DC20D MKII CONTROLLER COMMUNICATION PROTOCOL

This document is applicable to DC20D MKII controllers.

Software Version

No.	Version	Date	Note
1	V1.0	2021-06-30	Original release.
2	V1.1	2023-05-10	Change 01H control code and CRC code.
3	V1.2	2023-11-20	Updated data description

MEBAY®
Chongqing Mebay Technology Co.,Ltd
Add: No6-2,Building 4, Gangan Rd, Jiangbei District, Chongqing.
Tel：+86-23-6869 3061
Fax：+86-23-6765 8207
Web：http://www.mebay.cn
http://www.cqmb.cn
E_mail：sales@mebay.cn

Notes:

Catalogue

Summary

This communication protocol describes in detail the read and write command format of this machine's serial port communication and the definition of internal information data for third-party development and use.

The MODBUS communication protocol allows the effective transmission of information and data between the controller and a third-party programmable sequence device (PLC), RTU, SCADA system, DCS, or a third-party MODBUS compatible monitoring system. A set of monitoring system can be established by adding a set of central communication master control display software based on PC (or industrial computer).

MODBUS basic rules

All communication circuits should follow the master and slave mode.In this way, data can be transferred between a master station (such as a PC) and 32 sub-stations.

- ◆ No communication can be started from the slave station.
- ◆ All communications on the loop are transmitted in the form of "information frames".
- ◆ If the master station or the slave station receives an information frame containing an unknown command, it will not respond.

Data frame format

The communication transmission is asynchronous, and the unit is byte (data frame). Each data frame transmitted between the master station and the substation is a serial data stream of 10 bits (stop bit is 1 bit) or 11 bits (stop bit is 2 bits).

Start bit	1 bit
Data bit	8 bit
Parity check	None
Stop bit	1 bit
Baud rate	19200

Communication rules

When a communication command is sent to the instrument, the device that matches the corresponding address code receives the communication command, removes the address code, reads the information, and if there is no error, executes the corresponding task, and then returns the execution result to the sender. The returned information includes the address code, the function code to execute the action, the data after the action is executed, and the error check code (CRC). If there is an error, no information will be sent.

◆ **Information frame format**

Initial structure	Address code	Function code	Data area	CRC	End structure
Delay (equivalent to 4 bytes of time)	1 byte 8 bit	1 byte 8 bit	N byte N*8 bit	2 byte 16 bit	Delay (equivalent to 4 bytes of time)

◆ **Address code**

The address code is the first data frame (8 bits) in the information frame transmitted in each communication. The address range of the device is 1-255. This byte indicates

that the slave of the address code set by the user will receive the information sent by the master, and each slave has a unique address code, and the response is sent back with its own The address code starts. The address code sent by the host indicates the address of the slave machine to be sent to, and the address code sent by the slave machine indicates the address of the slave machine sent back.

◆ **Function code**

The function code is the second data transmitted in each communication. The MODBUS communication protocol defines the function code as 1-255 (01H-0FFH). This machine uses part of the function codes. Send as a host request, tell the slave what action to perform through the function code. As the slave machine response, the function code sent by the slave machine is the same as the function code sent by the master, and it indicates that the slave machine has responded to the operation of the master. If the highest bit of the function code sent by the slave is 1 (function code>127), it indicates that the slave has no response or an error.

The following table lists the specific meanings and operations of the function codes.

Function code	definition	operational
03H	Read register	Read one or more register data
06H	Write register	write single register data
10H	Write register	Write keyboard register data

03HRead register

The host uses the communication command with the function code of 03H to read the value register in the device (the value register saves the collected various analog quantities and parameter settings). The input register values of the data area mapped by the function code 03H are all 16 bits (2 bytes). In this way, the register values read from the device are all 2 bytes. The maximum number of registers that can be read at one time is 125. The command format of the slave machine response is slave machine address, function code, data area and CRC code. The data in the data area is a double-byte number with every two bytes as a group, with the high byte first.

06H Write register

The host uses this command to save data to the memory in the device. In the MODBUS communication protocol, the register refers to 16 bits (that is, 2 bytes), and the high bit is first. The points of such devices are all two bytes. The command format is slave address, function code, data area and CRC code.

