C-Power Multi-window control communication protocol

SHENZHEN LUMEN ELECTRONICS CO., LTD.

2012.8.4

Version V1.9

Recension Log:

Version	Date	Changes	Executor
V1.0	2009.9.1	The first version	
V1.1	2009.12.28	Increase the protocol of select play stored program and set variable value	
V1.2	2010.2.3	Increase the introduction of network data package	
V1.3	2010.2.24	Increase the introduction of cancel select play stored program	
V1.4	2010.5.20	Add protocol for setting user variable	
V1.5	2010.6.22	Add protocol for setting global display zone	
V1.6	2010.10.9	Add protocol for stopwatch timer	
V1.7	2011.5.14	Add protocol for template box play	
V1.8	2012.6.1	Improvement	
V1.9	2012.8.4	Improvement	

1、Intruction

General agreement of communication

Data packet is used to communicate between the PC and the controller, in order to enhance the reliability of data, expanding capabilities to deal with images and other data.

Communication process：

a).PC send a data packet to the controller；

b).The controller received the data packet, analysis and processing of data packet， and then return a data packet to PC if necessary；

c). PC receives the data packets returned from the control card, and analysis the received data packets to determine whether communication is successful.

Serial setting:

Baud rate: 115200, 57600, 38400, etc. selected by the selected tool.

Format string: "115200, 8, N, 1", you can change the baud rate value 115200 according to what you set to the controller.

Packet data checksum

Communication process using the packet data checksum to check the correctness of data transmission, Checksum calculations we should pay attention： Data checksum is cumulative each byte of data, use the 16bit (2 bytes) unsigned number to represent, so when the data validation is more than 0xFFFF, the checksum, and retain only 16bit value. For example, 0xFFFA + 0x09 = 0x0003.

Font size code

Font size code	Font size in pixels
0	8
1	12
2	16
3	24
4	32
5	40
6	48
7	56

Font style code

Font style code	Font style describe
0	Default font style
1	Font style 1
2	Font style 2
3	Font style 3
4	Font style 4
5	Font style 5
6	Font style 6
7	Font style 7

Text color code

1-byte color value

Max 8 colors. One bit for one basic color.

Bit 0: red color

Bit 1: green color

Bit 2: blue color

Other: not used

Example:

Color value	Color
1	Red
2	Green
3	Yellow
4	Blue
7	White

3-byte color value

RGB color, one byte for one basic color. It can express all kinds of color. Use each one byte to represent red、green、blue.

Byte 1: Red value of the color

Byte 2: Green value of the color

Byte 3: Blue value of the color

Picture effect code

Code	**Picture effect**
0	Center
1	Zoom
2	Stretch
3	tile

Special effect for text and picture

Code	Effect
0	Draw

1	Open from left
2	Open from right
3	Open from center(Horizontal)
4	Open from center(Vertical)
5	Shutter(vertical)
6	Move to left
7	Move to right
8	Move up
9	Move down
10	Scroll up
11	Scroll to left
12	Scroll to right
13	Flicker
14	Continuous scroll to left
15	Continuous scroll to right
16	Shutter(horizontal)
17	Clockwise open out
18	Anticlockwise open out
19	Windmill
20	Windmill （anticlockwise)
21	Rectangle forth
22	Rectangle entad
23	Quadrangle forth
24	Quadrangle endtad
25	Circle forth
26	Circle endtad
27	Open out from left up corner
28	Open out from right up corner
29	Open out from left bottom corner

30	Open out from right bottom corner
31	Bevel open out
32	AntiBevel open out
33	Enter into from left up corner
34	Enter into from right up corner
35	Enter into from left bottom corner
36	Enter into from lower right corner
37	Bevel enter into
38	AntiBevel enter into
39	Horizontal zebra crossing
40	Vertical zebra crossing
41	Mosaic(big)
42	Mosaic(small)
43	Radiation(up)
44	Radiation(downwards)
45	Amass
46	Drop
47	Combination(Horizontal)
48	Combination(Vertical)
49	Backout
50	Screwing in
51	Chessboard(horizontal)
52	Chessboard(vertical)
53	Continuous scroll up
54	Continuous scroll down
55	Reserved

56	Reserved
57	Gradual bigger(up)
58	Gradual smaller(down)
59	Reserved
60	Gradual bigger(vertical)
61	Flicker(horizontal)
62	Flicker(vertical)
63	Snow
64	Scroll down
65	Scroll from left to right
66	Open out from top to bottom
67	Sector expand
68	Reserved
69	Zebra crossing (horizontal)
70	Zebra crossing (Vertical)

When the random effect is expressed by one byte , the value is 255(0xFF) , when it is expressed by two bytes , the value is 32768(0x8000).

Clock format and display content

Clock format：

Represent by one byte：

bit 0: Signal timing(0: 12signal timing；1: 24 signal timing)

bit 1: Year by bit(0: 4 bit；1: 2 bit)

bit 2: Line folding(0: single-row； 1: multi-row)
bit 3~5: Reserved(set to 0)
bit 6: Show time scale ”Hour scale、Minute scale”
bit 7: Reserved(set to 0)

Clock display content：

Represent by one byte：

Ascertain the display content by bit:
bit 7: pointer
bit 6: week
bit 5: second
bit 4: minute
bit 3: hour
bit 2: day
bit 1: month
bit 0: year

Simple picture data format

Data composition：

Data head	Red data(optional)	Green data(optional)	Blue data(optional)

Data head description：

	0	1	2	3	4	5	6	7
0x00	identify		width		height		property	Reserved

Description：

Data name	Data size(byte)	Description
Identify	2	Set to “I1”。
Width	2	The width of the picture, low byte previous(little endian)
Height	2	The height of the picture,low byte previous(little endian)
Property	1	The gray-scale and color of the picture Bit0: Whether exist red data, exist when 1. Bit1: Whether exist green data, exist when 1. Bit2: Whether exist blue data, exist when 1. Bit3: Reserved, set to 0. Bit4~7: Gray-scale, only support 0 and 7 now.

		0: 2 levels gray, Each lattic data use 1 bit. 7: 256 levels gray, Each lattic data use 8 bit. Each line of the picture data is aligned by byte. As for 2 levels gray picture, when the line data is not enough for 8 bit, add 0.
Reserved	1	Set 0

Data description：

The color of the data is order by red、green、blue. If the identify bit of the property is 0, the color data is not exist.

For one color data, order by ” left to right, top to bottom”. First put the data to the first line, then second line and so on.

Formatted text data format

Rich3 Text

Each character is 3 bytes, the specific meaning of each byte are as follows:

Byte no	**Byte data**
1	Said the color and font size: 4bits (1-7) represent the color (red green yellow blue purple cyan white), low 4bits (= 0 indicates that 8-point text; = 2 16-point text; = 3 24-point text; = 4 32point text; = 5 40-point text; = 6 48-point text; = 7 56-point text).
2	High byte of the text encoding. For single-byte characters, the value is 0.
3	Low byte of the text encoding. For single-byte characters, the value of its ASCII code.

