BX-Y series program format Instructions

2019-2-18

Copyright

Version history:

Version	Date	Autor	Description
1.0	2018-5-15	Huang	The initial version
	2018-6-13	Huang	1. 整特技列表 Add Stunt list 2. 单行字幕分区的特技类型描述添加推荐类型 3. 公告区特技类型描述添加推荐类型 4. 更新炫动背景分区 update animation background area
	2018-6-14	Huang	更新炫彩文本分区 update colorful words
	2018-7-9	Huang	整理传感器分区 add sensor area
	2018-7-19	Huang	更新炫彩文本分区部分参数 upate colorful word area parameter
	2018-7-26	Huang	扩展计时分区 Extended timing area
	2018-8-16	Huang	修改传感器分区 correction 参数 update sensor area correction parameter
	2018-9-12	Huang	1. 视频分区增加 'clone'参数，支持分区克隆 2. 视频分区增加'startTime'参数，支持设定起始播放点(暂未实现) 3. 表盘、时间、计时、农历、传感器和炫动背景分区添加'duration' 参数 4. 新增背景音乐分区 5. 新增多彩边框分区 6. 列表文件新增'priority'，'loop'，'integrate'字段；新增'dates'，'times'子标签；增加行为说明 7. 表盘分区指针纹理化
	2019-2-18	Huang	1. 整理数据库分区 2. 节目文件中增加节目背景色参数：'bgColor' 3. 勘误
	2019-4-17	Huang	新增复合传感器各子传感器地址及功能序号

DIRECTORY

BX-Y series program format Instructions

1. Fast development guiding

This manual describes the program formats supported by the Onbon BX-Y series asynchronous full color controller, which is used to guide the program edit if using LedshowYQ, iLedcloud platform and third-party software ..

2. User working directory structure

The following is the root directory structure for internal users. The directories and files that can be used in users' program files are relative to the root directory. **Do not delete the private directory at the root.**

```
└────/
     ├──── backup/
     ├──── bulletins/
     ├──── download/
     ├──── dynamics_save/
     ├──── fonts/
     ├──── lists/
     ├──── log/
     ├──── nginx_upload/
     ├──── programs/
     ├──── share/
     ├──── update/
     ├──── upload/
     └──── user/
```

- ✓ backup: Backup directory
- ✓ bulletins: bulletins directory
- ✓ download: download directory
- ✓ dynamics_save: dynamics_save directory
- ✓ fonts: fonts directory
- ✓ lists: Special directory for saving playlist
- ✓ log: Special directory for saving log files
- ✓ nginx_upload: The user uploads the temporary transit directory of the file through HTTP protocol, which makes it effective after uploading the function
- ✓ programs: Special directory for saving program files
- ✓ share: Special directory for saving program material files
- ✓ update: Dedicated directory for system upgrade and maintenance
- ✓ user: User self maintenance directory

3. Program format and examples

3.1 Summary

There are three types of programs: Normal programs , dynamic programs and bulletin programs.

Normal program is the main program form of asynchronous controllers. It is a partitioned program form described by playlist file and program file in XML format: all partitions in the same program file are uniformly scheduled; partitions described by different program files are played in turns according to the organization of playlist file.

Dynamic program is a kind of special program form which is not saved (usually) after power failure. The dynamic area can be used as a global program to schedule independently from the normal program, or it can be bound with the normal program to participate in the scheduling of the normal program.

The bulletin area program is actually a special area. At the front of the screen, the user-defined text information is displayed, which is independent of ordinary programs and dynamic area programs, and is scheduled separately.

Without special instructions, all color related fields are given as 0xaarrggbb string format.

All XML files are encoded in UTF-8.

3.2 Normal programs

Normal programs are composed of playlist files and program files in XML format: the playlist file determines the controller type and screen size of the program, and sequence and play time of each program; the program file contains the area and contents of the program.

