Universal IR Learning Remote Module

Manual for Modbus Registers and IR Learning

(Supports IR-210/IR-712A/IR-712-MTCP)

Version 1.2

2016/6/23

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for damages resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, or for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Trademark

The names used for identification only may be registered trademarks of their respective companies.

Contents

1. Basic Settings and IR Learning

1-1 Basic Parameters Settings

A Modbus Master can write communication setting values to the holding registers of an IR learning remote module for basic parameters settings. Please refer to Table 3-2 in chapter 3.

^ Basic setting registers for IR-210/IR-712A: holding register address [1122]~[1140]
^ Basic setting registers for IR-712-MTCP: holding register address [1122]~[1140] and [1142], [1143]

2-1-1 Register for Setting Basic Settings to IR-210/IR-712A

Holding register address[1140] is for writing the basic settings of holding registers[1122]~[1131] to the flash memory. The valid values for address [1140] are 0, 1, and 2. Please refer to Table 3-2.

(1) holding register [1140] = 0
Do nothing.

(2) holding register [1140] = 1
The values in the holding registers [1122]~[1131] will be effective immediately but not written to the flash memory. The basic settings will be back to the previous ones after power cycling/reboot the IR module.

(3) holding register [1140] = 2
Command the IR module to write the values of the holding registers [1122]~[1131] to the flash memory and these values take effect immediately. The new settings are still effective after power cycling/reboot the IR module.

2-1-2 Register for Setting Basic Settings to IR-712-MTCP

Holding register address[1140] is for writing the basic settings of holding registers[1122]~[1138], [1142] and [1143] to the flash memory. Please refer to Table 3-2. The valid values for address [1140] are described as follows:

(1) High Byte of the holding register [1140]
bit 0: Reboot the module
Value: 0 => do nothing,

1 => reboot

bit 1~7: reserved

(2) Low Byte of the holding register [1140]

bit 0~2: Write settings to the MCU's flash and reset(reboot) the module.

Value: 0 => nothing,

1 => Write & reboot.

2 => Only write. No reboot.

bit 3~7: reserved

The new basic settings will be effective after reboot the IR module.

1-2 Modbus Master Configures IR Command Quantity

Modbus master device should in advance configures the IR device total quantity, IR command total quantity, IR command quantity in each IR device, IR device names, IR command names (comments) before downloading/loading IR commands to/from the IR module. Please refer to Table 3-2 for the instructions of holding registers [1162] ~ [1212], [1213] ~ [1263] and [1264] ~ [1276]. Among which the “IR Command Summary” is composed of “IR device total quantity”, “IR command total quantity” and “IR command quantity in each IR device”.

1-3 Procedure for IR Learning

The flowchart of learning IR commands is depicted in figure 1-1 followed the comments in steps. (Modbus master is called Master for short)