2、Data packet format

2.1 RS232/RS485 data packet format：

2.1.1 The data packet format of RS232/RS485 sending：

0xa5 0x68 0x32 ID 7B FF LL LH PO TP CC 。。。。。。SH SL 0xae

Data	**Value**	**Length(Byte)**	**Description**
Start code	0xa5	1	The start of a packet
Packet type	0x68	1	Recognition of this type of packet
Card type	0x32	1	Fixed Type Code
Card ID	0x01~0xFE 0XFF	1	Control card ID, the screen No, valid values are as follows: 1 ~ 254: the specified card ID 0XFF: that group address, unconditionally receiving data
Protocol code	0X7B	1	Recognition of this type of potocol
Additional information/ confirmation mark	FF	1	The meaning of bytes in the packet is sent, "Additional Information", is a packet plus instructions, and now only use the lowest: bit 0: whether to return a confirmation, 1 to return and 0 not to return bit1 ~ bi7: reserved, set to 0
Packed data length LL LH	0x0000˜0xffff	2	Two bytes, the length of the "CC ……" part content . Lower byte in the former
Packet number PO	0x00~0x255	1	When the packet sequence number is equal to when the last packet sequence number, indicating that this is the last one package.
Last packet number TP	0x00~0x255	1	The total number of packages minus 1.
Packet data	CC ……	Variable-length	Command sub-code and data
Packet data checksum SH SL	0x0000~0xffff	2	Two bytes, checksum。 Lower byte in the former。 The sum of each byte from " Packet type " to " Packet data" content
End code	0xae	1	The end of a packet（Package tail）

2.1.2 The data packet format of the control card returned to RS232/RS485 sender：

0xa5 0x68 0x32 ID 7B FF LL LH PO TP CC 。。。。。。 SH SL 0xae

Data	**Value**	**Length(Byte)**	**Description**
Start code	0xa5	1	The start of a packet
Packet type	**0xE8**	1	Recognition of this type of packet

			0xE8 = (0x68 | 0x80), for the app 3.2 or below return 0x68, app 3.3 or above return 0xe8 to same as other protocol(such as “set time” protocol), so you can ignore the highest bit (0x80), then it works for all app version
Card type	0x32	1	Fixed Type Code
Card ID	0x01~0xFE 0XFF	1	Control card ID, the screen No, valid values are as follows: 1 ~ 254: the specified card ID 0XFF: that group address, unconditionally receiving data
Protocol code	0X7B	1	Recognition of this type of protocol.
Return value	RR	1	RR = 0x00: that successful; RR = 0x01 ~ 0xFF: that the failure error code. (0x01: checksum error) (0x02: packet sequence error) (Other: to be confirmed) In addition, a certain period of time does not receive the returned data packet, said communication failures.
Packed data length LL LH	0x0000˜0xffff	2	Two bytes, the length of the "CC 。。。。。。" part content . Lower byte in the former
Packet number PO	0x00~0x255	1	When the packet sequence number is equal to when the last packet sequence number, indicating that this is the last one package.
Last packet number TP	0x00~0x255	1	The total number of packages minus 1.
Packet data	CC 。。。。。。	Variable-length	Command sub-code and data
Packet data checksum SH SL	0x0000~0xffff	2	Two bytes, checksum 。 Lower byte in the former。 The sum of each byte from " Packet type " to “ Packet data” content 。
End code	0xae	1	The end of a packet（Package tail）

"Packet number", "Last packet number" in the return package was re-calculated by the number of packets returned。

2.1.3 RS232/RS485 packet data transcoding description：

The following process is done sending and receiving low-level functions, If you write your own PC side of the sending and receiving programs, you must implementation as below conventions. Use the without transcoding code to calculation checksum.

Send：

Between start code and end code, if there is 0xA5, 0xAA or 0xAE, it should be converted to two code：

0xa5 ➔ 0xaa 0x05。 The purpose is to avoid the same with the start character 0xa5

0xae ➔ 0xaa 0x0e。 The purpose is to avoid the same with the end of the symbol 0xae。

0xaa ➔ 0xaa 0x0a。 The purpose is to avoid the same with the escape character 0xaa。

Receive：

Received symbol 0xa5, said that the beginning of a packet

Received symbol 0xae, said that the end of a packet

When PC receive data from controller, if there is 0xA5, 0xAA or 0xAE, it should convert two code to one code, specifically for

0xaa 0x05 ➔ 0xa5

0xaa 0x0e ➔ 0xae

0xaa 0x0a ➔ 0xaa

2.2 Network data packet format

2.2.1 The data packet format of network sending

Data	**Value**	**Length(Byte)**	**Description**
ID Code	0x00000000 ~ 0xffffffff	4	Control card ID, high byte in the former。 Need to set to the same value on the card.
Network data length	0x0000 ~ 0xffff	2	The byte length that from "Packet type" to "Packet data checksum".
Reservation	0x0000	2	Reservations. Fill in 0
Packet type	0x68	1	Recognition of this type of packet
Card type	0x32	1	Fixed Type Code
Card ID	0x01~0xFE 0XFF	1	Control card ID, the screen No, valid values are as follows: 1 ~ 254: the specified card ID 0XFF: that group address, unconditionally receiving data
Protocol code	0X7B	1	Recognition of this type of protocol.
Additional information/ confirmation mark	FF	1	The meaning of bytes in the packet is sent, "Additional Information", is a packet plus instructions, and now only use the lowest: bit 0: whether to return a confirmation, 1 to

			return and 0 not to return bit1 ~ bi7: reserved, set to 0
Packed data length LL LH	0x0000~0xffff	2	Two bytes, the length of the "CC 。。。。。。" part content . Lower byte in the former
Packet number PO	0x00~0x255	1	When the packet sequence number is equal to when the last packet sequence number, indicating that this is the last one package.
Last packet number TP	0x00~0x255	1	The total number of packages minus 1.
Packet data	CC 。。。。。。	Variable-length	Command sub-code and data
Packet data checksum SH SL	0x0000~0xffff	2	Two bytes, checksum。 Lower byte in the former。 The sum of each byte from " Packet type " to " Packet data" content 。

The network packet data does not need to do transcoding processing.

2.2.2 The data packet format of the control card returned to network sender

Data	Value	Length(Byte)	Description
ID Code	0x00000000 ~ 0xffffffff	4	Control card ID, high byte in the former。 Need to set to the same value on the card.
Network data length	0x0000 ~ 0xffff	2	The byte length that from "Packet type" to "Packet data checksum".
Reservation	0x0000	2	Reservations. Fill in 0
Packet type	0xE8	1	Recognition of this type of packet 0xE8 = (0x68 | 0x80), for the app 3.2 or below return 0x68, app 3.3 or above return 0xe8 to same as other protocol(such as "set time" protocol), so you can ignore the highest bit (0x80), then it works for all app version
Card type	0x32	1	Fixed Type Code
Card ID	0x01~0xFE 0XFF	1	Control card ID, the screen No, valid values are as follows: 1 ~ 254: the specified card ID 0XFF: that group address, unconditionally receiving data
Protocol code	0X7B	1	Recognition of this type of protocol.