3.2.1 Playlist file

Label	attribute	Type	Instruction
<list>		-	Play list
√	deviceType	int	Controller type,used to specify the controller type used in the list
√	screenWidth	int	Screen width
√	screenHeight	int	Screen height
<program>		-	Sub tag of <list>, program description Maximum 64 (Can be extend)
√	order	int	Order play : from program 1 to program N
√	playMode	string /int	Play mode 'Timer'/0 - Timer play 'Counter'/1 - Counter play
–	priority	Int	Programs priority：（new） 1-16, priority, 1 is the highest, 16 is the lowest (default value) .In a certain period of time, priority is given to high-level programs. Any values not in the range of 1-16 are treated as 16 the- lowest priority.
–	loop	int	Rotation times (new) : 0 - always participate in rotation;1~1000 - times of rotation.Default0.
√	programFile	string	Relative path of program file
–	playTime	int	Play time :valid when use time play mode, unit: second, default 5 seconds
–	playCount	int	Default1 Play times , valid when use count time play mode ,default 1
–	integrate	string/int	Continued play (new) 'yes' / 1 - continuously play the specified number of times of playtime ; 'No' / 0 - discontinuous play: after playing once, switch to other programs with less than the same priority. Valid when playing for a certain number of times, default: 'yes' / 1.
–	startDate	string	Start data of the play aging Format：'yyyy-MM-dd'， default：'1970-01-01'
–	startTime	string	Start time from the day. Format：'hh:mm:ss'， Default：'00:00:00'
–	stopDate	string	Stop date of the play aging. Format：'yyyy-MM-dd'， Default：'2099-12-31'
–	stopTime	string	Stop time of the day Format：'hh:mm:ss'， Default：'23:59:59'
–	weekFlag	int	Week Flag 1, 2, 4, 8, 16, 32, 64 represent Monday to Sunday respectively. .Add the numbers corresponding to the date you want to play to get the result; default 127. Eg: '1' means broadcast only on Monday, '17' means broadcast on Monday and Friday, '127' means broadcast from Monday to Sunday

–	dates	string	： Play date segment (New): Basic format: '[start date] [end date]' The start / end date is given in the form of 'yyyy MM DD', separated by spaces in the middle; Maximum 8 groups are allowed, each group is separated by commas; if it is empty, press as' 1970-01-01 2099-12-31 '. Eg: "August 1, 2018 August 4,2018-08-06, 2018-0809" The date defined by all tags of this type also needs to be processed to be valid for weekflag.
–	times	string	Playback period (New): Basic format: '[start time] [end time]' The start / end date is given in the form of 'HH: mm: ss', separated by spaces in the middle;Maximum 8 groups are allowed, each group is separated by commas; if it is empty, press as' 00:00:00 24:00:00' Eg '07:30:00 09:00:00,11:00:00 13:30:00'

✓ Example：

```
<?xml version="1.0" encoding="utf-8"?>
  <list deviceType="9048" screenWidth="1280" screenHeight="1024">
    <program order="0" playMode="1" playTime="1"
      startDate="" stopDate="" startTime="" stopTime="" weekFlg="127"
      programFile="programs/1c761cf57e689b7012de839a9776025e.xml" />
    <program order="1" playMode="1" playTime="1"
      startDate="" stopDate="" startTime="" stopTime="" weekFlag="127"
      programFile="programs/0f3b1ab03f2c622a3a3242cecb8972c6.xml" />
  </list>
```

OR NEW EXAMPLE：

```
<?xml version="1.0" encoding="utf-8"?>
  <list deviceType="9048" screenWidth="1280" screenHeight="1024">
    <program order="0" playMode="1" playTime="1" priority="1"
      loop="5" dates="" times="08:00:00 11:30:00" weekFlg="127"
      programFile="programs/1c761cf57e689b7012de839a9776025e.xml" />
    <program order="1" playMode="1" playTime="1" priority="2"
      loop="5" dates="" times="08:00:00 11:30:00" weekFlg="63"
      programFile="programs/0f3b1ab03f2c622a3a3242cecb8972c6.xml" />
  </list>
```

3.2.1.1.1 Behavior description

3.2.1.1.2 Time playing programs are processed for once. The duration set by 'playtime' is counted as one time. If the set duration is different from the actual duration of the program itself, the set duration shall be taken as the standard. If the actual duration of the program is less than the set duration, the program will be played repeatedly (if the remaining duration is insufficient, the program will be played at the end); otherwise, the program will be cut off and only the previous part will be played

3.2.1.1.3 When all programs of the same priority complete the broadcasting time limit, it is determined whether to count again according to their respective 'loop'

3.2.1.1.4 when the loop is not 0, The screen might black. To avoid this, users need to design the playlist correctly: for example, Users can design a full-time effective background program with the lowest priority.

3.2.1.1.5 New added attribute < times > and the original < starttime > / < stopTime > can appear at the same time, but < times > is priority. It is recommended that the upper computer decide which attributes to distribute according to different versions (versions after v18082700, excluding v18082700). When there is intersection or convergence in different periods, it shall be merged automatically.

