FUEL FLOW METERS
DFM Marine 485

Modbus RTU protocol register map

Version 1.0

TECHNOTON
ADVANCED MACHINERY TELEMATICS

The RS-485 digital interface of DFM Marine 485 fuel flow meters complies with the international EIA/TIA-485 standard and supports data transmission via the Modbus RTU protocol in "request–response" mode.

For identification within a common RS-485 network, the DFM Marine 485 must be assigned a unique network address in the range 0…255 (default — 111).

The data transmission rate for DFM Marine 485 can be selected from the following values: 2400, 4800, 9600, 19200, 38400, 57600, or 115200 bit/s (default — 9600 bit/s).

To configure the fuel flow meter's connection parameters with an external device, the flow meter must be connected to a PC via the S6 SK service adapter. Configuration is performed using the Service DFM Marine software via the K-Line interface (ISO 14230). For details on the S6 SK, see the User Manual of the CAN j1939/S6 Telematics Interface.

IMPORTANT: A **mandatory condition** for correct data transmission by DFM Marine 485 fuel flow meters is the presence of two **120 Ohm** termination resistors installed at both ends of the RS-485 communication line between wires 485A and 485B.

The Modbus RTU data transmission protocol is based on a Master–Slave architecture. The register map of DFM Marine 485 output messages available via the Modbus RTU protocol is shown in table 1.

To read data from the register map, the standard Modbus RTU protocol function **3 (0x03) Read Holding Registers** is used

The fuel flow meter's output messages via the Modbus RTU protocol contain:

- A unique device network address (Slave ID) in the range 0…255 (default — 111).
- Function code (FCode = 3).
- Data.
- Checksum (CRC).

The data in DFM Marine 485 output messages is represented as an unsigned integer — **unsigned int**.
The data size in each register is **2 bytes**.
The request execution timeout is **1000 ms**.

To read a register, the external Master device (for example, a telematics terminal) sends a request to the DFM Marine 485 (Slave device). The request contains function code 3 (Read Holding Registers), the address of the requested register (Reg Addr), and the number of registers to be read (Reg Count).

In response, the DFM Marine 485 sends back a data packet that includes its network address (Slave ID), protocol function code (FCode = 3), the number of bytes in the data field (Bytes Count), and the data field (DATA) containing the value of the requested register. To read multiple consecutive registers, the request must specify the address of the first register and the total number of registers to be read (see figure 1).

Using the Modbus RTU protocol, it is possible to **reset the counters** "Total fuel consumption. Resettable" (SPN 521314/28.0) and "Operating time of the flow meter. Resettable" (SPN 521171/28.0), which are stored in the internal memory of the DFM Marine 485. To reset these counters, a value of **1** must be written to register **200**. After this, the register will issue a command (SPN 521167) to clear the counters.

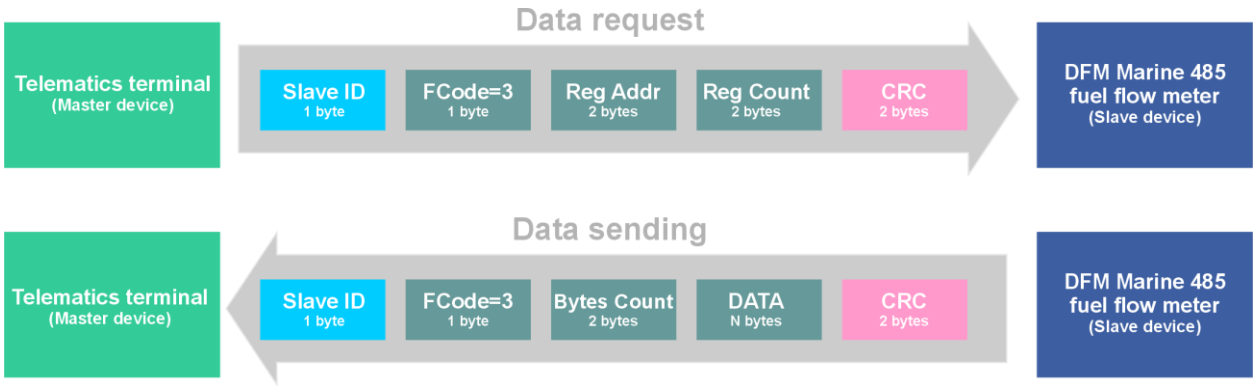

Figure 1 — Data exchange diagram via the Modbus RTU protocol between the DFM Marine 485 fuel flow meter and the telematics terminal