Return value	RR	1	RR = 0x00: that successful; RR = 0x01 ~ 0xFF: that the failure error code. (0x01: checksum error) (0x02: packet sequence error) (Other: to be confirmed) In addition, a certain period of time does not receive the returned data packet, said communication failures.
Packed data length LL LH	0x0000~0xffff	2	Two bytes, the length of the "CC 。。。。。。" part content . Lower byte in the former
Packet number PO	0x00~0x255	1	When the packet sequence number is equal to when the last packet sequence number, indicating that this is the last one package.
Last packet number TP	0x00~0x255	1	The total number of packages minus 1.
Packet data	CC 。。。。。。	Variable-length	Command sub-code and data
Packet data checksum SH SL	0x0000~0xffff	2	Two bytes, checksum 。 Lower byte in the former。 The sum of each byte from " Packet type " to " Packet data" content 。

"Packet number", "Last packet number" in the return package was re-calculated by the number of packets returned。

The network packet data does not need to do transcoding processing.。

2.3 Command sub-code and data：CC。。。。。。

CC： A sub-byte instruction code, specifying the meaning of the data。

。。。。。。**：** Data content for different sub-code instructions, there are different elements。

If the data needs to be divided into several packages, command sub-code only in the first data packet appears, the other only contains the data content of data packets。

2.2.1 Command sub-code includes：

General protocol command

Commad sub-code(CC)	**meanings**
0x01	Division of display window (area)
0x02	To send text data to a specified window

0x03	To send image data to the specified window
0x04	Static text data sent to the specified window
0x05	To send clock data to the specified window
0x06	Exit show to return to play within the program
0x07	Save / clear the data
0x08	Select play stored program (single-byte)
0x09	Select play t stored program (double-byte)
0x0a	Set variable value
0x0b	Select play single stored program, and set the variable value
0x0c	Set global display area
0x0d	Push user variable data
0x0e	Set timer control
0x0f	Set the global display area and variable values
0x12	Send pure text to the specified window

Program template command

Commad sub-code(CC)	**meanings**
0x81	Set program template command
0x82	In or out program template command
0x83	Query program template command
0x84	Delete program command
0x85	Send text to special window
0x86	Send picture to special window
0x87	Clock / temperature display in the specified window of the specified program
0x88	Send alone program
0x89	
0x8a	Set program property
0x8b	Set play plan
0x8c	Delete play plan
0x8d	Query play plan

2.2.2 The specific definition of command sub-code and data：

Division of display window：CC=0x01:

Data Items	Value	Length(byte)	Description
CC	0x01	1	Note This command is divided into display window (area)
Window Number	0x01~0x08	1	The window should be divided into the number of valid value 1 ~ 8.
Window 1 X-coordinate XH XL	0x0000~0xffff	2	Window x-coordinate, high byte in the former
Window 1 Y-coordinate YH YL	0x0000~0xffff	2	Window y-coordinate, high byte in the former
The width of the window 1 WH WL	0x0000~0xffff	2	The width of the window, high byte in the former
The height of the window 1 HH HL	0x0000~0xffff	2	The height of the window, high byte in the former
……			
Window N X-coordinate XH XL	0x0000~0xffff	2	Window x-coordinate, high byte in the former
Window N Y-coordinate YH YL	0x0000~0xffff	2	Window y-coordinate, high byte in the former
The width of the window N WH WL	0x0000~0xffff	2	The width of the window, high byte in the former
The height of the window N HH HL	0x0000~0xffff	2	The height of the window, high byte in the former

- Based on the above definition, requires 8 bytes for each window's location and size, then divided into N windows, data on a total of 2 +8 * N bytes

Send text data to a specified window：CC=0x02:

Data Items	Value	Length(byte)	Description
CC	0x02	1	Description This is a text data packet
Window No	0x00~0x07	1	The window sequence number, valid values 0 ~ 7.
Mode	1	1	Refer to Special effect for text and picture
Alignment	0～2	1	0: Left-aligned 1: Horizontal center 2: Right-aligned
Speed	1～100	1	The smaller the value, the faster
Stay time	0x0000~0xffff	2	High byte in the former. Unit: second.
String		Variable-length	Every 3 bytes to represent a character. Refer to Rich3 text of Formatted text data format.

Send image data to a specified window：CC=0x03:

Data Items	Value	Length(byte)	Description
CC	0x03	1	Description This is a image data packet
Window No	0x00~0x07	1	The window sequence number, valid values 0 ~ 7.
Mode	0x00	1	0x00: Draw
Speed	1	1	The smaller the value, the faster。Now appears that this value is invalid
Stay time	0x0000~0xffff	2	High in the former. Units of seconds.
Image Data Format	0x01	1	0x01: gif image file format 0x02: gif image file references. 0x03: picture package picture reference. 0x04: simple image format.
Image Display X Position	0x0000~0xffff	2	Began to show the location of X coordinate. Relative upper-left corner the window.
Image Display Y Position	0x0000~0xffff	2	Began to show the location of Y coordinate. Relative upper-left corner the window.
Image Data		Variable-length	According to "image data format" is defined to determine the meaning of the data. Image data format is 0x01: gif image file of

			the actual data, which contains the image width, height and other information; Image data format is 0x02: the gif image file name stored in the control card. Image data format is 0x03: The image package file name and image number that stored in the controller. The middle separated by spaces. For example, "images.rpk 1" Image data format is 0x04： Simple picture data, see the description format.

Send static text：CC=0x04:

Data Items	**Value**	**Length(byte)**	**Description**
CC	0x04	1	Description of the data packet is static text
Window NO	0x00~0x07	1	Window sequence number, valid values 0 to 7
Data type	1	1	0x01: Simple text data
The level of alignment	0~2	1	0: left Alignment 1: center Alignment 2: right Alignment
Display area X	0x0000~0xffff	2	The X coordinate of upper left corner of the display area。Upper left corner of the window relative
Display area Y	0x0000~0xffff	2	The Y coordinate of upper left corner of the display area。Upper left corner of the window relative
Display area width	0x0000~0xffff	2	The width of display area。High byte in the former.
Display area height	0x0000~0xffff	2	The height of display area。High byte in the former.
Font		1	Bit0~3: font size Bit4~6: font style Bit7: Reserved
Text color R	0~255	1	The red color component
Text color G	0~255	1	The green color component
Text color B	0~255	1	The blue color component
Text		Variable-length	Text string to the end of 0x00.