3.2.1.1.6 Different versions (versions after v18082700, excluding v18082700) has different distributor, decide which attributes to distribute. In case of intersection or convergence of different date segments, they shall be merged automatically.

3.2.1.1.7 ram> tag<times>/<dates> describes a limited number of date segments/time periods. To achieve more date segments/time periods for the same program, you can add several <program> tags with different date segments/time periods definitions in the list, and their 'programFile' points to the same program file.However, you must avoid overlapping <times> and <dates> in different <program> tags.

3.2.1.1.8

when a program with low priority is interrupted by a program with high priority, it shall be deemed that the program has not started processing.

3.2.1.1.9 。when there is operation related to locking program (locking program instruction or remote control switching program), time effective control and priority control will be invalid.

3.2.2 Program file

✓ Program file lable and specification

Label	attribute	Type	Instruction
<program>		-	program description
√	name	string	Program name (custom, maximum 64 bytes) It is suggested to use the method of "program" + number to name the program
–	bgColor	string	Background color . Default '0x00000000' - transparent black

✓ Eg：

```
<?xml version="1.0" encoding="utf-8"?>
<program name="program_0" bgColor="0xFFFF8000">
   <picturepanel x="0" y="0" w="347" h="236" zOrder="0"
     transparency="100" pictureType="0" lastPicMoveWidth="1">
       <picUnit file="/share/2310a45b2885be33c1b4e07882f56b84.jpg"
       fileType="jpg" stuntType="1" stuntSpeed="16" stayTime="1"
       order="0"/>
       <picUnit file="/share/120a1224dd21116d1f128642e194d0ac.bmp"
       fileType="bmp" stuntType="1" stuntSpeed="16" stayTime="1"
       order="1"/>
   </picturepanel>
</program>
```

The total areas in a program cannot be exceed 128.

3.2.3 Image area

Label	attribute	Type	Instruction
<picturepanel>或<picture>		-	Image area
√	xCoord/x	int	area upper left corner X coordinate
√	yCoord/y	int	area upper left corner Y coordinate
√	width/w	int	area width
√	height/h	int	area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100
<picUnit>		√	<picture> subtag, picture subunit definition The maximum allowed is tentatively 128, playing in order
√	order	int	Play order 0-127
√	file	string	Image file path
√	fileType	string	Image format, gif 'means GIF, others are non GIF, default is non gif
√	stuntType	int	Index of stunt type
√	stuntSpeed	int	Stunt speed level :1~16， 1 is the fatest
√	stayTime	int	Image stay time , unit :seconds

3.2.4 Single line subtitle partition

This partition is shown as a single line when 'unitType' is in 'text' (control card self rendering) mode.For multi-line display, use 'image' (host rendering) mode.

Label	attribute	Type	Instruction
<textpanel>Or <text>		-	Single-line subtitle partition
√	xCoord/x	int	area upper left corner X coordinate
√	yCoord/y	int	area upper left corner y coordinate
√	width/w	int	area width
√	height/h	int	area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100
√	stuntType	int	Partition special effects type index Only 0,50 to 57 are recommended
√	unitType	string/int	Subunit type selection: 'image' /0 - renders a good text image and the upper computer renders it 'text' /1 - unrendered text, control card self rendering
<imageUnit>		-	<text> subtag, valid when subunit type is' image ' The maximum number is allowed to play in order
√	order	int	Play order 0-127
√	file	string	Image file path
√	stuntSpeed	int	Stunt speed level :1~16， 1 is the fatest
√	stayTime	int	Image stay time , unit :seconds
-	lastPicMoveWidth	int	The movement width/height of the last image of the whole subtitle When multiple pictures are not the last, the value must be 0 The entire caption has only one image, which is fixed to the window width/height
<textUnit>		-	The sub label of < text > is valid when the sub unit type is' text '.The maximum number is allowed to be temporarily 128 and played in order
√	order	int	Play order : 0~127
√	content	string	content
√	stuntSpeed	int	Stunt speed level :1~16， 1 is the fatest
√	stayTime	int	Image stay time , unit :seconds
-	bgColor	string	Background color , Default ‘0x00000000’- transparent black
-	fontColor	string	Font color , Default ‘0xFFFFFFFF’ -opaque white
√	fontName	string	font name Font name
√	fontSize	int	font sizeFont size
-	fontSizeType	string/int	Font size unit (default is "pixel"): ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
-	fontAttributes	string/int	Font additional attributes: It includes five types, such as' bold ',' Italic ',' normal ','

			underline 'and' strikeout '. It can be combined by' & ', for example:' bold & Italic & underline '
-	fontAlignment	string	pending