10HWrite register

The host uses this command to save data to the memory in the device. In the MODBUS communication protocol, the register refers to 16 bits (that is, 2 bytes), and the high bit is first. The points of such devices are all two bytes. The command format is slave address, function code, data area and CRC code.

◆ **Data area**

The data area varies with the function code.

1. The format of the data area corresponding to the function code 03H:

Host sends:

Data sequence	1	2
Data meaning	Initial address	Number of read registers
Number of bytes	2	2

Slave response

Data sequence	1	2
Data meaning	Number of bytes sent back	N register data
Number of bytes	1	N

2. Data area format corresponding to function code 06H:
Host sends:

Data sequence	1	2
Data meaning	Register address	Register value
Number of bytes	2	2

Slave response

Data sequence	1	2
Data meaning	Register address	Register value
Number of bytes	2	2

3. Data area format corresponding to function code 10H:
Host sends:

Data sequence	1	2	3
Data meaning	Register address	Register number	Register value 0~N
Number of bytes	2	2	2*N

Slave response

Data sequence	1	2
Data meaning	Register address	Register number
Number of bytes	2	2

◆ **CRC code**

The master or slave can use the check code to judge whether the received information is wrong. Sometimes, due to electronic noise or some other interference, the information will change slightly during the transmission. The error check code ensures that the host or slave does not work on the information that is wrong during the transmission. This increases the safety and efficiency of the system. The error check code adopts the CRC-16 check method.
Two-byte error check code, high byte first, low byte last (can be set to low byte first and high byte last through the controller).

Note: The format of the information frame is the same: address code, function code, data area and error check code.

Redundant cyclic code (CRC) contains 2 bytes, namely 16-bit binary. The CRC code is calculated by the sender and placed at the end of the sent message. The device at the receiving end recalculates whether the CRC code of the received message is the same as the received one. If the two are different, it indicates an error.

The calculation method of the CRC code is to first preset the 16-bit registers to all 1. Then gradually process every 8 bits of data information. When calculating the CRC code, only 8 data bits are used, and the start bit and stop bit do not participate in the CRC code calculation.

When calculating the CRC code, the 8-bit data is different from the register data. The result obtained is shifted by one bit to the lower bit, and the highest bit is filled with 0. Check the lowest bit again, if the lowest bit is 1, XOR the contents of the register with the preset number, if the lowest bit is 0, no XOR operation will be performed.

This process has been repeated many times. After the 8th shift, the next 8 bits are different from the contents of the current register. This process is repeated 8 times as last time. When all the data information is processed, the content of the final register is the CRC code value.

The calculation steps of CRC-16 code are:
1. Set the 16-bit CRC register to hexadecimal FFFF;
2. XOR an 8-bit data with the lower 8 bits of the CRC register, and put the result in the CRC register;
3. Shift the content of the CRC register to the right by one bit, fill the highest bit with 0, and check the shifted out bit;
4. If the lowest bit is 0: repeat step 3 (shift again), If the lowest bit is 1: The CRC register is XORed with the hexadecimal number A001;
5. Repeat steps 3 and 4 until the right shift is 8 times, so that the entire 8-bit data has been processed;
6. Repeat steps 2 to 5 for the next data processing;
7. The final CRC register value is the CRC code. When transmitting, the lower 8 bits are sent first, and the upper 8 bits are sent last.
Note: The calculation of the CRC code starts from the <slave address>, except for all the bytes of the <CRC code>.

◆ **Transmission rate of information frames**

Note:Data frame transmission response time, minimum 500ms.

◆ **Example of Information frame format**

Function code 03H

The slave address is 10H, 3 data with the starting address of 1000H (each data is 2 bytes).

The data address in this example is:

Address	Data (Hexadecimal)
1000H	0020H
1001H	0023H
1002H	0026H

Host send	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Send to slave 10
Function code	1	03 Read register
Initial address	2	10 The starting address is 1000H

		00
Number of reads	2	00 Read 3 data (6 bytes in total) 03
CRC code	2	4A CRC code calculated by the host 02

Slave response	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Return slave address 10
Function code	1	03 Read register
Number of bytes read	1	06 3 data (6 bytes in total)
Point 1 data	2	00 The address is the content within 1000H 20
Point 2 data	2	00 The address is the content within 1001H 23
Point 3 data	2	00 The address is the content within 1002H 26
CRC code	2	F2 CRC code calculated by the slave 10

Function code 06H

The slave address is 10H, Set the content of the one points with the starting address 2001H .