Send clock：CC=0x05:

Data Items	**Value**	**Length(byte)**	**Description**
CC	0x05	1	Description of the data packet is clock
Window NO	0x00~0x07	1	Window sequence number, valid values 0 to 7
Stay time		2	Stay time in second。High byte in the former
Calendar		1	0: Gregorian calendar date and time 1: Lunar date and time 2: Chinese lunar solar terms 3: Lunar time and date + Solar Terms
Format		1	Format: Format bit 0: when the system (0: 12 hour; 1: 24 hours system) bit 1: Year digit (0: 4; 1: 2) bit 2: Branch (0: single; 1: multi-line) bit 3 ~ 7: reserved (set to 0)
Content		1	By bit to determine the content to display. bit 7: Pointer bit 6: weeks bit 5: seconds bit 4: minute bit 3: hour bit 2: day bit 1: month bit 0: year
Font		1	Bit0~3: font size
Text color R	0~255	1	The red color component
Text color G	0~255	1	The green color component
Text color B	0~255	1	The blue color component
Text		Variable-length	Text string to the end of 0x00.

Exit show and return to play within the program：CC=0x06:

Data Items	**Value**	**Length(byte)**	**Description**
CC	0x06	1	Play programs stored on the card

Save/clear the data：CC=0x07:

Data Items	Value	Length(byte)	Description
CC	0x07	1	The data packet is a request control card to save data in the window
Save/clear	0x00/0x01	1	0x00: save data to flash. 0x01: Clear flash data
Reserve	0x00 0x00	2	Reserved for later expansion

Select play stored program (single-byte)：CC=0x08

Data Items	Value	Length(byte)	Description
CC	0x08	1	Description of the data packet is stored program data select play(single-byte)
Options		1	Bit0: Whether to save select play message to flash. 0 not to save，1 save。 Bit1~7: Reserved, set to 0
The number of programs	1~255 or 0	1	The program number that to be selected to play, if the number is 0,the controller will exit the select play state.
The program number table	1~255	Variable-length	The program number list,1 byte for each program. Exceed the number of programs stored program number is ignored

Select play stored program (double-byte)：CC=0x09

Data Items	Value	Length(byte)	Description
CC	0x09	1	Description of the data packet is stored program data select play(double-byte)
Options		1	Bit0: Whether to save select play message to flash. 0 not to save，1 save。 Bit1~7: Reserved, set to 0
The number of programs	1~512 or 0	2	The program number that to be selected to play, the max value is 512, high byte in the former. if the number is 0,the controller will exit the select play state.
The program number table	1~65535	Variable-length	The program number list,2 bytes for each program. Exceed the number of programs

			stored program number is ignored

Set variable value：CC=0x0a

Data Items	**Value**	**Length(byte)**	**Description**
CC	0x0a	1	Description of the data packet is the data set variable value
Options		1	Bit0: Whether to save all variable value to flash, 0 not to save，1 save。 Bit1: Whether to clear all variable value before save, 0 not to clear，1 clear。 Bit1~7: Reserved, set to 0
Variable number and allow cross-variable zone	1~100	1	Bit0~6：The variable number Bit7：Whether to allow cross-variable zone setting. 0 is not permitted; 1 is permit Corresponds to a variable number of each variable area size of each variable region is 32 bytes. Multiple continuous variables can be linked to a variable area used, occupied area of the variable number of variables can not be used。 When does not allow cross-variable area, more than 32 bytes of data are discarded；When allow cross-variable area, calculate the length of the data area to use the number of variables.
Variable data length table	n（0～255）	Variable-length	Specified by the order of bytes of data for each variable. The length of variable number and data is (1 + n) bytes.
Variable number and data		Variable-length	The first byte is a variable number, followed by a specified length of variable data

Note：

Valid values for the variable number is 1~100。Number of variables corresponding to each variable area can store 32 bytes of data, a number of continuous variable area can be used together for a variable, the variable area occupied number of variables can not be used。

When variable values are not updated and just save the variable value to the FLASH, it can set the " Variable number **" of the value of 0, set the "** Options **" to save**

Select play single stored program and set the variable value：

CC=0x0b

Data Items	Value	Length(byte)	Description
CC	0x0b	1	Description of the data packet is the data select single program and set variable value
Options		1	Bit0: Whether to save program number to flash 0 not to save，1save。 Bit1: Whether to save all variable value to flash , 0 not to save，1save。 Bit2: Whether to clear all variable value before save, 0 not to clear，1 clear。 Bit3~7: Reserved, set to 0
Program numbers	1~65535	Variable-length	The program number list,2 bytes for each program. Exceed the number of programs stored program number is ignored
Variable number and allow cross-variable zone	1~100	1	Bit0~6：The variable number Bit7：Whether to allow cross-variable zone setting. 0 is not permitted; 1 is permit Corresponds to a variable number of each variable area size of each variable region is 32 bytes. Multiple continuous variables can be linked to a variable area used, occupied area of the variable number of variables can not be used。 When does not allow cross-variable area, more than 32 bytes of data are discarded；When allow cross-variable area, calculate the length of the data area to use the number of variables.
Variable data length table	n（0～255）	Variable-length	Specified by the order of bytes of data for each variable. The length of variable number and data is (1 + n) bytes.
Variable number and data		Variable-length	The first byte is a variable number, followed by a specified length of variable data

Note：

Valid values for the variable number is 1~100。Number of variables corresponding to each variable area can store 32 bytes of data, a number of continuous variable area can be used together for a variable, the variable area occupied number of variables can not be used。

When variable values are not updated and just save the variable value to the FLASH, it can set the " Variable number **" of the value of 0, set the "** Options **" to save。**

Set global display area：CC=0x0c

Data Items	**Value**	**Length(byte)**	**Description**
CC	0x0c	1	Describe the packet is the data which to set the global display zone.
Options		1	Bit0: Whether to save the setup to FLASH 0 not to save，1 save。 Bit1~7: Reserved, set value to 0
ZoneArea count	1~8	1	The count of global display zone which is to be set. Cancel all the zone when zone count is 0
Synchronization		1	Bit0: Synchronous display. 0 not synchronous, 1 synchronous. Bit1~7: Reserved
Retention		2	Set value 0
Zone Definition		Zone Count*16	The specific definition of global display zone

Area Definition：（16 bytes each item）

The first byte is zone type, available zone types:

Vale	**Type**
1	Display the variable’s specify text
2	Display the variable’s specify file (.gif)
6	Display hint text of other zone
7	Display stopwatch timer value
other	Reserved

Type = 1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x00	Type	reserved	x		y		cx		cy		start	end	stay		font	align

Explanation：

Data Name	**Data Size (byte)**	**Description**
Type	1	1：Display the variable’s specify text
x	2	Zone start point X. High byte previous
y	2	Zone start point Y. High byte previous
cx	2	Zone width. High byte previous

cy	2	Zone Height. High byte previous
Start	1	Start variable number, valid value 1～100
End	1	End variable number, valid value 1～100
stay	2	The stay time when display each valid variable's content, the unit is second. High byte previous
font	1	Font size and color Bit0~2: font size(8,12,16,24,32,40,48,56) Bit3: Color invert Bit4: Red value of color Bit5: Green value of color Bit6: Blue value of color Bit7: Reserved
align	1	Text alignment