3.2.5 Video area

Label	attribute	Type	Instruction
<videopanel>或<video>		-	Video area
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100
-	clone	string	Coordinates, width and height of clone area Given in the format of 'X: y [: W: H]', there is no '[: W: H]' for using area width and height; at most 3 clones can be cloned (the extra ones will not be processed), separated by , Eg：‘100:200,200:200:200:160,100:400:200:160’
√	videoType	string/int	Video type: ‘local’/0 - Local video or web streaming ‘capture’/1 - External input video (Y series not support currently)
√	volumeMode	string/int	Mute: 'Unmute' /0 - not mute;'.mute / 1 - Mute
√	rotationMode	int	Counter clockwise rotation angle: Only 0 / 90 / 180 / 270, default0
√	scaleMode	string/int	Zoom mode ‘original’/0 - Original scale (Y Series not suppprt currently) ‘window’/1 - 窗口比例 window scale
<videoUnit>		√	Sub label of < video >, definition of video sub unit The maximum allowed number is 128 tentatively, playing in order
	order	int	Play order: 0~127
√	file	string	Video file path or streaming URL The video file path is given as a relative path, for example: "Share/abcdtest. Mp4" The streaming media URL is given in the form of a valid streaming media link, for example: "Rtmp://live.hkstv.hk.lxdns.com/live/hks" The tested streaming protocols are: RTMP, RTSP, HLS
√	source	string/int	Video file path or streaming URL The video file path is given as a relative path, for example:“share/abcdtest.mp4” The streaming URL is given in the form of an effective streaming link, for example: “rtmp://live.hkstv.hk.lxdns.com/live/hks” The tested streaming media protocols are: RTMP, RTSP, HLS External input video type: 'CVBS' / 0 or' HDMI '/ 1 Valid when videotype is' capture '(not supported for Y

			Series)
-	startTime	int	Play start time unit: seconds: Valid when playing video file, default 0; cannot be longer than the video file play time
√	playTime	int	Playback time unit: seconds: 0 indicates that the program will be played all the time or determined by the length of the file
√	volume	int	Video sound volume：0~100

3.2.6 Dial area

Label	attribute	Type	Instruction
<clockpanel >或<clock>		-	Dial area
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100
-	duration	int	Play time: In seconds, Default5 seconds;The Default duration for this partition is defined, and the actual duration of the program is the maximum duration for all partitions in the program
√	bgImage	string	background image path
-	timeEquation	string	adjust time， format‘hh:mm:ss’， Default‘00:00:00’
-	positiveTE	bool	adjust time direction： ‘True’ is +， ‘False’ is -， Default‘True’
-	hourColor	int	hour color， Default‘0xFFFFFF00’ - yellow
-	minuteColor	int	Minute color， Default‘0xFF00FF00’ - green
-	secondColor	int	second color ,no default value
<hour>		–	<clock> subtag, defining the hour hand If this tag exists, it automatically ignores' hourColor '
-	image	string	Clockwise image file path Is empty, means the control card draws a simple straight line
-	color	int	hour color： iColor of hour hand: Default '0xFFFFFF00' - yellow
–	length/l	int	The length of the hour hand, temporarily unsupported, is temporarily fixed at 30% of the smaller partition width and height
√	width/w	int	width of the partition or the height of the smaller one
<minute>		–	<clock> subtag, defining the minute hand With this tag, 'minuteColor' is automatically ignored
-	image	string	Minute pin image file path Is empty, means the control card draws a simple straight line
-	color	int	Minute color： image is empty will use， Default‘0xFF00FF00’ - green
–	length/l	int	The length of the minute hand, temporarily unsupported, is temporarily fixed at 35% of the width and height of the partition
√	width/w	int	The width of the minute hand shall not exceed 10% of the width of the zone or the height of the zone
<second>		-	<clock> subtag, defining the second hand This tag exists and 'secondColor' is automatically

			ignored.If there is neither the child tag nor 'secondColor', the second hand is not drawn
-	image	string	Second hand picture file path Is empty, means the control card draws a simple straight line
-	color	int	Second hand color: When the image is empty, there is no Default color.
-	length/l	int	The length of the second hand, temporarily unsupported, is temporarily fixed to 40% of the smaller partition width and height
√	width/w	int	The width of the second hand shall not exceed 10% of the width of the zone