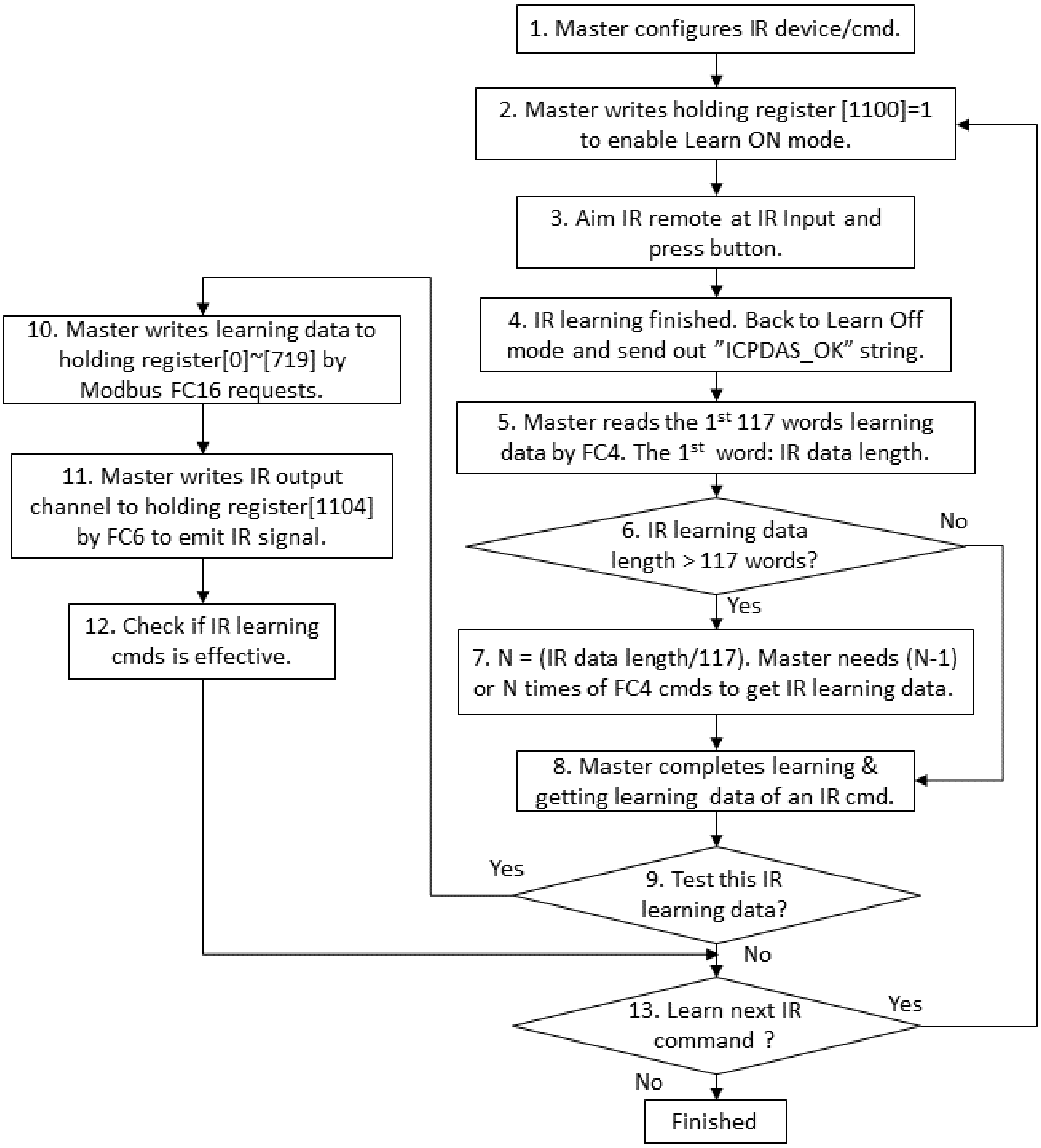

1. Master configures IR device/cmd.
2. Master writes holding register [1100]=1 to enable Learn ON mode.
3. Aim IR remote at IR Input and press button.
4. IR learning finished. Back to Learn Off mode and send out "ICPDAS_OK" string.
5. Master reads the 1st 117 words learning data by FC4. The 1st word: IR data length.
6. IR learning data length > 117 words?
No
Yes
7. N = (IR data length/117). Master needs (N-1) or N times of FC4 cmds to get IR learning data.
8. Master completes learning & getting learning data of an IR cmd.
9. Test this IR learning data?
Yes
No
10. Master writes learning data to holding register[0]~[719] by Modbus FC16 requests.
11. Master writes IR output channel to holding register[1104] by FC6 to emit IR signal.
12. Check if IR learning cmds is effective.
13. Learn next IR command ?
Yes
No
Finished

Figure 1-1 Flowchart of IR learning

- **Steps of learning an IR command**

Step 1:

Modbus master configures the number of the IR devices and IR commands for learning.

Step 2:

The master use FC6 command to write holding register [1100] = 1 to enable the IR learning mode (Learn ON, LN LED is ON).

Step 3:

Aim the emitter of the IR remote at the IR Input of the IR module (less than 3 cm) and press the button of the remote.

Step 4:

IR module has learned the IR command. LN LED is off (Learn OFF state). The "ICPDAS_OK" string (ASCII Code) is sent out to inform the master that IR learning is finished.

Step 5:

Modbus master reads back the Input Registers [0] ~ [719] (for an IR learning data) from the IR module by the FC4 commands. Firstly, read back the input registers [0] ~ [116] (i.e. 117 words = 234 bytes). Please refer to Table 3-1 for Modbus Input Registers.

Step 6:

From prior **Step 5**, the input register [0] means the total length (including register [0]) of the subsequent IR learning data. Check if the total length is larger than 117 words (234 bytes). If not, go to Step 8 to finish the learning procedure. If it is, go to **Step 7**.

Step 7:

If the length of the IR learning data is larger than 117 words, master should read back the IR learning data by Modbus FC4 for several times N:

N = (Total length of an IR learning data) / 117

If there is a remainder in the above equation, N more times of FC4 commands (read 117 words) are necessary to read back the whole data. The remainder determines the termination point of the IR learning data in the last set of 117 words.

If there is no remainder in the above equation, (N-1) more times of FC4 commands (read 117 words) are necessary to read back the whole data. Then go to **Step 8**.

Step 8:

Modbus master has complete the procedure of reading back an IR learning data in this step.

Step 9:

Determine if it needs to testing the IR learning data. Of it does, go to **Step 10**, else go to **Step 13.**

Step 10:

Modbus master should send the previous IR learning data to the holding registers [0] ~[719] (Table 3-2) by FC16 commands. Several times of FC16 commands are necessary if the length of the IR learning data is larger than 117 words. The holding register [0] is the total length of the IR learning data (including the holding register[0]). Please refer to the principle of previous Step 6 and 7.

Step 11:

Plug in an IR emitter cable in one of the IR output jacks. Aim the emitter head of the IR cable to the IR appliance. Master can request the IR module to emit the IR command signal by writing the holding register [1104] (IR output channels) by FC6. The value of the holding register [1104] will be back to zero after emitting IR signal.

Step 12:

Check if the IR appliance is controlled by the IR command. Then go to step 13.

Step 13:

If it needs to learn the IR command again or the next one, go to Step 2. Go to end if no IR commands to learn.

2 Master Download/Load IR Cmds for IR module

2-1 Format of the IR learning data file

The extension of the IR learning data file is "ird". The file includes the following information:

1. IR device quantity
2. Total IR command quantity
3. Each device's IR cmd quantity

} IR command summary (Refer to Table 3-2)

4. IR device name
5. IR command name (comments)
6. IR learning data (hexadecimal)

} IR learning data (Refer to Table 3-2)

02 10 00 00 00 46 8C 00 46 00 05 ....