All "reserved" values need to be set to 0

Type = 2

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x00	Type	reserved	x		y		cx		cy		start	end	stay		mode	reserved

Explanation：

Data Name	Data Size (byte)	Description
Type	1	2: Display the variable's specify file (.gif)
x	2	Zone start point X. High byte previous
y	2	Zone start point Y. High byte previous
cx	2	Zone width. High byte previous
cy	2	Zone Height. High byte previous
Start	1	Start variable number, valid value 1～100
End	1	End variable number, valid value 1～100
stay	2	The stay time when display each valid variable's content, the unit is second. High byte previous
mode	1	Image draw mode: 0: left top
Reserved	1	

All "reserved" values need to be set to 0

Type = 7

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x00	Type	reserved	x		y		cx		cy		font	format	reserved			

Explanation：

Data Name	Data Size (byte)	Description
Type	1	7: Display stopwatch timer value
x	2	Zone start point X. High byte previous
y	2	Zone start point Y. High byte previous
cx	2	Zone width. High byte previous
cy	2	Zone Height. High byte previous
font	1	Font size and color Bit0~2: font size(8,12,16,24,32,40,48,56) Bit3: Color invert Bit4: Red value of color Bit5: Green value of color Bit6: Blue value of color Bit7: Reserved
format	1	0: “mm:ss” 1: “mm:ss:nn”
Reserved	4	

All “reserved” values need to be set to 0

Push and set the variable value：CC=0x0d

Data Item	Value	Lenght(byte)	Description
CC	0x0d	1	Describe the package is the data which push and set the variable value
Options		1	Bit0: Whether to save all the variable value to FLASH. 0 not save，1 save. Bit1: direction。0 push back,1 push forward Bit 2~3: retention，set to 0 Bit4~7: push count. +1 means the variable count of push
Variable area count		1	Bit0~6: The variable count for push 1~100 Bit7: Retention, set 0
Variable data length		1	Specify variable data’s byte. The total length of variable number and data is (n+1) byte
Variable number and data		changeable	The first byte is variable number, follow by specify length variable data.

Set Timer：CC=0x0e

Data Item	Value	Lenght(byte)	Description
CC	0x0e	1	Describe the package is the data of set stopwatch
Select Timer		1	Select the Timer by bite. Bit value 1 means the Timer valid Bit0: Timer 1. Bit1: Timer 2. Bit3: Timer 3. Bit4: Timer 4. Bit5: Timer 5. Bit6: Timer 6. Bit7: Timer 7.
Action		1	1：Initialize Timer 2：Reset Timer 3：Startup Timer 4：Pause Timer 5：Save the setup of Timer Other：Retention
property		1	Have different value according to the Action. Check the detail information in the below table.
Value		4	Have different value according to the Action. Check the detail information in the below table.

The description of all Actions and the correspondence property and value

Action	Description	Property	Value
Initialize Timer		Bit0: 0 Time，1 Countdown Bit1: 0 Pause，1 start immediately Bit2~3: retention Bit4~7: Time count	High byte previous. The initialization value of countdown, measure time by millisecond. The value retention when time, set to 0
Reset Timer		Bit0: 0 Use old value，1 Use new value Bit1: 0 Pause，1 start immediately Bit2~3: Retention	High byte previous. Countdown: Use as a new initialization value when the property is set to use new value. Ignore when the property is set to use the old value. Time: Retention, set 0.
Start Timer		Reserved, set 0	Retention, set 0

Pause Timer		Retention, set 0	Retention, set 0
Save the setup of Timer		Retention, set 0	Retention, set 0

Set the global display area and variable values: CC=0x0f

By the command，set the global display area and variable values

Data Item	Value	Lenght(byte)	Description
CC	0x0f	1	Describe the package is the data which to set the global display area and variable values
Effective control		2	Play times: High byte first。 The value of 0 has been effective. Bit15: reserved, fill in 0 Bit0 to 14: play times
Reserved		2	Reserved, fill in 0
Area option		1	Bit0: Whether to save the setting to the flash , 0 : Don’t save，1 : Save。 Bit1~3: Reserved, fill in 0 Bit4: Whether to clear the others defined global display area Bit5 ~7: Reserved, fill in 0
Area number	1～8	1	To set the number of global zone
Area no	1～8	Area number	Bit0~3: Specified in the global area no. Valid values are 1 to 8. If the current no have used, then overwrite the original area information. Bit 4~7: Reserved, fill in 0
Area definition		16 x Area number	Specific definition of the global display area. Specific definition see CC = 0x0c (that set the global display area).
Variable Options		1	Bit0: Whether to save all variable value to flash, 0 not to save，1 save。 Bit1: Whether to clear all variable value before save, 0 not to clear，1 clear。 Bit1~7: Reserved, set to 0
Variable number and allow cross-variable zone		1	Bit0~6：The variable number Bit7：Whether to allow cross-variable zone setting. 0 is not permitted; 1 is permit Corresponds to a variable number of each variable area size of each variable region is 32 bytes. Multiple continuous variables can be

			linked to a variable area used, occupied area of the variable number of variables can not be used。 When does not allow cross-variable area, more than 32 bytes of data are discarded；When allow cross-variable area, calculate the length of the data area to use the number of variables.
Variable data length table	n（0～255）	Variable-length	Specified by the order of bytes of data for each variable. The length of variable number and data is (1 + n) bytes.
Variable number and data		Variable-length	The first byte is a variable number, followed by a specified length of variable data

Note：After use this command，The global area automatically becomes synchronized display.

Send pure text to specified window：CC=0x12

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x12	1	Send pure text to the specified window
Window no	0x00~0x07	1	The window sequence number, valid values 0 ~ 7.
Effect	1	1	See: Special effect for text and picture
Alignment	0～2	1	Bit0~1: the horizontal alignment (0: Left-aligned 1: Horizontal center 2: Right- aligned) Bit2~3: vertical alignment (0:Top-aligned 1:Vertically center, 2: Bottom- aligned) Bit4~6: Reserved, set to 0 Bit7: Reserved, set to 0
Speed	1～100	1	The smaller the value, the faster
Stay time	0x0000~0xffff	2	High byte in the former. Units of seconds.
Font		1	Bit0~3: font size,see： Font size code Bit4~6 :font style,see :Font style code Bit7 : reserved (set to 0)
Text color R	0~255	1	The red color component
Text color G	0~255	1	The green color component
Text color B	0~255	1	The blue color component

Text		Variable-length	Text string to the end of 0x00.

Example 1：A5 68 32 01 7B 01 0f 00 00 00 12 00 00 00 00 00 03 02 ff 00 00 61 62 63 00 62 03 AE

Example 2：A5 68 32 01 7B 01 13 00 00 00 12 00 00 01 00 00 03 02 ff 00 00 61 62 63 0d 63 62 61 00 9a 04 AE

2.2.3 Detail of program template command code and data

The program template agreement is a set of relatively independent of the agreement, the basic concept of this agreement are as follows:

"Program" is a standalone player within a certain time, a message is displayed on the screen can be divided into multiple regions, each region can be specified individually.

