Debug DPM8600 with serial assistant

1、 According to the "simple communication protocol" and "MODBUS communication protocol" provided by us, we can control the DPM8600. The two communication protocols can be freely selected in the DPM8600 function "5-CS", 0 means "custom communication protocol", 1 means "MODBUS Communication Protocol"

2、 You can use the serial port assistant to debug commands before programming with the communication protocol.

3、 For example: 01w10=1234,\r\n, indicating that the voltage setting is: 12.34V

Simple communication protocol

Overview

The structure of the control instruction is command line. The communication rate can be selected between seven baud rates (2400, 4800, 9600, 19200, 38400, 57600, 115200). The machine address code can be set in the range of 01-99. For the specific setting operation, please refer to the manual.

The command is sent by the PC, and the machine analyzes and executes， when the address code is the same, the result is returned to the PC. When the address code is different, no information is returned. This is very suitable for the centralized control of multiple machines.

Send command format is as follows:

start symbol	address code	function symbol	function number	equal symbol	operand	comma	end code
:	01~99	w,r	00~99	=	0~65535	,	\r\n

1、 The start symbol is ":"

2、 The address code is the local address, and the setting range is 01~99.

3、 The function code is ‘w’ or ‘r’, indicating write or read.

4、 The function number is the number for different functions, and the different values represent different parameter settings.

5、 The operand is the operand of the command

6、 Comma: Each operand is distinguished by ","

7.End code:The end of a command is ‘\r\n’, this is actually a return character, and a newline character in ASCII, hexadecimal representation is 0x0d, 0x0a.

I、w instruction

(1) 10 command write voltage setting.
The format is :01w10=****,
"****" means a value, indicates the value of voltage setting.
For example: 01w10=1234,\r\n, indicates the voltage setting is: 12.34V

(2) 11 command write current setting.
The format is :01w11=****,
"****" means a value, indicating the value of current setting.
For example: 01w11=12345, indicating the current setting is: 12.345A

(3) 12 command Output status setting
The format is :01w12=*,
"*" means a value, representing the output status
For example: :01w12=0, means to turn off the output
:01w12=1,\r\n, means to turn on the output.

(4) 20 command Set both voltage and current settings The format is
:01w20=****,####,
"****"means the voltage setting,"####"means the current setting
For example: 01w20=1234,12345, means the voltage and current setting: 12.34V, 12.345A

I、 r instruction

(1) 00 command Read maximum output voltage
The format is :01r00=0,
For example: send :01r00=0, the return is :01r00=6000,indicates the maximum output voltage is 60V.

(2) 01 command Read maximum output current
The format is :01r01=0,
For example: send :01w01=0,
If the return is :01r01=24000, indicates the maximum output current of this model is 24A, and the corresponding model is DPM-8624.
If the return is :01r01=16000, indicates the maximum output current of this model is 16A, and the corresponding model is DPM-8616.
If the return is :01r01=8000, indicates the maximum output current of this model is 8A, and the corresponding model is DPM-8608.
If the return is :01r01=5000, indicates the maximum output current of this model is 5A, and the corresponding model is DPM-8605.

(3) 10 command Read the voltage setting
The format is :01r10=0,
For example: send :01w10=0, the return is :01r10=1234, indicates the voltage setting is 12.34V

(4) 11 command Read the current setting
The format is :01r11=0,
For example: send :01r11=0, the return is :01r11=12345, indicates the

current setting is 12.345A

(5) 12 command Read output status

The format is :01r12=0,

For example: send :01w12=0,

If the return is :01r12=0, indicates the output status is currently off.

If the return is :01r12=1,indicates the output state is currently on.

(6) 30 command Read output voltage measurement value

The format is :01r30=0,

For example: send :01r30=0, the return is :01r30=2345, indicates the measured value of the output voltage is 23.45V.

(7) 31 command Read the output current measurement value

The format is :01r31=0,

For example: send: 01r31=0, the return is :01r31=12345, indicates the measured value of the output current is 12.345A

(8) 32 command Read CC or CV status

The format is :01r32=0,

For example: send :01w32=0,

If the return is :01r32=0, indicates constant voltage output(CV)

If the return is :01w32=1, indicates constant current output(CC)

(9) 33 command Read temperature

The format is :01r33=0,

For example: send :01r33=0, the return is :01r33=30, Indicates that the internal temperature of the machine is 30 ° C

Modbus Communication Protocol

Communication protocol overview:

The communication protocol for this device is specifically MODBUS RTU. This can be used via RS485Note, This product only supports MODBUS RTU Function Codes 0x03 (03H), 0x06 (06H), 0x10 (10H)

II. Communication protocol described:

Message transmission is asynchronous, Modbus-RTU is in 11-bit bytes.