where

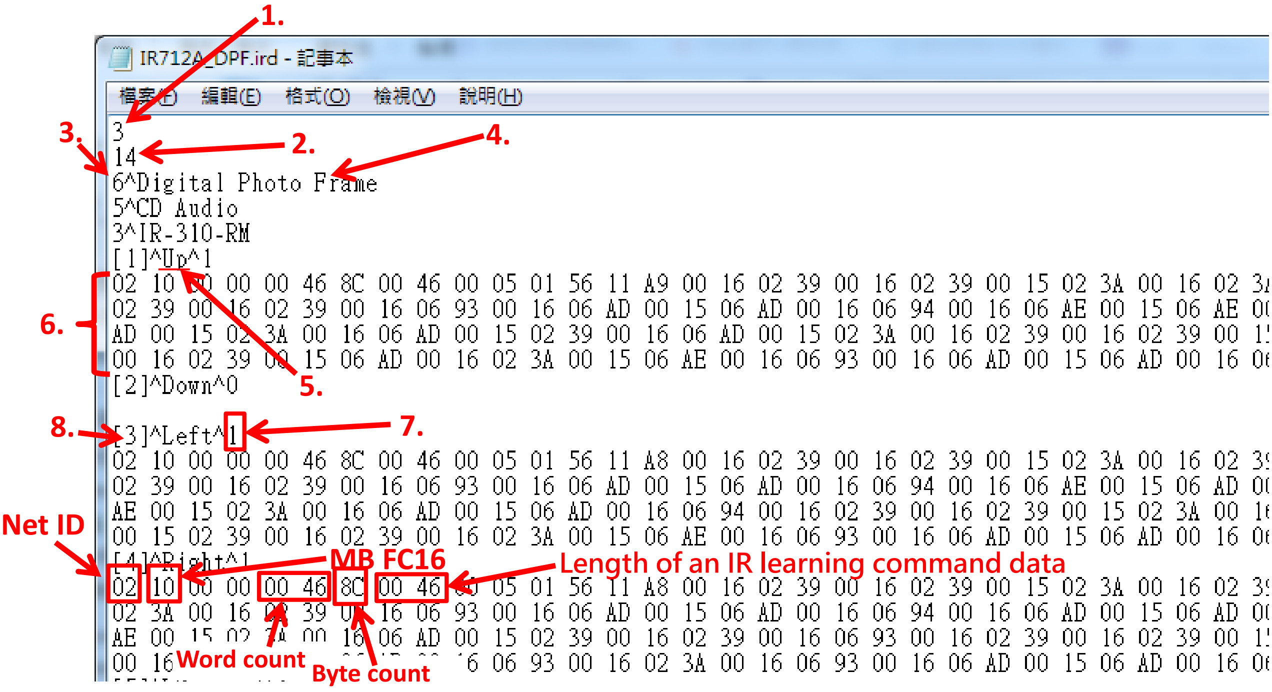
02: Net ID

10: Modbus FC16

7. The count of Modbus FC16 requests for an IR learning command.
8. IR command number (1~224 or 1~512)

In the file, the character "^" is the delimiter for IR utility.

- Example 1: IR learning data file of a digital photo frame



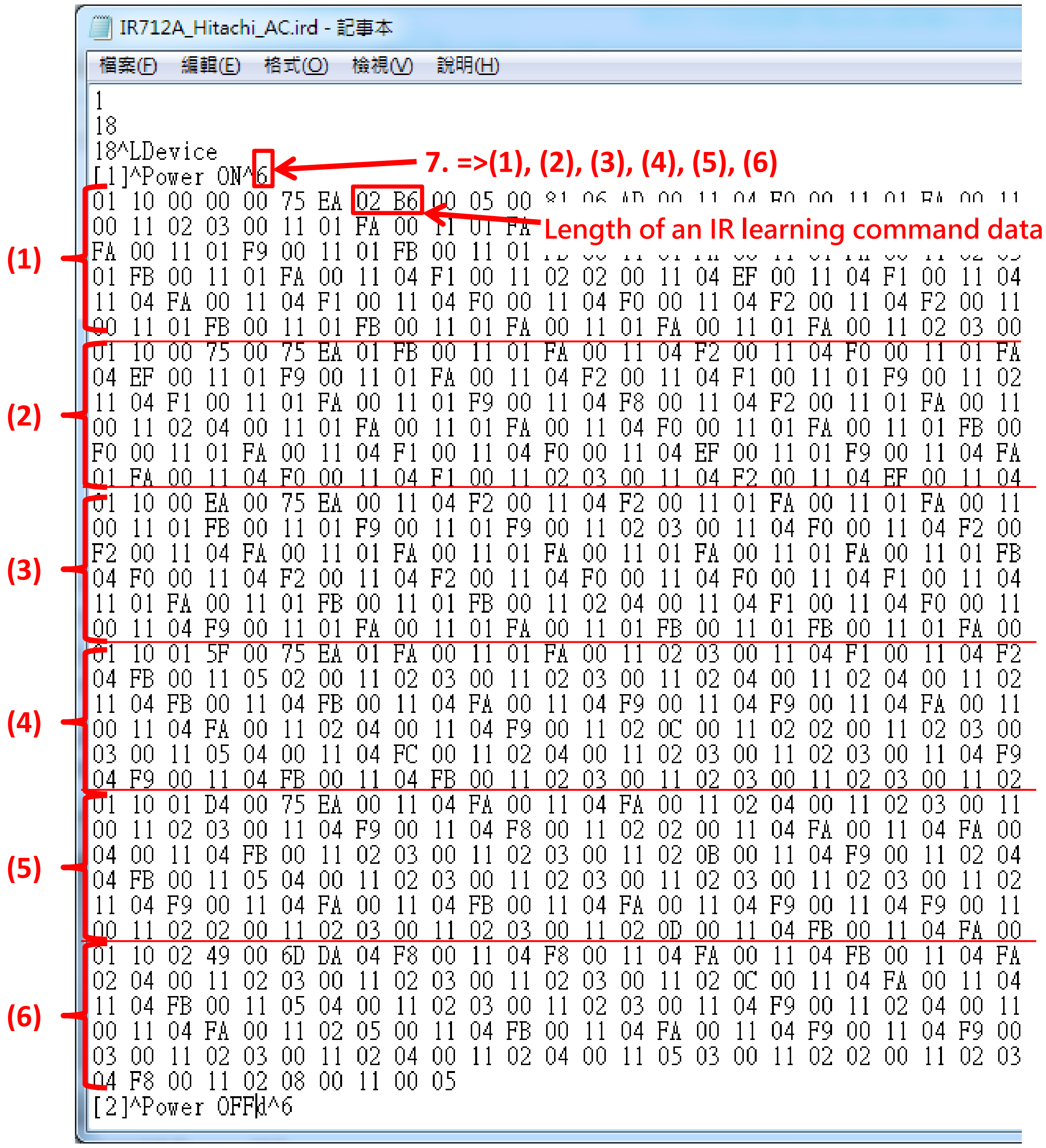
```
3
14
6^Digital Photo Frame
5^CD Audio
3^IR-310-RM
[1]^Up^1
02 10 00 00 00 46 8C 00 46 00 05 01 56 11 A9 00 16 02 39 00 16 02 39 00 15 02 3A 00 16 02 3
02 39 00 16 02 39 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 06 94 00 16 06 AE 00 15 06 AE 0
AD 00 15 02 3A 00 16 06 AD 00 15 02 39 00 16 06 AD 00 15 02 3A 00 16 02 39 00 16 02 39 00 1
00 16 02 39 00 15 06 AD 00 16 02 3A 00 15 06 AE 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 0
[2]^Down^0

[3]^Left^1
02 10 00 00 00 46 8C 00 46 00 05 01 56 11 A8 00 16 02 39 00 16 02 39 00 15 02 3A 00 16 02 3
02 39 00 16 02 39 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 06 94 00 16 06 AE 00 15 06 AD 0
AE 00 15 02 3A 00 16 06 AD 00 15 06 AD 00 16 06 94 00 16 02 39 00 16 02 39 00 15 02 3A 00 1
00 15 02 39 00 16 02 39 00 16 02 3A 00 15 06 AE 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 0
[4]^Right^1
02 10 00 00 00 46 8C 00 46 00 05 01 56 11 A8 00 16 02 39 00 16 02 39 00 15 02 3A 00 16 02 3
02 3A 00 16 02 39 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 06 94 00 16 06 AD 00 15 06 AD 0
AE 00 15 02 3A 00 16 06 AD 00 15 02 39 00 16 02 39 00 16 06 93 00 16 02 39 00 16 02 39 00 1
00 16 [illegible] 16 06 93 00 16 02 3A 00 16 06 93 00 16 06 AD 00 15 06 AD 00 16 0
```