The maximum number of programs is 100, corresponding to an effective program number from 1 to 100.Send program information, new information covering the same program number.

Set program template：CC=0x81:

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x81	1	Describe the package is the data which to set program template.
Color and gray		1	Bit0: Red mark Bit1: Green mark Bit2: Blue mark Bit3: Reserved Bit4～6: Gray level 0: 2 level gray，7: 256 level gray Bit7: Reserved
Screen width		2	High byte first。
Screen height		2	High byte first。
Window number		1	The display window number, the maximum number is 10
Options		1	Bit0: Forced into the program template run Bit1: Save the template position. 0: user disk, 1: system disk. If the template is saved to the system tray, the original template of the user tray is cleared; if the template is saved to the user's disk, the original template of the system disk is cleared。 Bit2~7: Reserved
Default parameter	Stay time/ Scroll times	2	Stay time/Scroll times: High byte first。When the show effect is scroll , it means scroll times(0 scroll one times,1 scroll

			two times,…) , for others , it means stay time, unit is second.
	Speed	1	The smaller the faster
	Font size	1	Bit 0~3:Font size, Font size code Bit 4~6:Font style , Font style code
	Font color.	1	Font color.。 Text color code
	Show effect	1	Show effect。Special effect for text and picture
	Picture mode	1	Picture mode。 Picture effect code
	Clock Format	1	Clock Format。 Clock format and display content
	Clock content	1	Clock content。 Clock format and display content
	Text alignment	1	Text alignment and line space Bit0~1: the horizontal alignment (0: Left-aligned 1: Horizontal center 2: Right- aligned) Bit2~3: vertical alignment (0:Top-aligned 1:Vertically center, 2: Bottom- aligned) Bit4~7: Line space 0~15point
	Reserved	6	Reserved， fill in 0
Window parameter		Variable-length	window parameter. Each window has a 16 bytes length parameter. The total length of the data is: the number of the window*16. You can see the detail at Appendix 1 : Window position and property

Enter into or exit program template mode：CC=0x82:

Data Item	Value	Lenght(byte)	Description
CC	0x82	1	Describe the package is the data which to enter into or exit program template mode
In or out	1 / 0	1	Bit0: Mode Action。 1: Enter into program template mode 0: Exit program template mode Bit1~2: Reserved Bit4: Save mode。 1 Save， 0 Not save。 Bit5~7: Reserved

* The meaning of "return value" in the return packet:

0x01 program template is invalid, can not into the program template.

Query program template status parameter ：CC=0x83:

Send data:

Data Item	Value	Lenght(byte)	Description
CC	0x83	1	Describe the package is the data of query program template status parameter.
Options	0x00	1	Bit0: Whether to query program template status parameter Bit1:Whether to return the template definition color gray, screen size information Bit2~7: Reserved

Return data

Data Item	Value	Lenght(byte)	Description
CC	0x83	1	Describe the package is the return data of query program template status parameter.
Options		1	The same value with send value of "Options".
Template mode		1	0: Not program template 1: program template
Template status		1	Bit0~1: template availability 0: the template is not available 1: the template can be used others: Reserved Bit2~7: Reserved

Delete program：CC=0x84:

Data Item	Value	Lenght(byte)	Description
CC	0x84	1	Describe the package is the data which to delete program
Options		1	Bit0: Delete program option 0：Delete all program 1：Delete specified program others: Reserved
Program number		1	Delete all programs do not need this data
Program list		Variable-length	Length number of bytes equal to the number of programs. Each program is 1 byte, program number from

			1

* The meaning of "return value" in the return packet:

0x01 the template is not available

0x80 Current is not a program template way

Send text to the specified window of the specified program：

CC=0x85:

Send data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x85	1	Describe the package is the data which to send text to the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program No		1	Valid value:1~100
Window No		1	Valid value:1～10，Invalid when out of program template definition.
Property		1	Bit0~3: Text type 0：Common Text ; 1：Format Text Bit4: Display format. 0: default format 1:specify format Bit5~7: Reserved
Show format (**Note**： When the “Property " display format is zero, do not need this data)	Stay time/ Scroll times	2	Stay time/Scroll times: High byte first。 When the show effect is scroll，it means scroll times(0 scroll one times,1 scroll two times,…)，for others，it means stay time, unit is second.
	Speed	1	The smaller the faster
	Font size	1	Bit 0~3:Font size, Font size code Bit 4~6:Font style , Font style code
	Font color.	1	Font color.。 Text color code
	Show effect	1	Show effect。Special effect for text and picture
	Text alignment	1	Text alignment and line space Bit0~1: the horizontal alignment (0: Left-aligned 1: Horizontal center 2: Right- aligned) Bit2~3: vertical alignment (0:Top-aligned 1:Vertically center,

			2: Bottom- aligned) Bit4~7: Line space 0~15point
	Reserved	1	Reserved， fill in 0
Text		Variable-length	Text data according to different types of text, the text type to see the definition of the " Property ". Common Text :The text string, the end to 0x00 Format text: The first byte is 0x01, the followed Rich3 text, a detailed description see Formatted text data format

Return data

Data Item	Value	Lenght(byte)	Description
CC	0x83	1	Describe the package is the return data of send text to the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program no		1	The same value with send value “Program no”. Valid value:1~100
Window No		1	The same value with send value “Window no”. Valid value:1~10,Invalid when out of program template definition.
Packet loss number		1	The number of packets that have not yet received. Sends the first packet loss number is the total number of packets minus one.
The packet number of the packet loss		Variable-length	Packet loss packet number. Always in accordance with small to large; the first packet packet number is 0. Each package a byte.

*Must first send the first packet. Best to confirm the first packet sent successfully, and then send subsequent packets.

* The meaning of "return value" in the return packet:

0x01 program template is invalid

0x11 program number is out of range

0x12 window number out of range

0x80 currently is not program template way

Send picture to the specified window of the specified program：CC=0x86:

Send data

Data Item	Value	Lenght(byte)	Description

CC	0x86	1	Describe the package is the data which to send picture to the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program No		1	Valid value:1~100
Window No		1	Valid value:1 ～ 10 , Invalid when out of program template definition.
Picture type		1	Bit0~3: Picture type 1： gif image file format 2： gif image file references 4 ： simple image format ， see Simple picture data format Bit4: Display format. 0: default format 1:specify format Bit5: Do you play immediately. (1:Now Playing) Bit6~7: Reserved
Show format (**Note**： When the “Property " display format is zero, do not need this data)	Stay time/ Scroll times	2	Stay time/Scroll times: High byte first。 When the show effect is scroll , it means scroll times(0 scroll one times,1 scroll two times,…) , for others , it means stay time, unit is second.
	Speed	1	The smaller the faster
	Show effect	1	Show effect。Special effect for text and picture
	Picture mode	1	Picture mode。 Picture effect code
	Reserved	3	Reserved，fill in 0
Picture data		Variable-length	Picture data

Return data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x83	1	Describe the package is the return data of send picture to the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program No		1	The same value with send value “Program no”. Valid value:1~100
Window No		1	The same value with send value “Window no”. Valid value:1~10,Invalid when out of program template definition.
Packet loss number		1	The number of packets that have not yet received. Sends the first packet loss number is the total number of packets minus one.
The packet		Variable-length	Packet loss packet number. Always in

number of the packet loss			accordance with small to large; the first packet packet number is 0. Each package a byte.

