
  "~id": 47,
  "title": "Line 1 frequency",
  "description": "The frequency of AC power flowing through line 1. (Kind \u0022FREQ\u0022 for \u0022FREQ.L1\u0022)",
  "type": "number",
  "minimum": 0,
  "maximum": 999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "FREQ.L1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 31,
  "title": "Pulse input 1 pulse count",
  "description": "Cumulative pulse count for pulse input 1. The reported pulse count increments once p
  "type": "number",
  "minimum": 0,
  "maximum": 99999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COUNT.P1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8
}
```

JSON Schema

```
{
  "~id": 32,
  "title": "Pulse input 2 pulse count",
  "description": "Cumulative pulse count for pulse input 2. The reported pulse count increments once p
  "type": "number",
  "minimum": 0,
  "maximum": 99999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COUNT.P2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8
}
```

JSON Schema

```
{
  "~id": 33,
  "title": "Pulse input 3 pulse count",
  "description": "Cumulative pulse count for pulse input 3. The reported pulse count increments once p
  "type": "number",
  "minimum": 0,
  "maximum": 99999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COUNT.P3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8
}
```

JSON Schema

```
{
  "~id": 64,
  "title": "Pulse input 1 state",
  "description": "The current state of pulse input 1. (Kind \u0022CHOICE\u0022 for \u0022Single choice)",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "Pulse input 1 low (open).",
  }, {
    "enum": ["ON"],
    "~id": 1,
    "title": "Pulse input 1 high (closed).",
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_INPUT_STATE",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 65,
  "title": "Pulse input 2 state",
  "description": "The current state of pulse input 2. (Kind \u0022CHOICE\u0022 for \u0022Single choice)",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "Pulse input 2 low (open).",
  }, {
    "enum": ["ON"],
    "~id": 1,
    "title": "Pulse input 2 high (closed).",
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_INPUT_STATE",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 66,
  "title": "Pulse input 3 state",
  "description": "The current state of pulse input 3. (Kind \u0022CHOICE\u0022 for \u0022Single choice)",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "Pulse input 3 low (open).",
  }, {
    "enum": ["ON"],
    "~id": 1,
    "title": "Pulse input 3 high (closed).",
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_INPUT_STATE",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 61,
  "title": "Line 1 power flow direction",
  "description": "The direction of power flowing through line 1. (Kind \u0022CHOICE\u0022 for \u0022S",
  "type": "string",
  "oneOf": [{
    "enum": ["UPSTREAM"],
    "~id": 0,
    "title": "Power through L1 is currently flowing upstream towards the grid."
  }, {
    "enum": ["DOWNSTREAM"],
    "~id": 1,
    "title": "Power through L1 is currently flowing downstream from the grid."
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_POWER_DIRECTION",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 62,
  "title": "Line 2 power flow direction",
  "description": "The direction of power flowing through line 2. (Kind \u0022CHOICE\u0022 for \u0022S",
  "type": "string",
  "oneOf": [{
    "enum": ["UPSTREAM"],
    "~id": 0,
    "title": "Power through L2 is currently flowing upstream towards the grid."
  }, {
    "enum": ["DOWNSTREAM"],
    "~id": 1,
    "title": "Power through L2 is currently flowing downstream from the grid."
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_POWER_DIRECTION",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 63,
  "title": "Line 3 power flow direction",
  "description": "The direction of power flowing through line 3. (Kind \u0022CHOICE\u0022 for \u0022S",
  "type": "string",
  "oneOf": [{
    "enum": ["UPSTREAM"],
    "~id": 0,
    "title": "Power through L3 is currently flowing upstream towards the grid."
  }, {
    "enum": ["DOWNSTREAM"],
    "~id": 1,
    "title": "Power through L3 is currently flowing downstream from the grid."
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_POWER_DIRECTION",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 59,
  "title": "Output 1 state",
  "description": "The state of output 1. (Kind \u0022CHOICE\u0022 for \u0022Single choice\u0022: A va",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "Output 1 off."
  }, {
    "enum": ["ON"],
    "~id": 1,
    "title": "Output 1 on."
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_OUTPUT_STATE",
  "CONST.PINSN_LENGTH": 1,
}
```

JSON Schema

```
{
  "~id": 60,
  "title": "Output 2 state",
  "description": "The state of output 2. (Kind \u0022CHOICE\u0022 for \u0022Single choice\u0022: A va",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "Output 2 off."
  }, {
    "enum": ["ON"],
    "~id": 1,
    "title": "Output 2 on."
  }]
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW_COMPOUND",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "COMPOUND.OMNIV4A_OUTPUT_STATE",
  "CONST.PINSN_LENGTH": 1,
}
```

Schema E.T1

JSON Schema

```
{  
  "~id": 4,  
  "title": "Tariff 1 energy",  
  "description": "Cumulative energy flowing in either direction during the Tariff 1 time period. (Kind",  
  "type": "number",  
  "minimum": 0,  
  "maximum": 9999999.9  
}
```

Protocol instruction

```
{  
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",  
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",  
  "CHOICE.PINSN_FIELD": "E.T1",  
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",  
  "CONST.PINSN_LENGTH": 8,  
  "CHOICE.PINSN_TRANSFORM": "SCALE",  
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",  
  "CONST.PINSN_TRANSFORM_ARG": 1  
}
```

JSON Schema

