International Electrotechnical Commission, Electricity metering–Data exchange for meter reading, tariff and load control

This document specifies a subset of the IEC 62056-21 communications protocol used to transfer data between a master computer station and the meter. The document provides the complete information necessary to develop third-party communications software capable of communication with the meter. The IEC 62056-21 protocol is supported via the optical IR port COM1 (it is the default mode for COM1) and serial port COM3. The initial baud rate for data exchange is programmable for IEC 62056-21 ports. For the optical port, it can be selected from 300 Bd to 19200 Bd. For port COM3, it can be set to one of the standard baud rates from 300 Bd to 19200 Bd, and to not-standard baud rates 38400 Bd (Z=7), 57600 Bd (Z=8) or 115200 Bd (Z=9). The default factory set baud rate is 19200 Bd (Z=6) for all serial ports.

According to IEC 62056-21 protocol there are two modes to read data as follows...

Data readout mode

Programming mode

We are using Data readout mode. The meter provides a standard data readout of ASCII-coded data with optional baud rate switching. A data readout includes energy, maximum demand and cumulative maximum demand readings for all configured total and tariff registers for the present and three previous billing periods. Data readings are provided with the OBIS data identification code (see below) and unit information.

Data transmission protocol:

Request message: /?deviceaddress!CRLF

Identification message: /XXXZidentificationCRLF

Acknowledgement message: ACK

Repeat request message: NAK

Explanations of message contents:

1. Start character "/" (forward oblique).
2. Transmission request command "?" (question mark)
3. End character "!" (exclamation mark).
4. Completion character (CR, carriage return, LF, line feed).
5. Acknowledge character (ACK, acknowledge).
6. Negative acknowledgement (NAK)
7. Flag ID from meter manufacturer (XXX)
8. Request baud rate by meter (Z)

After ACK response device gives each and every data present in it in the form of OBIS codes. Further we have to decode them and get desired output information.