*Must first send the first packet. Best to confirm the first packet sent successfully, and then send subsequent packets.

* The meaning of "return value" in the return packet:

0x01 program template is invalid

0x11 program number is out of range

0x12 window number out of range

0x80 currently is not program template way

Show clock/temperature in the specified window of the specified program：CC=0x87:

Send data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x87	1	Describe the package is the data which to show clock/temperature in the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program No		1	Valid value:1~100
Window No		1	Valid value:1 ～ 10 , Invalid when out of program template definition.
Type		1	Bit0~3: Type 2：Clock ; 3：Temperature Bit4: Display format. 0: default format 1:specify format Bit5~7: Reserved, fill in 0

Format		Variable-length	The meaning of the attribute data according to different types
			Type=2：Data see appendix 1 Clock/Calendar type
			Type=3：Data see appendix 1 Temperature and Humidity type

Return data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x87	1	Describe the package is the return data which to show clock/temperature in the specified window of the specified program
Append code		4	The user’s append code, high byte previous.
Program No		1	The same value with send value “Program no”.

			Valid value:1~100
Window No		1	The same value with send value “Window no”. Valid value:1~10,Invalid when out of program template definition.
Packet loss number		1	The number of packets that have not yet received. Sends the first packet loss number is the total number of packets minus one.
The packet number of the packet loss		Variable-length	Packet loss packet number. Always in accordance with small to large; the first packet packet number is 0. Each package a byte.

Send alone program：CC=0x88:

Send data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x88	1	Describe the package is the data which to send alone program Alone program is independent of the program template, it can split windows.
Append code		4	The user’s append code, high byte previous.
Program No		1	Valid value:1~100
Option		1	Bit0~4: Reserved, fill in 0. Bit5: Do you play immediately.(1 Now Playing) Bit6~7: Reserved, fill in 0.
Reserved		3	Reserved, fill in 0.
Window number		1	Valid value 1～10
Window information table		22* Window number	Every window information table has a 22 bytes length parameter. The 1~16 bytes are window position and property, You can see the detail at Appendix 1： Window position and property; The 17~19 bytes are window data offset; The 20~22 bytes are window data length. High byte first. If no data ,then window data offset and window data length all are 0. The total length of the data is: the number of the window*22.
Window data		Variable-length	Window play data: “Text” 、 “Picture” …

			Byte 1：Data Type(1 Text；4 Picture) Byte 2：Data Format （Like “Text type” in command 0x85 and “Picture type” in command 0x86） Byte 3：Text data or picture data。

Return data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x88	1	Describe the package is the return data which to send alone program
Append code		4	The user’s append code, high byte previous.
Program No		1	Valid value:1~100
Reserved		1	Reserved, fill in 0.
Packet loss number		1	The number of packets that have not yet received. Sends the first packet loss number is the total number of packets minus one.
The packet number of the packet loss		Variable-length	Packet loss packet number. Always in accordance with small to large; the first packet packet number is 0. Each package a byte.

*Must first send the first packet. Best to confirm the first packet sent successfully, and then send subsequent packets.

* The meaning of "return value" in the return packet:

0x01 program template is invalid
0x11 program number is out of range
0x12 window number out of range
0x13 The definition of the window outside the screen size of the program template definition
0x80 currently is not program template way

Query program information：CC=0x89:

Send data:

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x89	1	Describe the package is the data which to query program info
Info flag		1	Special which program info will to be query 1: Query valid programs count and program number 2: Query specifies program information. Other: Reserved
parameters		5	The meaning of the parameters according to the info flag different.

Parameter and return data description

- **Query“valid program count and program number”**

Parameter

Byte 1～5	Reserved, fill 0

Return data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x89	1	Describe the package is the return data packet of query program info
Info flag		1	Same with send value ”info flag”
parameters		5	Same with send value ”parameters”
Valid program count		1	Valid program count
Valid program number		Variable-length	Each byte identifies an effective program。Valid value 1～100。

* The meaning of "return value" in the return packet:

0x01 Controller not running in program template mode

0x10 Unknown info flag

- **Query specifies program information**

Parameter

Byte 1	Program number
Byte 2～5	Reserved, fill 0

Return data

Data Item	**Value**	**Lenght(byte)**	**Description**
CC	0x89	1	Describe the package is the return data packet of query program info
Info flag		1	Same with send value ”info flag”
parameters		5	Same with send value ”parameters”
Information count		1	Now only return one information
Program number		1	Program number
User append code		4	User append code

* The meaning of "return value" in the return packet:

0x01 Controller not running in program template mode

0x10 Unknown info flag

0x11 Invalid programs

0x12 Can’t get program information

Set program property：CC=0x8a:

Data Item	**Value**	**Lenght(byte)**	**Description**

CC	0x8a	1	Describe the package is the data which to set program property
Option		1	Bit0: Set the range of the program property 0: All programs 1: Specify program Other: Reserved
Program number		0/1	The count of the program. When it sets all program property , don't need this data.
Program list		Variable-length	The list of the program. When it sets all program property , don't need this data. Length number of bytes equal to the number of programs. Each program is 1 byte, program number from 1
Property flag 1		1	marked which property you want to set by byte, set 0 if the data not exist. Bit0: The level of the program. Bit1: The cycle count. Bit2: Valid time. How long will the program be valid from now on. Bit3: Interval time Bit4~7: Reserved
Property flag 2		1	Bit0~4: valid time. >0 the count of the valid time.<=4 Bit5~7: Reserved
Program level		1	The program level. 1～3 level, The high level of the program is priority.
Play loop count		2	Loop count, High byte previous(big-endian). 0: Do not play the program, use to shield program temporarily. 1~255: The loop count of the program.
Valid time		2	High byte previous (big-endian). In minute. 0: Not limit play time >0: Specify play time in minute.
Expiration date		12	Validity start date: "Year Month Day Hour Minute Second" The expiry date: "Year Month Day Hour Minute Second" "Year Month Day Hour Minute Second" each

			one byte
Effective period		6* period number	Period start: "Hour Minute Second" Period end: "Hour Minute Second" "Hour Minute Second" each one byte

The meaning of "return value" in the return packet:

0x01 program template is invalid

0x80 currently is not program template way

Set play plan：CC=0x8b:

Data Item	Value	Lenght(byte)	Description
CC	0x8b	1	Describe the package is the data which to set play plan
Append code		4	The user's append code, high byte previous.
Plan No		1	Valid value 1~100。Total support 100 plans, For each plan No, the new data cover the old data
Format and level		1	Bit0~3: Data format，fill in 0x01 Bit4~7: Indicates the priority level. The priority level the greater the value, the more priority to play, 0 is the lowest priority.。
Weekday		1	Bit0~6: 7-bit logo Sunday to Saturday
Begin date		3	3 bytes: Byte1:Year,Valid value0~99,means 2000~2999 Byte2:Month Byte3:Day
End date		3	3 bytes: Byte1:Year,Valid value0~99,means 2000~2999 Byte2:Month Byte3:Day
Begin time		3	3 bytes: Byte1:Hour Byte2:Minute Byte3:Second
End time		3	3 bytes: Byte1:Hour Byte2:Minute Byte3:Second
Program number		1	Valid value:1～100
Program No		Variable-length	Each byte represents a program. Numbered in

			ascending order, do not repeat....。

Total support 100 plans, For each plan No, the new data cover the old data .