Figure 2-1

- Example 2: IR learning data file of an air conditioner

The IR code of the air conditioner is longer. It needs multiple Modbus commands to write an IR learning data to the holding registers 4xxxx. It needs six Modbus FC16 request for this example to send an IR command data to the IR module as shown in figure 2-2.


IR712A_Hitachi_AC.ird - 記事本
檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)
7. =>(1), (2), (3), (4), (5), (6)
Length of an IR learning command data
(1)
(2)
(3)
(4)
(5)
(6)

```
1
18
18^LDevice
[1]^Power ON^6
01 10 00 00 00 75 EA 02 B6 00 05 00 81 06 AD 00 11 04 F0 00 11 01 FA 00 11
00 11 02 03 00 11 01 FA 00 11 01 FA
FA 00 11 01 F9 00 11 01 FB 00 11 01 FB 00 11 01 FA 00 11 01 FA 00 11 02 03
01 FB 00 11 01 FA 00 11 04 F1 00 11 02 02 00 11 04 EF 00 11 04 F1 00 11 04
11 04 FA 00 11 04 F1 00 11 04 F0 00 11 04 F0 00 11 04 F2 00 11 04 F2 00 11
00 11 01 FB 00 11 01 FB 00 11 01 FA 00 11 01 FA 00 11 01 FA 00 11 02 03 00
01 10 00 75 00 75 EA 01 FB 00 11 01 FA 00 11 04 F2 00 11 04 F0 00 11 01 FA
04 EF 00 11 01 F9 00 11 01 FA 00 11 04 F2 00 11 04 F1 00 11 01 F9 00 11 02
11 04 F1 00 11 01 FA 00 11 01 F9 00 11 04 F8 00 11 04 F2 00 11 01 FA 00 11
00 11 02 04 00 11 01 FA 00 11 01 FA 00 11 04 F0 00 11 01 FA 00 11 01 FB 00
F0 00 11 01 FA 00 11 04 F1 00 11 04 F0 00 11 04 EF 00 11 01 F9 00 11 04 FA
01 FA 00 11 04 F0 00 11 04 F1 00 11 02 03 00 11 04 F2 00 11 04 EF 00 11 04
01 10 00 EA 00 75 EA 00 11 04 F2 00 11 04 F2 00 11 01 FA 00 11 01 FA 00 11
00 11 01 FB 00 11 01 F9 00 11 01 F9 00 11 02 03 00 11 04 F0 00 11 04 F2 00
F2 00 11 04 FA 00 11 01 FA 00 11 01 FA 00 11 01 FA 00 11 01 FA 00 11 01 FB
04 F0 00 11 04 F2 00 11 04 F2 00 11 04 F0 00 11 04 F0 00 11 04 F1 00 11 04
11 01 FA 00 11 01 FB 00 11 01 FB 00 11 02 04 00 11 04 F1 00 11 04 F0 00 11
00 11 04 F9 00 11 01 FA 00 11 01 FA 00 11 01 FB 00 11 01 FB 00 11 01 FA 00
01 10 01 5F 00 75 EA 01 FA 00 11 01 FA 00 11 02 03 00 11 04 F1 00 11 04 F2
04 FB 00 11 05 02 00 11 02 03 00 11 02 03 00 11 02 04 00 11 02 04 00 11 02
11 04 FB 00 11 04 FB 00 11 04 FA 00 11 04 F9 00 11 04 F9 00 11 04 FA 00 11
00 11 04 FA 00 11 02 04 00 11 04 F9 00 11 02 0C 00 11 02 02 00 11 02 03 00
03 00 11 05 04 00 11 04 FC 00 11 02 04 00 11 02 03 00 11 02 03 00 11 04 F9
04 F9 00 11 04 FB 00 11 04 FB 00 11 02 03 00 11 02 03 00 11 02 03 00 11 02
01 10 01 D4 00 75 EA 00 11 04 FA 00 11 04 FA 00 11 02 04 00 11 02 03 00 11
00 11 02 03 00 11 04 F9 00 11 04 F8 00 11 02 02 00 11 04 FA 00 11 04 FA 00
04 00 11 04 FB 00 11 02 03 00 11 02 03 00 11 02 0B 00 11 04 F9 00 11 02 04
04 FB 00 11 05 04 00 11 02 03 00 11 02 03 00 11 02 03 00 11 02 03 00 11 02
11 04 F9 00 11 04 FA 00 11 04 FB 00 11 04 FA 00 11 04 F9 00 11 04 F9 00 11
00 11 02 02 00 11 02 03 00 11 02 03 00 11 02 0D 00 11 04 FB 00 11 04 FA 00
01 10 02 49 00 6D DA 04 F8 00 11 04 F8 00 11 04 FA 00 11 04 FB 00 11 04 FA
02 04 00 11 02 03 00 11 02 03 00 11 02 03 00 11 02 0C 00 11 04 FA 00 11 04
11 04 FB 00 11 05 04 00 11 02 03 00 11 02 03 00 11 04 F9 00 11 02 04 00 11
00 11 04 FA 00 11 02 05 00 11 04 FB 00 11 04 FA 00 11 04 F9 00 11 04 F9 00
03 00 11 02 03 00 11 02 04 00 11 02 04 00 11 05 03 00 11 02 02 00 11 02 03
04 F8 00 11 02 08 00 11 00 05
[2]^Power OFF^6
```

