

DASTEC Corporation  
457A Carlisle Drive  
Herndon, VA 20170 USA  
Tel: 1-703-709-0515  
Fax: 1-703-709-0985  
www.dastec.com

## **Modbus Serial Communication Interface API**

The DASTEC Corporation **Modbus Serial Communication Interface API** allows the user to implement bi-directional serial communications to exchange data between applications running on a Windows/WinCE-based system with other devices supporting the Modbus RTU or ASCII protocol. The devices can be Modicon devices, other host computers or even other system applications using the API.

The **Modbus Serial Communication Interface API** enables a system to act as the master device to other Modbus devices, initiating read and write operations on behalf of the system applications. The API also allows the system to emulate a Modicon PLC to respond to read and write requests and thus acts as a “virtual PLC” or slave to the Modbus master device. The API is available for different Windows/WinCE-based systems/platforms and can be used with C/C++ or Visual Basic.

The API consists of two component functionalities, client (master) side and server (slave) side. The client side functionality is implemented with a single API DLL. Server side functionality is implemented with a DLL/executable pair. Together these components manage all aspects of the protocol and data exchange including responding to the master device with proper acknowledgements, error/success codes and protocol data byte ordering. The system application need only to deal with the data values exchanged in native byte order. The user can employ either the API’s client, server or both functionalities with minimal code implementation.

The **Modbus Serial Communication Interface API** supports the following Modbus function codes for both the RTU and ASCII protocol:

- 1** - Read Coil Status
- 2** - Read Input Status
- 3** - Read Holding Registers
- 4** - Read Input Registers
- 5** - Force Single Coil
- 6** - Preset Single Register
- 7** - Read Exception Status
- 15** - Force Multiple Coils
- 16** - Preset Multiple Registers

***“Modbus Serial Communication Interface API”***



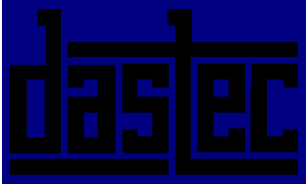
**DASTEC Corporation**  
**457A Carlisle Drive**  
**Herndon, VA 20170 USA**  
**Tel: 1-703-709-0515**  
**Fax: 1-703-709-0985**  
**www.dastec.com**

## **Client API Functionality**

To exchange data with Modbus devices, a system application can initiate read and/or write operation(s) to the devices by simply calling the client DLL functions. The functions include the ability to configure the communication serial port(s), create handle(s) for the device(s) and then using those handle(s) to call DLL read and write functions. Operation results are returned directly to the calling application, as is data in the case of read operations.

### **Client API Supports**

- One or more serial communication ports with configurable communication parameters.
- Modbus RTU or ASCII protocol.
- Global addressing.
- Defining of multiple device(s) representing Modbus devices.
- Functions to read data from and write data to defined device(s).
- Coil and Discrete Input data can be read via as packed bits or as bytes.
- Coil data can be written via as packed bits or as bytes.
- Multiple user applications can use the client API simultaneously.



**DASTEC Corporation**  
**457A Carlisle Drive**  
**Herndon, VA 20170 USA**  
**Tel: 1-703-709-0515**  
**Fax: 1-703-709-0985**  
**www.dastec.com**

## **Server API Functionality**

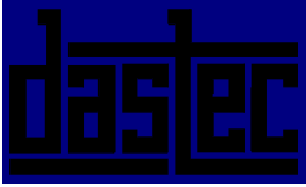
Using the server executable program provided as part of the API, a system can appear as a Modbus device (Modicon PLC) on the network. By managing all Modbus slave communication operations, the server executable receives and responds to both read and write requests from the master device. Data written by the master is stored into separate databases maintained by the server executable for each Modbus data type (Coil, Discrete Input, Input Register, Holding Register and Exception Status). The data received is stored in the byte order appropriate for the system's processor. The data for read requests received from the master is retrieved and returned from these databases as well.

No user programming is required for the system to act in this server mode. The master Modbus device can read and write to the system as if it were a Modicon PLC with no user application programs running. Applications have access to the server's databases indirectly through specific API DLL functions calls or directly via shared memory access. In this way, the user applications obtain data written from the master and can update the system's "virtual PLC's" data registers, coils, etc. so that the master can retrieve it.

### Server API Supports:

- One serial communication port with configurable communication parameters.
- Modbus RTU or ASCII protocol.
- Global addressing.
- Modbus Data Types: Coil, Discrete Input, Input Register, Holding Register and Exception Status. A separate database is maintained for each data type.
- Adjustable database sizes (except Exception Status)
- Access to databases via function calls and/or through shared memory.
- Multiple user applications can access the server databases simultaneously.

***"Modbus Serial Communication Interface API"***

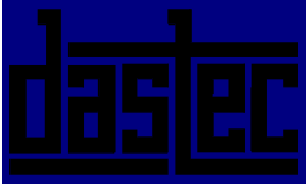


**DASTEC Corporation**  
457A Carlisle Drive  
Herndon, VA 20170 USA  
Tel: 1-703-709-0515  
Fax: 1-703-709-0985  
www.dastec.com

## **Specifications:**

- Supported Platform:
  - Any computer running Windows 98/NT/2000/XP operating system
  - WinCE
    - Intelligent Instrumentation EDAS CE and LanPoint CE
- Supported serial communication ports: 1 – 32
- Supported Modbus protocols: RTU (binary) or ASCII
- Supported Modbus function codes:
  - 1** - Read Coil Status (Maximum 2000 Coils)
  - 2** - Read Input Status (Maximum 2000 Discrete Inputs)
  - 3** - Read Holding Registers (Maximum 125 Holding Registers)
  - 4** - Read Input Registers (Maximum 125 Input Registers)
  - 5** - Force Single Coil
  - 6** - Preset Single Register
  - 7** - Read Exception Status
  - 15** - Force Multiple Coils (Maximum 800 Coils)
  - 16** - Preset Multiple Registers (Maximum 100 Registers)
- Global addressing supported.
- Supports multiple, multi-thread user applications simultaneously
- Client API Supports
  - One or more serial communication ports with configurable communication parameters.
  - Defining of multiple device(s) representing Modbus devices.
  - Read data from and write data to defined device(s) via different function calls.
  - Coil and Discrete Input data can be exchanged as packed bits or as bytes.
  - Multiple user applications can use the client API simultaneously.
- Server API Supports:
  - One serial communication port with configurable communication parameters.
  - Modbus Data Types: Coil, Discrete Input, Input Register, Holding Register and Exception Status.
  - Separate databases maintained for each data type.
  - Adjustable database sizes (except Exception Status): 0 - 65536 (1000 default).
  - Access to databases by multiple applications.
  - Access to databases via function calls or through shared memory.

***“Modbus Serial Communication Interface API”***



DASTEC Corporation  
457A Carlisle Drive  
Herndon, VA 20170 USA  
Tel: 1-703-709-0515  
Fax: 1-703-709-0985  
www.dastec.com

## Ordering Information:

Product Name: **Modbus Serial Communication Interface API**

Platform Part Numbers:

- Computer running Windows 98/NT/2000/XP  
Part No: WinPC-MBUSAPI
- WinCE
  - Intelligent Instrumentation EDAS CE and LANpoint CE  
Part No: IIIWinCE-MBUSAPI

Software package includes 3-1/2" diskette and User's manual (electronic image)

*Contact DASTEC about other supported platforms or to inquire about supporting other platforms.*

***"Modbus Serial Communication Interface API"***