

X-Hybrid II Inverter upgrading communication protocol

1. Protocol between Client Computer and Core Chip

We connect the Client computer and chip through cable, actually, every command contains 12 bytes, shows as below

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5
Transaction Process Identify	Transaction Process Identify	Functional Domain Identify	Functional Domain Identify	Length Field(High)	Length Field(Low)
Byte6	Byte7	Byte8	Byte9	Byte10	Byte11

The first two bytes are defined different in different protocols.

The 3rd and 4th bytes are the function identify: 0x00 0x00 are defined as the ATE Modbus protocol;

0xAA 0x55 are defined as the data upgrade requirement of Client Computer.

The 5th and 6th bytes are the length of whole command

The 7th to 12th bytes will return the different data according to the different protocol, such as the content of Modbus protocol or upgrading control command.

2. Protocol upgrade

Pre-upgrading command: judge the different branch of upgrading;

For instance:

Inquire present firmware: 0x40 00 AA 55 00 0B 00 00 00 00 00 00

Instruction:

As the 1st byte is 0x40, represent pre-upgrade transaction; the first number is 0, represent inquiring for other Client Computers;

Upgrade situation, the second byte is 00 represent inquiring version number

01 enter into upgrading manager program status

02 enter into upgrading inverter program status

03 enter into upgrading charger program status

We can stop upgrade when are in the status of inquire version number, but cannot disconnect when enter into the upgrade status(restart again by restarting the ARM board if accidentally disconnect)

After enter into the upgrading, all sending content by TCP are the upgrading package, ARM is bin files and every data packet is 256 bytes, DSP is HEX files, and every packet is one line of HEX.

Core Chip Response:

The first four bytes will accord with the receiving request package; 0x40 00 AA 55

The 5th, 6th bytes are the length of package , actually is 12, 00 0B

The 7th to 12th bytes will be filled with different upgrade control commands;

Upgrade content of response:

Inquire version number: when the 7th byte is AA represent inquiring successful; 8th byte is manager version number; 9th byte is inverter version number; 10th byte is charger version number.

Enter upgrading status: when the received data packets are correct, the 7th byte is AA, otherwise is 00 and need to send again.

The status of Manager upgrade complete: when the received packager is less than 256 bytes, and send over, no reply of the chip jump; if the length of last package is 256 bytes, it will send 4 ff to over the sending data package.

3. DSP process:

When receiving the Client Computer 40 02 AA 55 00 0B 00 00 00 00 00 commands ,it represents entering into the inverter upgrading status. If the 2nd byte is 03, it represents entering into charger upgrading status.

4. DSP upgrading error handling:

1. Client Computer will check the HEX files first:
 - A, first line check: check the first line of HEX files, to avoid mixing the different DSP HEX files, such as 28335, 28035 of DSP types. The method of checking: the 13th byte of the hex first line: if is 0x0e , it represent the charger firmware; if is 0x00, it represent the inverter firmware.
 - B, check every line: every line of HEX files will have the check byte(refer to the check method of intel HEX files), they will check all over files before, and each check error will stop the upgrading next.
2. The error of burn DSP(the wrong checksum, package incomplete overtime, burn failure and so on), they all reply char 'f' error, and then the ARM will stop sending, and resend the upgrading start command to the DSP, when the DSP jump and erase successful , it will inform the Client computer restart sending the first package, and begin the new burning.
3. Network break when the DSP burning; if the break is instant, and connect immediately, the Client computer will continue send the data package(if the DSP report error right now, it will restart from the last command),if the network cannot recover, we will confirm the system are not break, then restart send data from the first package, and begin the new burning.
4. The error cannot recover when the burning break, it only restart burning through the emulator or serial port.