Figure 2-2

2-2 Download IR Learning Data to the IR Module

The procedure of downloading IR learning data to the IR module is shown in figure 2-3 follows the description of steps.

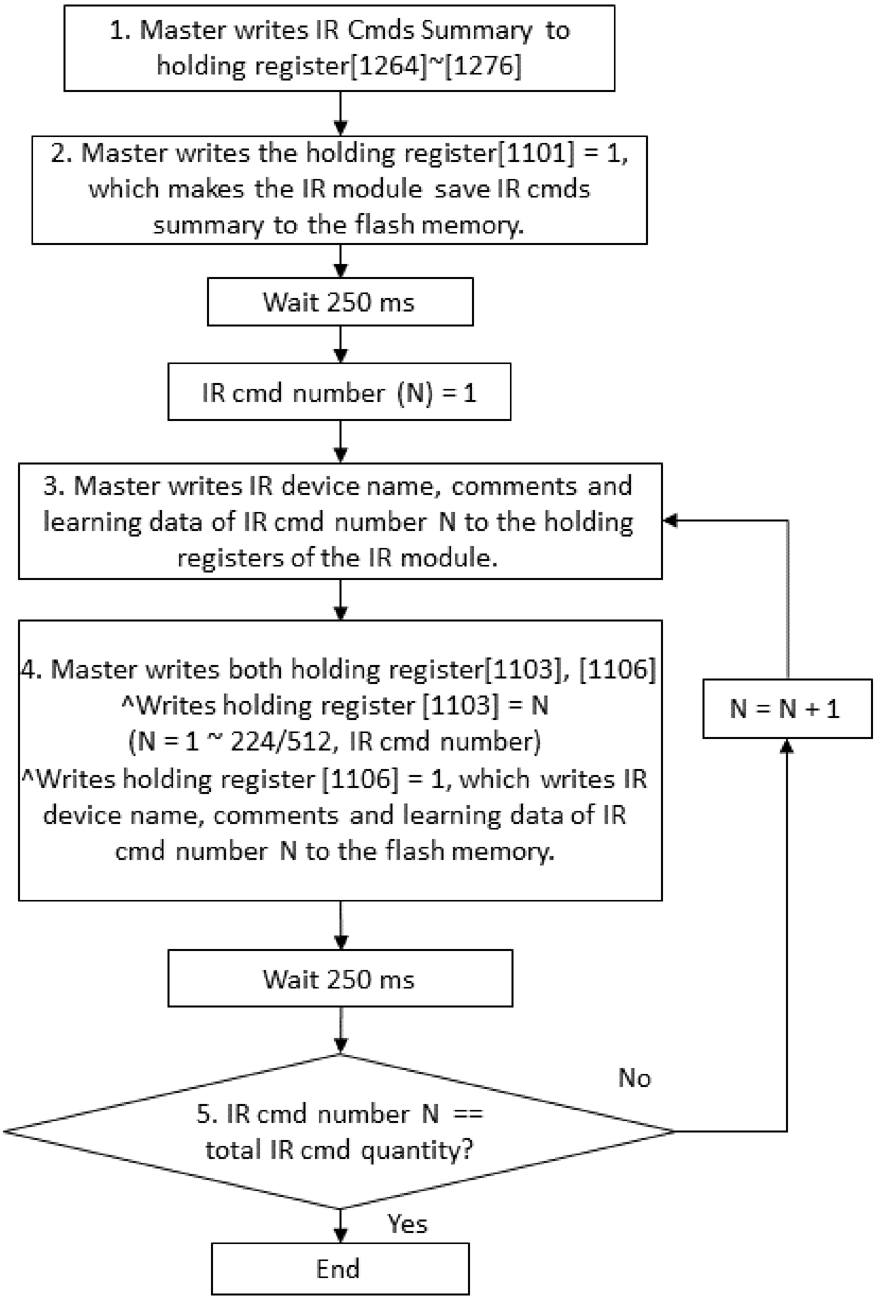

1. Master writes IR Cmds Summary to holding register[1264]~[1276]
2. Master writes the holding register[1101] = 1, which makes the IR module save IR cmds summary to the flash memory.
Wait 250 ms
IR cmd number (N) = 1
3. Master writes IR device name, comments and learning data of IR cmd number N to the holding registers of the IR module.
4. Master writes both holding register[1103], [1106]
^Writes holding register [1103] = N
(N = 1 ~ 224/512, IR cmd number)
^Writes holding register [1106] = 1, which writes IR device name, comments and learning data of IR cmd number N to the flash memory.
Wait 250 ms
5. IR cmd number N == total IR cmd quantity?
No
N = N + 1
Yes
End

Figure 2-3 Master download IR learning data to an IR module.

- **Modbus master download IR commands to the IR module**

Modbus master writes the IR commands to the memory of the IR module sequentially.

Step 1:

Write the IR command summary to the holding register [1264]~[1276]. IR command summary includes IR device quantity, Total IR command quantity, IR command quantity for each IR device.

Step 2:

Modbus master write value 1 to the holding register[1101] (=1). It makes the IR module save the IR command summary to the flash memory.

^**Note**: After writing the holding register[1101], please wait for 250 ms to write data to the flash memory. Please be careful to write this register in programming because there is a limit of one hundred thousand times of writing data to the flash memory.

Step 3:

The maximum storage capacity of the IR module for IR commands is 224(IR-210/IR-712A) or 512(IR-712-MTCP). Master needs to send all IR learning data (should start from IR command number N = 1) to the flash memory of the IR module.

Modbus master should write the IR device names, IR command names(comments) and IR learning data to the corresponding holding registers [1162]~[1212], [1213]~[1263], [0]~[719] by Modbus FC16 requests. Please refer to Table 3-2.

The holding register [0] means the length of an IR command learning data. If the length is larger than 117 words, it needs multiple Modbus commands to transfer the learning data. Refer to the step 6 and 7 in section 1-3.

Step 4:

Modbus master should write values to the holding register [1103] and [1106] simultaneously:

^Modbus master writes an IR command number N to the holding register [1103] = N (N = 1~224/512, IR cmd number).

^Modbus master writes value = 1 to the holding register [1106] = 1 which makes the IR modules write the IR device names, IR command names(comments) and IR learning data in **Step 3** to the flash memory.

^It is suggested to use FC16 to write holding register [1103]~[1106] at once, where the holding register [1104] and [1105] should be zero.

Note: After writing the holding register[1101], please wait for 250 ms to write data to the flash memory. Please be careful to write this register in programming because there is a limit of one hundred thousand times of writing data to the flash memory.

Step 5：

Check if the IR command number (N) is equal to the total IR command quantity. If it is, stop the download procedure. If it is not, go back to **Step 3** to download the next IR command learning data (N = N +1, the IR command number should be continuous). The total IR command quantity in this step should be the same as the total IR command quantity in the “IR command summary” in **Step 1** and **Step 2**.

2-3 Load IR Learning Data from the IR Module

Modbus master can read the IR learning data from an IR module. The procedure is shown in figure 2-4 followed descriptions of steps.