Return data

Data Item	Value	Lenght(byte)	Description
CC	0x8b	1	Describe the package is the return data which to set play plan
Append code		4	The user’s append code, high byte previous.
Plan No		1	Valid value 1~100。Total support 100 plans, For each plan No, the new data cover the old data

The meaning of "return value" in the return packet:

0x01 program template is invalid

0x80 currently is not program template way

Delete play plan：CC=0x8c:

Data Item	Value	Lenght(byte)	Description
CC	0x8c	1	Describe the package is the data which to delete play plan
Append code		4	The user’s append code, high byte previous.
Play plan number		1	The number of play plan will to be delete。0 means delete all play plans.
Play plan No		Variable-length	Valid value 1~100。Each byte represents a play plan no。 When delete all play plan, the length of this data is one , value is 0xff.

Total support 100 plans。

When delete all play plans, the play plan number fill in 0, the length of play plan no is one , value is 0xff.

Return data

Data Item	Value	Lenght(byte)	Description
CC	0x8c	1	Describe the package is the return data which to delete play plan
Append code		4	The user’s append code, high byte previous.
Plan No		1	Valid value 1~100。Total support 100 plans, For each plan No, the new data cover the old data

The meaning of "return value" in the return packet:

0x01 program template is invalid

0x11 The number of play plan will to be delete is 0.

0x80 currently is not program template way

Query play plan：CC=0x8d:

Send data

Data Item	Value	Lenght(byte)	Description
CC	0x8d	1	Describe the package is the data which to query play plan
Append code		4	The user’s append code, high byte previous.
Query type		1	0: Query all valid play plan. 1: Query specified play plan no Other: Reserved
Play plan No		1	Valid value:1~100。 When query type is 0，this data fill in 0。

Total support 100 plans。

Return data

Data Item	Value	Lenght(byte)	Description
CC	0x8d	1	Describe the package is the return data which to query play plan
Append code		4	The user’s append code, high byte previous.
Query type		1	0: Query all valid play plan. 1: Query specified play plan no Other: Reserved
Count/Number		1	When query type is 0，this value is valid play schedule count When query type is 1， this value is play schedule number.
Play schedule number table/ play schedule content		Variable-length	When query type is 0，this value is valid play schedule number table When query type is 1， this value is play schedule content. Data format like command 0x8B.

You must deal with the return data according to the different query type.

The meaning of "return value" in the return packet:

0x01 program template is invalid

0x11 Don’t support the query type.

0x12 Invalid play plan no.

0x80 currently is not program template way

Appendix 1 : Window position and property

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x00	x		y		cx		cy		Window property							

Description：

Data Item	Data Size(BYTE)	Description
x	2	Window start point x, high byte previous (big endian).
y	2	Window start point Y, high byte previous (big endian).
cx	2	Window width. High byte previous (big endian).
cy	2	Window height. High byte previous (big endian).
Window property	8	Window default type and parameters.

Window property is represent by 8 bytes：

	0	1	2	3	4	5	6	7
0x00	mode	Parameter						

The definition of the window mode：

Mode value	Description
0	Blank(Show nothing)
1	Text
2	Clock、calendar
3	Temperature、Humidity
4	Picture、Reference of the picture
Other	Reserved

The parameter has different values according to the mode. There are all the mode’s parameters. All reserved position should be set 0x00.

Blank type

	0	1	2	3	4	5	6	7

0x00	0	Reserved

Text type

	0	1	2	3	4	5	6	7
0x00	1	Effect	Font size	Font color	Speed	Stay time		Align

Data Item	Value	Length(BYTE)	Description
Effect	1	1	Effect: See in Special effect for text and picture
Font size		1	Bit0~2: Font size, see in Font size code Bit 4~6:Font style , Font style code
Font color		1	Bit0~2: Font color , see 1-byte color value in Text color code
Speed	0～9	1	The smaller the faster
Stay time/ Scroll times	0x0000~0xffff	2	Stay time/Scroll times: High byte first。 When the show effect is scroll , it means scroll times(0 scroll one times,1 scroll two times,…) , for others , it means stay time, unit is second.
Align		1	Text alignment and line space Bit0~1: the horizontal alignment (0: Left-aligned 1: Horizontal center 2: Right- aligned) Bit2~3: vertical alignment (0:Top-aligned 1:Vertically center, 2: Bottom- aligned) Bit4~7: Line space 0~15point

Clock/Calendar type

	0	1	2	3	4	5	6	7
0x00	2	Font size	Font color	Stay time		calendar	Format	Content

Data Item	Value	Length(BYTE)	Description
Font size		1	Bit0~2: Font size, see in Font size code Bit 4~6:Font style , Font style code
Font color		1	Bit0~2: Font color , see 1-byte color value in Text color code
Stay time	0x0000~0xffff	2	High byte previous(big endian), unit is second.
Calendar		1	0: The gregorian calendar
Format		1	Clock format: See in Clock format and display content
Content		1	Clock content: see in Clock format and display content

Temperature and Humidity type

	0	1	2	3	4	5	6	7
0x00	3	Font size	Font color	Stay time		Format	Reserved	

Data Item	Value	Length(BYTE)	Description
Font size		1	Bit0~2: Font size, see in Font size code Bit 4~6:Font style , Font style code
Font color			Bit0~2: Font color , see 1-byte color value in Text color code
Stay time	0x0000~0xffff	2	High byte previous(big endian), unit is second.
Format		1	0: Celsius 1: Fahrenheit 2: Humidity

Picture and reference to the picture

	0	1	2	3	4	5	6	7
0x00	4	Effect	Speed	Stay time		Reserved		

Data Item	**Value**	**Length(BYTE)**	**Description**
Effect		1	Effect: See in Special effect for text and picture
Speed	0～9	1	The smaller of the value, the faster. Invalid when display immediately.
Stay time	0x0000~0xffff	2	Stay time/Scroll times: High byte first 。When the show effect is scroll , it means scroll times(0 scroll one times,1 scroll two times,…) , for others , it means stay time, unit is second.