DTU sensor data formats.

1. DTU sensor supports four different software protocols over RS-485 – LSit, Omnicomm-2, Omnicomm-3, Modbus RTU. At one time the DTU runs only one protocol. The selection of desired protocol and its options can be done through DTU configurator.
2. Protocol LSit
    a. LSit is a very simple binary protocol. Each device in RS-485 network has its own unique net address from 1 till 247. Each device answers for any command that contains device's net address or address zero (broadcast address). LSit protocol runs on fixed 2400 baud rate, no parity, 8 data bits, 1 stop bit.
    b. Each command from master (the device that wants to get data from DTU) consists of N bytes: B1, B2, … B_n. B_{n-1} and B_n are CRC bytes (B_{n-1} is the lowest CRC byte, B_n is the highest CRC byte). B1 is net address byte, B2 is command byte. The CRC calculation algorithm is represented as C code in Addendum A.
    c. Command 'Get Data'. N=6, B1=net address, B2=1, B3=0, B4=0, B5 CRC(low byte), B6 CRC (high byte). DTU's answer has 15 bytes: B1…B15:
        i. B1=LSit net address
        ii. B2 fuel temperature (in direct format)
        iii. B3 low byte of fuel level, B4 – high byte of fuel level. The value 1 corresponds to level 0.1 mm. For compatibility the most significant bit 16 is one.
        iv. B5 low byte of fuel density, B6 high byte of fuel density in kg/m^3
        v. B7, B8 reserved
        vi. B9 - DTU software version
        vii. B10 – fuel type. 0 – diesel universal, 1 – diesel summer, 2 – diesel winter, 3 – diesel arctic, 4 – kerosene RT, 5 – kerosene TC, 6 – petrol AI-80, 7 – petrol AI-92, 8 – petrol AI-95
        viii. B11 accelerometer data update counter, increments one time at four seconds
        ix. B12 low byte of top cap
        x. B13 high byte of top cap. The value one corresponds to 0.1 mm.
        xi. B14 low CRC byte, B15 high CRC byte.
    d. Command 'Set settings' (used to set new net address and fuel type). N=64, B1=net address, B2=101 (decimal), B3…B34 – array with settings with its own CRC, B63 CRC (low byte), B64 crc (high byte). New net address will be used after restarting the DTU. DTU's answer has three bytes.
        i. Command 'Set setting' format: B3=0x77, B4= new net address, B5=fuel type, B6…B22 – reserved, B33…B34 -> CRC
        ii. Answer from DTU: three bytes: B1 -> net address, B2 low CRC byte, B3 high CRC byte.
    e. Command 'Get settings' (used to read net address and fuel type). N=6, B1=net address (it can be zero if only one DTU connected), B2=102 (decimal), B3=0, B4=0, B5 = low CRC, B6 = high CRC.
        i. Answer from DTU: N=41, B1…B32 array with settings, B40,B41 – CRC. Settings array format: B1=0x77, B2=net address written in flash, B3=fuel type, B31..B32=CRC.
3. Protocol Omnicomm-2
    a. Omnicomm-2 protocol is simple binary protocol with one command supported (command 6 – 'get data'). RS-485 has following setting: 19200 baud rate, no parity, 8 bit data, 1 stop bit.
    b. Command 6 format: length is 4 bytes. B1=0x31, B2=net address, B3=6, B4=CRC (8-bit CRC). The CRC algorithm is represented in Addendum B.