(For a full description of the general MODBUS RTU Protocol see http://modbus.org/docs/PI_MBUS_300.pdf)
Modbus is a generic protocol. The implementation of the location and function of registers is specific to the designer of this device however the format of the command structure for the Modbus Communication is fixed and as detailed in the above referenced document. Below is a brief introduction to the Standard Modbus RTU protocol and the registers and Modbus RTU Function Codes used in this specific device.

Data Format:

Serial data format	10 (binary system)
Start bit	1
Data bits	8
Parity bit	No
Stop bit	1

Data frame structure:

Data-frame interval	**Address code**	**Function code**	**Data area**	**CRC Check**
3.5 Bytes more	1 Bytes	1 Bytes	N Bytes	2 Bytes

Before sending data, the bus silent period (no data transmition time) is more than 3.5 character, (eg: Baud rate is 9600, the time is 5ms).
RTU messages start with a silent interval of at least 3.5 character time periods.Following the last transmitted character, a similar interval of at least 3.5 character times marks the end of the message. A new message can begin after this interval.
The entire message frame must be transmitted as a continuous stream. If a silent interval of more than 3.5 character times occurs before completion of the frame, the receiving device flushes the incomplete message and assumes that the next byte will be the address field of a new message.
Similarly, if a new message begins earlier than 3.5 character times following a previous message, then the receiving device will consider it to be a continuation of the previous message. This will set an error, as the value in the final CRC field will not be valid for the combined messages.

1.1 Address Field:

The address field of a message frame contains eight bits, from 1 to 255. This byte shows the slaver of setting address by user will receive the message form main. Every slaver must have unique address field, and only the salver fits the address field, it can respond loopback message. When slaver send the message back, the loopback data will begin with its own address filed. The address field sent by the main is the address filed the salver received, and the loopback address filed is the loopback slaver filed address. The corresponding address filed show where it come from.

1.2 Function Field:

The Function Code field tells the addressed slave what function to perform. This device only supports the functions 0x03 0x06 0x10.

Function code	**Define**	**Operations (binary)**
0x03	Read Holding Register	Read one or more registers in the slave.
0x06	Write single register	Writes binary data to a single register.
0x10	Write multiple registers	Writes values into a sequence of holding registers.

1.3 Data area:

The data bytes contain any additional information that the slave will need to perform the function. For example, function code 03 will request the slave to read holding registers and respond with their contents. The data field must contain the information telling the slave which register to start at and how many registers to read. In the Response, the data bytes contain the data collected from the slave, such as register values or status. If an error occurs, the function code is modified to indicate that the response is an error response, and the data bytes contain a code that describes the error.

The 3 MODBUS RTU Function Codes used by this device: (ONLY these three codes)

0x03 Read Holding Register(s):

Address code	**Function code**	**Register starting address**	**Number of register addresses n**	**CRC Check**
1 Bytes	1 Bytes	2 Bytes	2 Bytes	2 Bytes

0x03 Read Holding Register(s) function returns the following format.

Address code	Function code	Returns the number of bytes 2*n	Register data	CRC Check
1 Bytes	1 Bytes	1 Bytes	2*n Bytes	2 Bytes

0x06 Write Single Register.

Address code	Function code	Register address	Register data	CRC Check
1 Bytes	1 Bytes	2 Bytes	2 Bytes	2 Bytes

0x06 Write Single Register function returns the following format.

Address code	Function code	Register address	Register data	CRC Check
1 Bytes	1 Bytes	2 Bytes	2 Bytes	2 Bytes

0x10 Write Multiple Registers.

Address code	Function code	Register starting address	Number of register addresses n	Number of bytes written 2*n	Register data	CRC Check code
1 Bytes	1 Bytes	2 Bytes	2 Bytes	1 Bytes	2*n Bytes	2 Bytes

0x10 Function: Returns the following format.

Address code	Function code	Register starting address	Number of register address n	CRC Check code
1 Bytes	1 Bytes	2 Bytes	2 Bytes	2 Bytes

Register Map for this device.

name	explain	Number of bytes	radix point	unit	read-write	postal address
Set-U	Voltage setting value	2	2	V	r/w	0000H
Set-I	Current setting value	2	3	A	r/w	0001H
SW	out switch	2	0	-	r/w	0002H
CCCV	output state	2	0	-	r	1000H
U	Voltage display value	2	2	V	r	1001H
I	Current display value	2	3	A	r	1002H
T	Temperature display value	2	0	℃	r	1003H

Note:

1.The Set-UFT-Set-I is two set values used to control the voltage and current settings of the power supply
2.Control whether the output is turned on, write 0 off, write 1 open, and can also read the state.
3.This is a read-only register, readout 0 means no output 1 for CV status and 2 for CC state.
4.U、I is two real-time measurements used to read back the voltage and current values of the power supply.
5.T is the temperature measurement value, which is used to read the internal temperature of the power supply.