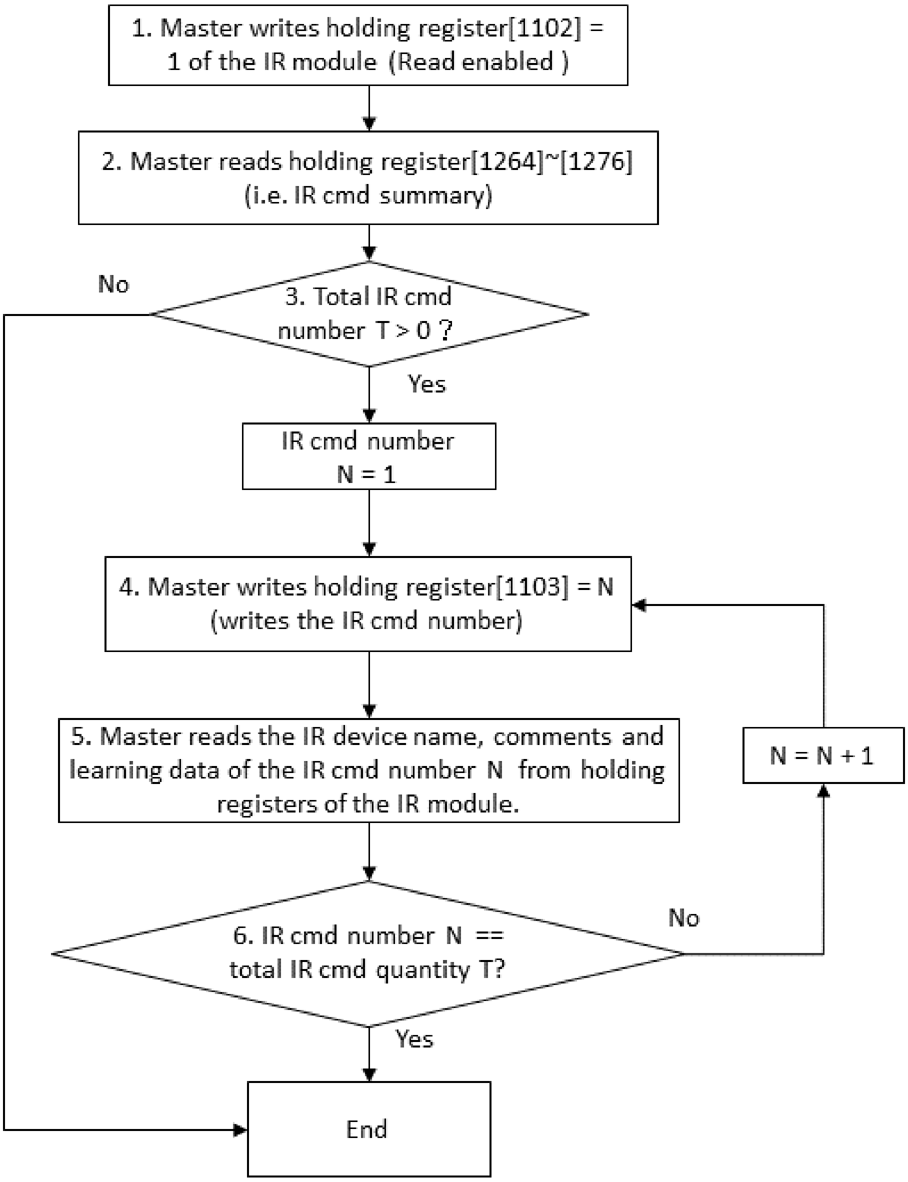

1. Master writes holding register[1102] = 1 of the IR module (Read enabled)
2. Master reads holding register[1264]~[1276] (i.e. IR cmd summary)
No
3. Total IR cmd number T > 0 ?
Yes
IR cmd number N = 1
4. Master writes holding register[1103] = N (writes the IR cmd number)
5. Master reads the IR device name, comments and learning data of the IR cmd number N from holding registers of the IR module.
N = N + 1
No
6. IR cmd number N == total IR cmd quantity T?
Yes
End

Figure 2-4 Master reads IR learning data from an IR module.

- **Modbus master load IR commands from the IR module**

Modbus master reads all IR commands from the IR module sequentially.

Step 1:

Modbus master writes value = 1 to the holding register [1102] = 1 which makes the IR module to read the IR command summary from flash memory and save it in the holding register [1264]~[1276] (refer to Table 3-2).

Step 2:

Modbus master send FC3 request to read back the holding register [1264]~[1276] (IR command summary) which tells the total IR command quantity for reading in the next steps.

Step 3:

Check the total IR command quantity. If it is zero, stop the procedure. If is not zero, go to Step 4.

Step 4:

The IR command number (N) starts from 1.

Modbus master send a FC6 request to write the IR command number(N) to the holding register [1103] = N. IR module will load the relevant learning data (an IR device name, an IR command name and a set of IR command learning data) from flash memory to the holding register [1162]~[1212], [1213]~[1263] and [0]~[719] (refer to Table 3-2).

Step 5:

Modbus master sends FC3 requests to read back the IR device name (holding register [1162]~[1212]), IR command name(holding register [1213]~[1263]) and IR command learning data (holding register [0]~[719]).

The holding register [0] means the total length of a set of IR command learning data. If the length is greater than 117 words, it needs multiple Modbus requests to read back the data as described in **Step 6** and **Step 7** of Section 1-3.

Step 6:

Check if the IR command number (N) is equal to the total IR command quantity. If it is, stop the reading procedure. If it is not, go back to **Step 4** for the next IR command number.

3. Modbus Registers Mapping Table

3-1 Modbus Input Registers (3xxxx)

Use FC4 to **read multiple registers**.
The size of one register(address) is a WORD (2 bytes). (Read Only)

Table 3-1 Input Registers List

Start Address	Value	Description
0 ~ 719 (0x0~0x2CF)		Store the IR learning data after IR learning finished. (720 words) [0]: Length of IR learning data, including address [0] (unit: WORD) [1 ~ 719]: IR learning data of one IR command
734 (0x2DE)		Firmware version (2 bytes), e.g. 0x0102 => ver. 1.02
735~736 (0x2DF~0x2E0)		Firmware date (4 bytes). [735]: year, e.g. 2015(dec) = 0x07DF [736]: month(hi byte) + day(lo byte), e.g. 0x0910 => Sep/16

3-2 Modbus Holding Registers (4xxxx)

Use FC3 to **read multiple registers**.
Use FC6 to **write single register**.
Use FC16 to **write multiple registers**.
The size of one register(address) is one WORD (2 bytes). (Read/Write)