Table 1 — Modbus RTU register map of DFM Marine 485 fuel flow meters

Address	Content	SPN*	Specifier	Access type
0	Engine Fuel Rate	521795		read
1	Engine Total Hours of Operation (high word)	247		read
2	Engine Total Hours of Operation (low word)	247		read
3	Engine Fuel Temperature 1	174		read
4	Engine Fuel Rate (high word)	521313		read
5	Engine Fuel Rate (low word)	521313		read
6	Engine Mode by Fuel Rate	521181		read
7	Total Fuel Used (high word)	521314		read
8	Total Fuel Used (low word)	521314		read
9	Total Fuel Used (high word)	521314	9.0 Idle	read
10	Total Fuel Used (low word)	521314	9.0 Idle	read
11	Total Fuel Used (high word)	521314	9.1 Optimal	read
12	Total Fuel Used (low word)	521314	9.1 Optimal	read
13	Total Fuel Used (high word)	521314	9.2 Overload	read
14	Total Fuel Used (low word)	521314	9.2 Overload	read
15	Total Fuel Used (high word)	521314	9.3 Cheating	read
16	Total Fuel Used (low word)	521314	9.3 Cheating	read
17	Total Fuel Used (high word)	521314	28.0 Clearable	read
18	Total Fuel Used (low word)	521314	28.0 Clearable	read
19	Flowmeter Hours Of Operation (high word)	521171		read
20	Flowmeter Hours Of Operation (low word)	521171		read
21	Flowmeter Hours Of Operation (high word)	521171	9.0 Idle	read
22	Flowmeter Hours Of Operation (low word)	521171	9.0 Idle	read
23	Flowmeter Hours Of Operation (high word)	521171	9.1 Optimal	read
24	Flowmeter Hours Of Operation (low word)	521171	9.1 Optimal	read
25	Flowmeter Hours Of Operation (high word)	521171	9.2 Overload	read
26	Flowmeter Hours Of Operation (low word)	521171	9.2 Overload	read

Address	Content	SPN*	Specifier	Access type
27	Flowmeter Hours Of Operation (high word)	521171	9.3 Cheating	read
28	Flowmeter Hours Of Operation (low word)	521171	9.3 Cheating	read
29	Flowmeter Hours Of Operation (high word)	521171	28.0 Clearable	read
30	Flowmeter Hours Of Operation (low word)	521171	28.0 Clearable	read
31	Flowmeter Hours Of Operation (high word)	521171	9.5 Interference	read
32	Flowmeter Hours Of Operation (low word)	521171	9.5 Interference	read
33	Global Fuel Used (high word)	521674		read
34	Global Fuel Used (low word)	521674		read
35	Engine Fuel Rate (high word)	521313	2.1 Mean	read
36	Engine Fuel Rate (low word)	521313	2.1 Mean	read
37	Keyswitch Battery Potential	158		read
38	Total CO2 Emission (high word)	521864		read
39	Total CO2 Emission (low word)	521864		read
40	Hourly CO2 Emission (high word)	521865		read
41	Hourly CO2 Emission (low word)	521865		read
200**	Command Code	521167		write

* For detailed information on SPN, see the S6 Database at https://s6.jv-technoton.com/en.
** Register 200 is used only for sending commands.
Supported command: 1 — reset of resettable counters in registers 17, 18, 29, 30.

Example 1: Read the value of SPN 521795 "Engine Fuel Rate, l/h" from the register

Reg Addr	Data	
	1 byte (high byte)	1 byte (low byte)
0	04	85

The Request structure: 0x6F 0x03 0x00 0x00 0x00 0x01 0x8C 0x84, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the requested flow meter: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
00 00 – address of the requested register: Reg Addr=0;
00 01 – number of registers to be read: Reg Count=1;
8C 84 – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

The Response structure: 0x6F 0x03 0x02 0x04 0x85 0x92 0xEE, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the flow meter from which data are read: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
02 – number of bytes in the data field: Bytes Count=2;
04 85 – data field of register 0 (2 bytes): Data=0485;
92 EE – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

Conversion of data for verification: 485 (Hex)=1157 (Dec) · 1 + 0 = 1157 l/h,
where 1 l/h – factor (resolution); 0 l/h – offset for calculation of values of SPN 521795.

Example 2: Read the value of SPN 174 "Engine Fuel Temperature 1, °C" from the register

Reg Addr	Data	
	1 byte (high byte)	1 byte (low byte)
3	00	44

The Request structure: 0x6F 0x03 0x00 0x03 0x00 0x01 0x7C 0x84, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the requested flow meter: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
00 03 – address of the requested register: Reg Addr=3;
00 01 – number of registers to be read: Reg Count=1;
7C 84 – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

The Response structure: 0x6F 0x03 0x02 0x00 0x44 0x51 0xBE, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the flow meter from which data are read: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
02 – number of bytes in the data field: Bytes Count=2;
00 44 – data field of register 3 (2 bytes): Data=0044;
51 BE – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

Conversion of data for verification: 44 (Hex)=68 (Dec) · 1 - 40 = 28 °C,
where 1 °C – factor (resolution); -40 °C – offset for calculation of values of SPN 174.

Example 3: Read the value of SPN 521314 "Total Fuel Used, m^3" from the register

Reg Addr	Data		Note
	1 byte (high byte)	1 byte (low byte)	
7	00	01	High word
8	13	28	Low word

The Request structure: 0x6F 0x03 0x00 0x07 0x00 0x02 0x8D 0x44, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the requested flow meter: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
00 07 – address of the first from the requested registers: Reg Addr=7;
00 02 – number of registers to be read: Reg Count=2;
8D 44 – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

The Response structure: 0x6F 0x03 0x04 0x00 0x01 0x13 0x28 0x29 0x1B, where
0x – prefixes of the hexadecimal system of numeration;
6F – network address of the flow meter from which data are read: Slave ID=111;
03 – number of the function of reading registers: FCode=3;
04 – number of bytes in the data field: Bytes Count=4;
00 01 – data field of register 7 (2 bytes): Data=0001;
13 28 – data field of register 8 (2 bytes): Data=1328;
29 1B – field of checksum (CRC) calculated according to crc16 algorithm (2 bytes).

Conversion of data for verification: 11328 (Hex)=70440 (Dec) · 0.00001 + 0 = 0.7044 m^3,
where 0.00001 m^3 – factor (resolution); 0 m^3 – offset for calculation of values of SPN 521314.