1.4 Error-checking:

In RTU mode, messages include an error–checking field that is based on a Cyclical Redundancy Check (CRC) method. The CRC field checks the contents of the entire message. It is applied regardless of any parity check method used for the individual characters of the message.

CRC Calculation method:

The Cyclical Redundancy Check (CRC) field is two bytes, containing a 16-bit binary value. The CRC value is calculated by the transmitting device, which appends the CRC to the message.

The receiving device recalculates a CRC during receipt of the message, and compares the calculated value to the actual value it received in the CRC field. If the two values are not equal, an error results.

The CRC is started by first preloading a 16-bit register to all 1's. Then a process begins of applying successive eight-bit bytes of the message to the current contents of the register. Only the eight bits of data in each character are used for generating the CRC. Start and stop bits, and the parity bit, do not apply to the CRC.

During generation of the CRC, each eight-bit character is Exclusive ORed with the register contents. The result is shifted in the direction of the least significant bit (LSB), with a zero filled into the most significant bit (MSB) position.

The LSB is extracted and examined. If the LSB was a 1 then an Exclusive OR (XOR) of the register with a preset, fixed value takes place. If the LSB was a 0, no Exclusive OR (XOR) takes place.

This process is repeated until eight shifts have been performed. After the last (eighth) shift, the next eight-bit character is exclusive ORed with the register's current value, and the process repeats for eight more shifts as described above. The final contents of the register, after all the characters of the message have been applied, is the CRC value.

Three examples of communication using the three available MODBUS Functions Codes supported on this device. (0x03) (0x06) and (0x10)

1 : **Read the displayed output voltage and current value.** (Using Function 0x03)

Message format sent from Host:

Host sends	Number of bytes	Information sent	Notes
Slave address	1	01	From host to slave address 01H
Function code	1	03	Read holding register(s)
Register starting address	2	0000H	Register starting address
Number of registers to read	2	0002H	A total of 2 registers (4 byte = 2 words)
CRC Checksum:	2	C40BH	CRC Checksum from Host:

For example if the displayed values are 05.00V , 5.000A , The Slave return response would be:

Slave responds	Number of bytes	The information returned	Notes
Slave address	1	01	From slave address 01H
Function code	1	03	Read holding register
Number of bytes read	1	04	A total of 4 Bytes
The contents of 1st register read	2	01F4H	Output voltage display values
The contents of 2nd register read	2	1388H	Output current display value
CRC Checksum:	2	B76BH	CRC Checksum from Slave

2 : **Host to set the voltage to 24.00V. (Using Function 0x06)** Message format sent from host:

Host sends	Number of bytes	Information sent	Notes
Slave address	1	01H	From host to slave address 01H
Function code	1	06H	Write single register
Register address	2	0000H	Register address
The value to write to the register	2	0960H	Set the output voltage
CRC Checksum	2	8FB2H	CRC Checksum from host

Message format returned from Slave:

Slave responds	Number of bytes	The information returned	Notes
Slave address	1	01H	From slave address 01H
Function code	1	06H	Write single register
Register address	2	0000H	The Register address
The contents of the register read	2	0960H	Value read
CRC Checksum	2	8FB2H	CRC Checksum from slave

3 : Host to set the voltage to 24.00V , The current 1.500A. (Using Function 0x10 (decimal16))

Message format sent from HOST:

Host sends	Number of bytes	Information sent	Notes
Slave address	1	01H	From host to slave address 01H
Function code	1	10H	Write multiple registers
Register starting address	2	0000H	Register the starting address
Quantity of registers to write to	2	0002H	2 registers
Number of bytes to write	1	04H	4 Bytes = 2 words = 2 registers
Contents of 1st register	2	0960H	Set the output voltage
Contents of 2nd register	2	05DCH	Set the output current
CRC Code:	2	F2E4H	CRC Checksum from Host:

Message format returned from Slave

From machine responses	Number of bytes	The information returned	Notes
Slave address	1	01H	From Slave address 01H
Function code	1	10H	Write multiple registers
Register the starting address	2	0000H	Register starting address
Number of registers written	2	0002H	2 registers written to
CRC Code:	2	41C8H	CRC Checksum from Slave: