

## Modbus Protocol

The Modbus protocol is a messaging structure widely used to establish communication that is independent of the underlying physical layer and is traditionally implemented using RS232, RS422 or RS485.

The Modbus method utilizes a Master-Slave technique in which only one device (the Master) can initiate transactions (queries). The other devices (the slaves) respond by supplying the requested data to the master or by taking the action requested in the query.

The Master can address individual Slaves or can initiate a broadcast message to all Slaves. However, Slaves respond only to queries that are addressed to them individually.

The Modbus protocol establishes the format for the Master's query by placing into it the device address, a function code defining the requested action, any data to be sent, and an error checking field.

The Slave's response message is also constructed using the Modbus protocol. It contains fields confirming the action taken, any data to be returned, and an error checking field. In the event of an error in receipt of the message, or if the Slave is unable to perform the requested action, the Slave will construct an error message and send it as a response.

### Modbus network standard transmission modes:

**ASCII** American Standard Code for Information Interchange. This is an eight bit byte in a message sent as two ASCII characters.

**RTU** Remote Terminal Unit. This is an eight bit byte in a message which contains two four bit hexadecimal characters.

### Definitions:

**Query** The function code in the query informs the addressed Slave device of the action to be performed.

**Response** The function code in the response is an echo of the function code in the query, and any data collected by the Slave.

**Message Framing** In either ASCII or RTU formats a Modbus message is placed by the transmitting device into a frame that has a known beginning and end point. This enables receiving devices to begin at the start of a message, read the address portion and determine which device is addressed, and to know when the message is completed.

**Function Field** The function code field of a message frame contains two characters (ASCII) or eight bits (RTU). Valid codes are in the range of 1...255 decimal.

**Data Field** The data field is constructed using two sets of two hexadecimal digits in the range of 00 to FF. These can be made from a pair of ASCII characters or from one RTU character.

**Error Checking** There are two kinds of error checking methods used for a Field standard Modbus network. The error checking field contents depend on the method used:

**ASCII** In this mode the error checking field contains two characters. This format utilizes a Longitudal Redundancy Check (LRC) calculation.

**RTU** In this mode the error checking field contains a 16 bit value implemented as two eight bit bytes. This format utilizes a Cyclical Redundancy Check (CRC) calculation.



**Finesse Solutions, LLC**  
3350 Scott Blvd, Bldg 1  
Santa Clara, CA 95054

**Finesse Solutions AG**  
Via Sogn Gieri 27a  
CH-7402 Bonaduz  
Switzerland

**877-204-8644**  
**www.finesse.com**