   c. Command 6 answer: B1=0x3E, B2=net address, B3=6, B4, B5, B6, B7, B8, B9=CRC (8-bit). Omnicomm-2 protocol is designed for special GSM/GPS trackers that supports command 6, but send over GSM only three bytes: B4, B5, B6 (normally B4 is one byte signed char temperature, B5,B6 = level; B7,B8 f_curr). Because DTU needs to send not three, but six bytes, one DTU answers to two network addresses: net address & net address+1. So when using protocol Omnicomm-2 all DTU net addresses must differ more than to 1 (for example, 1,3,5,7 and so on).
   d. Answer format for network address: net_address. B4 is signed char temperature in Celsius. B5,B6 is unsigned short int fuel level, the value 1 corresponds to 0.1 mm.
   e. Answer format for network address: net_address + 1. B4 is fuel type (0 – diesel universal, 1 – diesel summer, 2 – diesel winter, 3 – diesel arctic, 4 – kerosene RT, 5 – kerosene TC, 6 – petrol AI-80, 7 – petrol AI-92, 8 – petrol AI-95). B5,B6 is unsigned short int fuel density, the value 1 corresponds to 0.1 kg/m^3.
4. Protocol Omnicomm-3
   a. Omnicomm-3 protocol is simple binary protocol with one command supported (command 6 – 'get data'). RS-485 has following setting: 19200 baud rate, no parity, 8 bit data, 1 stop bit.
   b. Command 6 format: length is 4 bytes. B1=0x31, B2=net address, B3=6, B4=CRC (8-bit CRC). The CRC algorithm is represented in Addendum B.
   c. Command 6 answer: B1=0x3E, B2=net address, B3=6, B4, B5, B6, B7, B8, B9=CRC (8-bit). Omnicomm-3 protocol is designed for special GSM/GPS trackers that supports command 6, but send over GSM only two bytes: B5, B6. Because DTU needs to send not two, but six bytes, one DTU answers to three network addresses: net address, net address+1 & net address+2. So when using protocol Omnicomm-3 all DTU net addresses must differ more than to 2 (for example, 1,4,7,10 and so on).
   d. Answer format for network address: net_address. B5,B6 is unsigned short int fuel level, the value 1 corresponds to 0.1 mm.
   e. Answer format for network address: net_address + 1. B5,B6 is unsigned short int fuel density, the value 1 corresponds to 0.1 kg/m^3.
   f. Answer format for network address: net_address + 2. B5,B6 is signed short int fuel temperature, the value 1 corresponds to (1/128) ^{O}C.
5. Protocol Modbus
   a. DTU supports Modbus RTU protocol. Default parameters are: net address 1, baud rate 19200, parity – even, data bits – 8, stop bit -1. Using DTU configurator user can change: baud rate in range 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400; net address in range 1…247, parity: even, odd, none. The DTU's Modbus registers map is represented in Addendum C.

Addendum A. CRC algorithm for LSit protocol.

Calculation of 16-bit CRC for array b with size of N character.

```
unsigned short int CRC_Calc16(unsigned char *b, unsigned int N)
{
unsigned short int crc;
unsigned int i;
unsigned short int j;

for(i=crc=0; i<N; i++)
 {
  j=b[i];
  crc = crc ^ (j << 8);
  for(j=0;j<8;j++)
   {
   if(crc & 0x8000) crc = (crc<<1) ^ 0x1021;
     else crc <<= 1;
   }
 }

return crc;
}
```

Addendum B. 8-bit CRC in Omnicomm-2 & Omnicomm-3 protocols.

```
unsigned char calc_crc8( unsigned char *mas, unsigned char Len )
{
  unsigned char i,dat,crc,fb,st_byt;
  st_byt=0; crc=0;
  do{
    dat=mas[st_byt];
    for( i=0; i<8; i++) {
      fb = crc ^ dat;
      fb &= 1;
      crc >>= 1;
      dat >>= 1;
      if( fb == 1 ) crc ^= 0x8c;
    }
    st_byt++;
  } while( st_byt < Len );
  return crc;
}
```

Addendum C. DTU's Modbus registers map.

Modbus registers	Value Type	Access	Description
1-16	STRING(16)	R/O	DTU serial number in ASCII format (one character per one register)
17	UNSIGNED16	R/O	hardware version (in format byte1.byte0)
18	UNSIGNED16	R/O	software version (in format byte1.byte0)
19	UNSIGNED16	R/O	top level in mm (option)
20	UNSIGNED8	R/O	DTU type (0 -> normal DTU)
1000	UNSIGNED16	R/O	fuel level in tenth of mm, 1 corresponds to 0.1 mm
1001	UNSIGNED16	R/O	fuel dencity in tenth of kg/m^3, 1 corresponds to 0.1 kg/m^3
1002	SIGNED16	R/O	fuel temperature in ^oC
1003	UNSIGNED8	R/O	fuel type index (0 -> diesel fuel, all types; 1-> diesel summer fuel; 2-> diesel winter fuel; 3 -> diesel arctic fuel; 4 -> kerosene RT; 5 -> kerosene TS; 6 -> petrol AI-80; 7 -> petrol AI-92; 8 -> petrol AI-95)
3000	UNSIGNED8	R/O	modbus slave DTU address (from 1 to 247)
3001	UNSIGNED8	R/O	parity settings (0 -> none, 1 -> odd, 2 -> even)
3002, 3003	UNSIGNED32	R/O	Modbus DTU baud rate (bits per second)
5000	UNSIGNED8	R/O	calibration and settings flash status (0 -> good)