```
{  
  "~id": 5,  
  "title": "Tariff 2 energy",  
  "description": "Cumulative energy flowing in either direction during the Tariff 2 time period. (Kind",  
  "type": "number",  
  "minimum": 0,  
  "maximum": 9999999.9  
}
```

Protocol instruction

```
{  
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",  
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",  
  "CHOICE.PINSN_FIELD": "E.T2",  
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",  
  "CONST.PINSN_LENGTH": 8,  
  "CHOICE.PINSN_TRANSFORM": "SCALE",  
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",  
  "CONST.PINSN_TRANSFORM_ARG": 1  
}
```

JSON Schema

```
{
  "~id": 6,
  "title": "Tariff 3 energy",
  "description": "Cumulative energy flowing in either direction during the Tariff 3 time period. (Kind",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.T3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 7,
  "title": "Tariff 4 energy",
  "description": "Cumulative energy flowing in either direction during the Tariff 4 time period. (Kind",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.T4",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 9,
  "title": "Tariff 1 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid during the Tariff 1 time period. (Kin",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_T1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 10,
  "title": "Tariff 2 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid during the Tariff 2 time period. (Kin",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_T2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 11,
  "title": "Tariff 3 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid during the Tariff 3 time period. (Kin",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_T3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 12,
  "title": "Tariff 4 upstream energy",
  "description": "Cumulative energy flowing upstream to the grid during the Tariff 4 time period. (Kin",
  "type": "number",
  "minimum": 0,
  "maximum": 9999999.9
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "E.UPSTREAM_T4",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8,
  "CHOICE.PINSN_TRANSFORM": "SCALE",
  "CHOICE.PINSN_TRANSFORM_FIELD": "VAR.OMNIV4A_E_SCALE",
  "CONST.PINSN_TRANSFORM_ARG": 1
}
```

JSON Schema

```
{
  "~id": 26,
  "title": "Maximum power",
  "description": "The maximum RMS power flowing in either direction during the time period indicated b",
  "type": "number",
  "minimum": 0,
  "maximum": 99999999
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "P.MAX",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 8
}
```

JSON Schema

```
{
  "~id": 27,
  "title": "Maximum Demand Period",
  "description": "The time range which the reported maximum demand (\u005chyperlink{P.MAX}{P.MAX}) wa",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) field contains the maximum power usage",
  }, {
    "enum": ["MONTHLY"],
    "~id": 1,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) field contains the maximum power usage",
  }, {
    "enum": ["WEEKLY"],
    "~id": 2,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) field contains the maximum power usage",
  }, {
    "enum": ["DAILY"],
    "~id": 3,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) field contains the maximum power usage",
  }, {
    "enum": ["HOURLY"],
```

JSON Schema

```
{
  "~id": 34,
  "title": "Pulse input 1 ratio",
  "description": "The ratio of pulses counted by pulse input 1 to the pulse count reported in \u005chyp",
  "type": "number",
  "minimum": 0,
  "maximum": ""
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "RATIO.P1",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 35,
  "title": "Pulse input 2 ratio",
  "description": "The ratio of pulses counted by pulse input 2 to the pulse count reported in \u005chyp",
  "type": "number",
  "minimum": 0,
  "maximum": ""
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "RATIO.P2",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 36,
  "title": "Pulse input 3 ratio",
  "description": "The ratio of pulses counted by pulse input 3 to the pulse count reported in \u005chyp",
  "type": "number",
  "minimum": 0,
  "maximum": ""
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "RATIO.P3",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 29,
  "title": "Current transformer ratio",
  "description": "The ratio of current measured by the current transformer to CT current output in A /",
  "type": "number",
  "minimum": 0,
  "maximum": ""
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "RATIO.CT",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```

JSON Schema

```
{
  "~id": 57,
  "title": "Maximum demand reset period",
  "description": "The time period at which the reported maximum demand (\u005chyperlink{P.MAX}{P.MAX})",
  "type": "string",
  "oneOf": [{
    "enum": ["OFF"],
    "~id": 0,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) is never reset."
  }, {
    "enum": ["MONTHLY"],
    "~id": 1,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) is reset monthly."
  }, {
    "enum": ["WEEKLY"],
    "~id": 2,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) is reset weekly."
  }, {
    "enum": ["DAILY"],
    "~id": 3,
    "title": "The maximum demand (\u005chyperlink{P.MAX}{P.MAX}) is reset daily."
  }, {
    "enum": ["HOURLY"],
```

JSON Schema

```
{
  "~id": 67,
  "title": "",
  "description": " (Kind \u0022RATIO\u0022 for \u0022Ratio\u0022: A ratio between two values of the sa
  "type": "number",
  "minimum": 0,
  "maximum": ""
}
```

Protocol instruction

```
{
  "CHOICE.PINSN_DIRECTION": "IN_THEN_GW",
  "CHOICE.PINSN_FORMAT": "ASCII_NUMBER",
  "CHOICE.PINSN_FIELD": "RATIO.CF",
  "CHOICE.PINSN_OUTPUT_FORMAT": "DECIMAL",
  "CONST.PINSN_LENGTH": 4
}
```