3.2.7 Time partition

Label	attribute	Type	Instruction
<timepanel> or <time>		-	Time partition
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
-	duration	int	Play time: Play time: In seconds, Default5 seconds;The Default duration for this partition is defined, and the actual duration of the program is the maximum duration for all partitions in the program
-	bgColor	string	background color， Default'0x00000000' - transparent black
-	timeEquation	string	adjust timeadjust the time， format'hh:mm:ss'， Default'00:00:00'
-	positiveTE	bool	adjust time direction： True is +， False is -， DefaultTrue
<dateTime>		-	<time> subtag, subunit definition The maximum allowable number is tentatively 8 and displayed at the same time (no order).
-	fontColor	string	font color， Default'0xFFFFFFFF' - non-transparent white
√	fontName	string	font namefont name
√	fontSize	int	font size font size
-	fontSizeType	string/int	Font size unit： Default'pixel'/0 is unit "point"/1 unit is Libra
-	fontAttributes	string/int	Font additional attributes: Including 'bold', 'italic', 'normal', 'underline', 'strikeout', etc.Can be combined by '&', e.g. 'bold& italic& underline'
√	contentX	int	The horizontal deviation of the lower left corner of the text rendering area from the upper left corner of the partitionThe horizontal deviation of the lower left corner of the text rendering area from the upper left corner of the partition
√	contentY	int	The vertical offset of the lower left corner of the text rendering area from the upper left corner of the partitionThe vertical offset of the lower left corner of

			the text rendering area from the upper left corner of the partition
√	content	string	Format string conforming to rfc-2822 standard
			"%A": complete English week, e.g. "Monday" "%a": shortened to English week, as "Mon" "%B": full English month, such as "February" "%b": short for month, e.g. "Feb" "%d": two digit (0 added) date, '01' - '31' "%H": two digits show "hour", "00" - "23" "%I": two digits show "time", "01" - "12" "%k": two display "time", "0" - "23", filled with space "%l": two display "when", '1' - '12 ", filled with space "%M": two digits show "cent", "00" - "59" "%p": English upper/afternoon mark, 'AM'/' PM ' "%P": English upper/lower case sign, 'am'/' PM ' "%S": two digits show "seconds", '00' - '59' "%Y": full Gregorian calendar year, such as' 2018 ' "%y": 2 places after the Gregorian calendar year, such as' 18 ' "%m": two-digit (0 supplementary) digital month, '01' - '12'
			Custom string format (modified definition in rfc-2822)
			"%T": sign on/in Chinese, 'am'/' PM ' "%w": Chinese week, 'Monday' - 'Sunday'

3.2.8 timing partition

Label	attribute	Type	Instruction
<countpanel> or <count>		-	Timing partition
√	xCoord/x	int	The X coordinate of the upper left corner of the partition
√	yCoord/y	int	The Y coordinate of the upper left corner of the partition
√	width/w	int	partition width
√	height/h	int	partition height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layerPartition level, 0 represents the bottom level, the higher the number, the higher the level
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Partition transparency (0~100) : 100 is completely opaque, Default is 100
-	duration	int	Play time: Play time: In seconds, Default5 seconds;The Default duration for this partition is defined, and the actual duration of the program is the maximum duration for all partitions in the program
-	bgColor	string	backgound color， Default’ 0x00000000’ - transparent black
√	targetType	string	

			Timing type: 'end' - countdown, start - timing Countdown, the time decreases until the target moment, after which it is always 0;The positive timing starts from the timing moment and is always 0
√	targetDate	string	Target date: Format: 'yyyy-mm-dd', may not be specified;When not specified or empty, the daily loop is timed and switched at 0 local time
√	targetTime	string	target time： format：‘hh:mm:ss’， must specify
-	fontColor	string	font color， Default’0xFFFFFFFF’ - nontransparent white
√	fontName	string	font namefont name
√	fontSize	int	font sizefont size
-	fontSizeType	string/int	font size unit： Default’pixel’/0 is unit “point”/1 unit is Libra
-	fontAttributes	string/int	Font additional attributes: Including 'bold', 'italic', 'normal', 'underline', 'strikeout', etc.Can be combined by '&', e.g. 'bold& italic& underline'
√	contentX	int	The horizontal deviation of the lower left corner of the text rendering area from the upper left corner of the partition
√	contentY	int	The vertical offset of the lower left corner of the text rendering area from the upper left corner of the partition
√	content	string	Format string: 'dd' - days, 'hh' - hours, 'mm' - minutes, 'ss' - seconds;Such as: Distance from target: dd days hh, mm minutes, ss seconds
-	convert	string	Unit conversion (timing and accumulation) : 'no' - no;'yes' - is (Default) When 'yes' is used, if a format string does not exist, its value shall be converted to the next level where a format string exists, and incorporated into it.'no' does not do the above processing, but does not display the corresponding number of the empty format string. Such as: Format string at 1 day 1 hour 2 minutes 10 seconds from target: 'yes' will eventually display' 25 hours 130 seconds';And 'no' will eventually say '1 hour, 10 seconds.'