Table 3-2 Holding Registers List

Start Address	Value	Description
0 ~ 719 (0x0~0x2CF)		Store the IR learning data (one IR command) for emitting IR signal later. (720 words) [0]: Length of IR learning data, including address [0] (Length unit: WORD) [1 ~ 719]: IR learning data of one IR command
1100 (0x44C)	0, 1	Flag of Learn On/Off mode (1->On, 0->Off)
1101 (0x44D)	0, 1	Flag for writing "IR cmds summary" from holding registers to Flash. 1 => WRITE data from Modbus AO memory to flash. 0 => do nothing. (Auto change to 0 after writing)
1102 (0x44E)	0, 1	Flag for reading "IR cmds summary" from Flash to holding registers. 1 => read data from flash to Modbus AO memory. 0 => do nothing. (Auto change to 0 after reading)
1103 (0x44F)	1 ~ 224	IR command number. (1~224) For emitting IR signal or accessing learning data to the flash memory.
1104 (0x450)	01h ~ 2Fh	IR output channels. E.g. 0x01->pin1, 0x02->pin2, 0x04->pin3, 0x10->pin5, 0x03->pin1 & pin2
1106 (0x452)	0, 1	Flag for writing an "IR cmd" from holding register to Flash memory. 0=>do nothing (back to 0 after writing); 1=>Write

Table 3-2 Holding Registers List (Cont. 1)

Start Address	Value	Description
1122 (0x462)	1~247	Modbus Net ID
1123 (0x463)	1, 2	・Only for IR-210/IR-712A Current COM port of IR-210/IR-712A: 1->RS-485, 2->RS-232 **(Obsolete for firmware version >= 1.20)**
1124 (0x464)		・Only for IR-210/IR-712A high word of baud rate(bps) , e.g. 115200 == 0x0001C200, where HiWord => 0x0001, LoWord => 0xC200 Supports 9600, 19200, 38400, 57600, 115200 bps.
1125 (0x465)		・Only for IR-210/IR-712A Low word of baud rate(bps) , e.g. 115200 == 0x0001C200, where HiWord => 0x0001, LoWord => 0xC200 Supports 9600, 19200, 38400, 57600, 115200 bps.
1126 (0x466)	0 ~ 2	・Only for IR-210/IR-712A parity, 0 => none, 1 => odd, 2 => even
1127 (0x467)	8	・Only for IR-210/IR-712A databits, only 8 is effective.
1128 (0x468)	0~2	・Only for IR-210/IR-712A stopbits, 0 => none, 1 => 1, 2 => 2
1131 (0x46B)	6 ~ 200	Gap time (ms), default = 72 ms, range: 6 ~ 200 ms
1132 (0x46C)		・For IR-210/IR-712A Modbus command response delay time. unit: ms. (0 ~ 60 ms) ・For IR-712-MTCP Byte0(high byte) & Byte1(low byte) of IPv4 Address (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [192.168.255.1]
1133 (0x46D)		・Only for IR-712-MTCP Byte2(high byte) & Byte3(low byte) of IPv4 Address (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [192.168.255.1]
1134 (0x46E)		・Only for IR-712-MTCP Byte0(high byte) & Byte1(low byte) of Subnet Mask (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [255.255.0.0]
1135 (0x46F)		・Only for IR-712-MTCP Byte2(high byte) & Byte3(low byte) of Subnet Mask (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [255.255.0.0]

Table 3-2 Holding Registers List (Cont. 2)

Start Address	Value	Description
1136 (0x470)		・Only for IR-712-MTCP Byte0(high byte) & Byte1(low byte) of Gateway (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [192.168.0.1]
1137 (0x471)		・Only for IR-712-MTCP Byte2(high byte) & Byte3(low byte) of Gateway (4 bytes) where [Byte0.Byte1.Byte2.Byte3] := [192.168.0.1]
1138 (0x472)		・Only for IR-712-MTCP IP Address Type ^High byte of [1138]: bit 0: DHCP enabled state, 0=>Static IP, 1=>DHCP enabled bit 1~7: reserved, should be zero. ^Low byte of [1138]: Should be zero
1139 (0x473)		Reserved.
1140 (0x474)		・For IR-210/IR-712A Flag for writing basic settings above (from Net ID) to flash memory. Value: 0 => do nothing (Auto change to 0 after writing) 1 => set temporarily 2 => set permanently ・For IR-712-MTCP Reset (Reboot) the module and/or write basic settings to flash memory ^High byte of [1140]: bit 0: Reset the module. Value: 0 => do nothing; 1 => reset bit 1~7: reserved ^Low byte of [1140]: bit 0~2: Write settings to the MCU's flash and reset(reboot) the module. Value: 0 => nothing, 1 => Write & reset. 2 => Only write. No reset. bit 3~7: reserved

Table 3-2 Holding Registers List (Cont. 3)

Start Address	Value	Description
1142 ~ 1143 (0x476 ~ 0x477)	0 ~ 65535	・Only for IR-712-MTCP Idle timeout of a Modbus TCP connection. Unit: minute ^[1142]: Hi Word (reserved). Should be zero. ^[1143]: Lo Word 0: disable; 1~65535: enable timeout (unit: min)
1162 ~ 1212 (0x48A~0x4BC)		Device name of an IR device (max 51 words, the first word stores the word length of the subsequent string)
1213 ~ 1263 (0x4BD~0x4EF)		Comments of an IR command (max 51 words, the first word stores the word length of the subsequent string)
1264 ~ 1276 (0x4F0~0x4FC)		Store IR commands summary information which was read from Flash. Total 13 words [1264]: IR device Quantity (max device num = 11 for IR-210/IR-712-MTCP; = 3 for IR-712A) [1265]: Total number of IR cmds (max. = 224 for IR-210/IR-712A; = 512 for IR-712-MTCP) [1266]: IR cmds quantity of device #1 [1267]: IR cmds quantity of device #2 [….] ….. where [1265 + N]: IR cmds quantity of device #N (N <= 11) (N = 1~11 for IR-210 and IR-712-MTCP) (N = 1~3 for IR-712A)