The point data address in this example is:

Address	Data (Hexadecimal)	Data meaning
2001H	1111H	Keyboard commands

Host send	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Send to slave 10
Function code	1	06 Write register
Initial address	2	20 The starting address is 2001H 01
Data	2	11 Set point data (2 bytes in total) 11
CRC code	2	D7 CRC code calculated by the host 1C

Slave response	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Send to slave 10
Function code	1	06 Write register
Initial address	2	20 The starting address is 2001H 01
Data	2	11 Set point data (2 bytes in total) 11
CRC code	2	D7 CRC code calculated by the host

		1C

Function code 10H

The slave address is 10H, Set the content of the two points with the starting address 2000H (the two points must be written together when writing, otherwise it will be invalid).

The point data address in this example is:

Address	Data (Hexadecimal)	Data meaning
2000H	00000	Controller password
2001H	1111H	Keyboard commands

Host send	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Send to slave 10
Function code	1	10 Write register
Initial address	2	20 The starting address is 2000H 00
Number of writes	2	00 Write 2 data 02
Number of data	1	04 4 bytes in total
Data 1	2	00 Set 1 point data (2 bytes in total) 00
Data 2	2	11 Set 1 point data (2 bytes in total) 11
CRC code	2	0E CRC code calculated by the host F6

Slave response	Number of bytes	For example（Hexadecimal）
Slave address	1	10 Return slave address 10
Function code	1	10 Write register
Initial address	2	20 The starting address is 2000H 00
Number of writes	2	00 Write 2 data (2 bytes in total) 02
CRC code	2	49 CRC code calculated by the host 49

◆ **Error handling**

If the information received from the host is wrong, it will be ignored by the device.

Address and data

◆ **Table 1: Data area mapped by function code 03H**

Address (HEX)	Item	unit	calculate factor	Special values and their meaning	Length
1000H	Speed value	RPM	x 1	50000:Open state value	2Bytes
1001H	Battery voltage	V	x 0.1		2Bytes
1002H	Charging voltage	V	x 0.1		2Bytes

1003H	Generator frequency	Hz	x 0.1	20000:Disabled status value	2Bytes
1004H	Generator voltage L1	V	x 1	20000:Disabled status value	2Bytes
1005H	Generator voltage L2	V	x 1	20000:Disabled status value	2Bytes
1006H	Generator voltage L3	V	x 1	20000:Disabled status value	2Bytes
1007H	Generator voltage L1-L2	V	x 1	20000:Disabled status value	2Bytes
1008H	Generator voltage L2-L3	V	x 1	20000:Disabled status value	2Bytes
1009H	Generator voltage L3-L1	V	x 1	20000:Disabled status value	2Bytes
100AH	Generator current L1	A	x 0.1	20000:Disabled status value	2Bytes
100BH	Generator current L2	A	x 0.1	20000:Disabled status value	2Bytes
100CH	Generator current L3	A	x 0.1	20000:Disabled status value	2Bytes
100DH	Running time	H	x 0.1		2Bytes
100EH	Total running time.H	H	X6553.6	Total running time = (Total running time.H x 65536 + Total running time.L) x 0.1(H)	2Bytes
100FH	Total running time.L	H	x 0.1		2Bytes
1010H	Gear status			Reference:Gear status table	2Bytes
1011H	Oil pressure value	PSI	x 1	50000:Open state value 20000:Disabled status value	2Bytes
1012H	Water temperature value	℃	x 1	50000:Open state value 20000:Disabled status value	2Bytes
1013H	Reserve				2Bytes
1014H	Reserve				2Bytes
1015H	Reserve				2Bytes
1016H	Fuel level value	%	x 1	50000:Open state value 20000:Disabled status value	2Bytes
1017H	Oil pressure value	Kpa	x 1	50000:Open state value 20000:Disabled status value	2Bytes
1018H	Reserve				2Bytes
1019H	Switch input status			Reference: Input status table	2Bytes
101AH	Relay output status			Reference:Output status table	2Bytes

101BH	Reserve				2Bytes
101CH	Reserve				2Bytes
101DH	Reserve				2Bytes
101EH	Reserve				2Bytes
101FH	LED status			Reference:LED status table	2Bytes
1020H	Reserve				2Bytes
1021H	Reserve				2Bytes
1022H	Reserve				2Bytes
1023H	Reserve				2Bytes
1024H	Reserve				2Bytes
1025H	Reserve				2Bytes
1026H	Reserve				2Bytes
1027H	ATS status			Reference:ATS status table	2Bytes
1028H	Running status			Reference:Running status table	2Bytes
1029H	Reserve				2Bytes
102AH	Reserve				2Bytes
102BH	Reserve				2Bytes
102CH	Alarm code			Reference:Alarm code table	2Bytes
102DH	Warning code			Reference:Warning code table	2Bytes
102EH	Sensor 1 resistance	Ω	x 1	50000:Open state value 20000:Disabled status value	2Bytes
102FH	Sensor 2 resistance	Ω	x 1	50000:Open state value 20000:Disabled status value	2Bytes
1030H	Reserve				2Bytes
1031H	Reserve				2Bytes

◆ **Table 2: Data area mapped by function code 06H(Write only support)**

Address (HEX)	Item	Special instructions	Length
2001H	Key command	Stop：1111H Manual：2222H Automatic：3333H Start：5555H	2Bytes

◆ **Table 3: Data area mapped by function code 10H(Write only support)**

Address (HEX)	Item	Special instructions	Length
2000H	user password	unsigned	2Bytes

2001H	Key command	Stop：1111H Manual：2222H Automatic：3333H Start：5555H	2Bytes

◆ **Input status table**

Address (HEX)	Data bit	Information	Special instructions	Length
1019H	bit0	AUX. INPUT 1	Binary 1 is valid	1Bit
	bit1	AUX. INPUT 2	Binary 1 is valid	1Bit
	bit2	AUX. INPUT 3	Binary 1 is valid	1Bit

◆ **Output status table**

Address (HEX)	Data bit	Information	Special instructions	Length
1033H	bit0	FUEL OUTPUT	Binary 1 is valid	1Bit
	bit1	CRANK OUTPUT	Binary 1 is valid	1Bit
	bit2	AUX.OUTPUT 1	Binary 1 is valid	1Bit
	bit3	AUX.OUTPUT 2	Binary 1 is valid	1Bit

◆ **Gear status table**

Address (HEX)	Information
1010H	0033H：Stop
	0066H：Manual
	0099H：Auto

◆ **ATS status table**

Address (HEX)	Information
1027H	0066H：NC
	0099H：GEN Closing

◆ **Running status table**

Running Status Code (HEX)	Information	Special instructions
0000H	Stop Idle Speed	
0001H	Under stop	
0002H	Waiting	
0003H	Crank Cancel	
0004H	Crank Interval	
0005H	Alarm,Reset	This state does not display the delay value
0006H	Standby	This state does not display the delay value
0007H	Pre-heat	
0008H	Pre-oil Supply	
0009H	Crank Delay	
000AH	Crank Ready	
000BH	In Crank	
000CH	Safety Delay	
000DH	Idle speed	
000EH	Speed-up	

000FH	Temperature-up	
0010H	Volt-built/up	
0011H	High-speed warming	
0012H	Rated running	This state does not display the delay value
0013H	Mains revert	
0014H	Cooling running	
0015H	Gen return	
0016H	Under stop by radiator	This state does not display the delay value
0017H	Switching	This state does not display the delay value

◆ **LED status table**

Address	Data bit	Information	Special instructions	Length
101FH	bit0	Gens loading	Binary 1 is valid	1Bit

◆ **Alarm code table**

Alarm code(HEX)	Information
0000H	None
0001H	Emergency stop
0002H	Over speed
0003H	Under speed
0004H	Low oil pressure sensor
0005H	Low oil pressure switch
0006H	High water temperature sensor
0007H	High water temperature switch
0008H	High oil temperature switch
0009H	Low fuel level switch
000AH	Low water level switch
000BH	RPM Signal lost
000CH	Under battery voltage
000DH	Oil pressure sensor open
000EH	Water temperature sensor open
000FH	Over frequency
0010H	Under frequency
0011H	Over voltage
0012H	Under voltage
0013H	Over current
0014H	Crank failure
0015H	Stop failure/ with RPM
0016H	Stop failure/ with Hz
0017H	Stop failure/ with oil pressure
0018H	Stop failure/ oil pressure switch