3.2.9 Lunar calendar partition

Label	attribute	Type	Instruction
<calendarpanel>		-	Lunar calendar partition
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
-	duration	int	Play time:Play time: In seconds, Default5 seconds;The Default duration for this partition is defined, and the actual duration of the program is the maximum duration for all partitions in the program
-	bgColor	string	Background color, Default '0x00000000' - transparent black
-	timeEquation	string	adjust timeadjust the time， format‘hh:mm:ss’，Default‘00:00:00’
-	positiveTE	bool	adjust the time direction：‘True’ is +，‘False’ is - ，Default‘True’
<calendar>		-	<calendarpanel> sub-label, sub-unit definition The maximum allowable number is tentatively 3 and displayed at the same time (no order).
√	mode	string	Various forms of the lunar calendar: Heavenlystem: annals of ganzhi, jiazi - dechai Lunarcalendar: lunarcalendar, such as the sixth day of the first lunar month Solarterms: refers to the solarterm or the number of days away from the next solarterm, such as the beginning of summer, 10 days away from the beginning of summer, etc
-	fontColor	string	Font color, Default '0xFFFFFFFF' - opaque white
√	fontName	string	font name
√	fontSize	int	font size
-	fontSizeType	string/int	font size unit（Default’ pixel’ ）： ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
-	fontAttributes	string/int	Font additional attributes: Including 'bold', 'italic', 'normal', 'underline', 'strikeout', etc.Can be combined by '&', e.g. 'bold& italic& underline'
√	contentX	int	The horizontal deviation of the lower left corner of the text rendering area from the upper left corner of the partition

√	contentY	int	The vertical offset of the lower left corner of the text rendering area from the upper left corner of the partition

3.2.10 Sensor area

Label	attribute	Type	Instruction
<sensorpanel>或<sensor>		-	Sensor partition
√	xCoord / x	int	Area top left X coordinate
√	yCoord / y	int	Area top left Y coordinate
√	width / w	int	Area width
√	height / h	int	Area height
√	zOrder / z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency /t	int	Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
-	duration	int	Play time:
-	bgColor	hex	Background color, Default0x00000000 - transparent black
-	fontColor	hex	font color, Default0xFFFFFFFF -nontransparent white
√	fontName	string	font name
√	fontSize	int	font size
-	fontSizeType	string/int	font size unit（Default’ pixel’ ）： ‘pixel’/0 **unit is pixel ,"point"/1 unit is Libra**
-	fontAttributes	string/int	Font additional attributes: Include 'bold', 'italic', 'normal', etc.Can be combined by '&', for example: 'bold& italic'
√	contentX	int	The horizontal deviation of the lower left corner of the text rendering area from the upper left corner of the partition
√	contentY	int	The vertical offset of the lower left corner of the text rendering area from the upper left corner of the partition
√	content	string	display : '%%d' is the keyword, indicating the value of the sensor. example： ‘current temperature：%%d ℃’
√	threshValue	float	Sensor threshold: such as' -1.0 ', '35.0', etc
√	threshMode	int	Sensor threshold judgment mode (Default: '1') : '0' - less than the threshold value is judged to be valid;'1' - greater than the threshold is considered valid
-	threshFontColor	hex	Over threshold font color, Default0xFFFFFFFF - nontransparent white
-	decimal	int	The number of decimal places to display (0~10) : 0 -- integer mode (example: 25° C), Default value 2 -- decimal mode (example: 25.50° C)
-	unitCoefficient	float	Unit conversion factor Times the original value, so it can't be 0.Default is 1.0, for example: 0.001 - can convert millimeter value from distance sensor to meter
-	correction	float	Modified value (Default is 0) : [final value]=[original value]*[unitCoefficient]+

			[correction]
√	sensorAddress	int	pls checkAppendix 2
√	sequence	int	pls checkAppendix 3
-	unitType	int	Sensor unit identification (valid for specific sensors) : Temperature sensor: 0-celsius;1 to Fahrenheit Liquid level sensor: 0-state (alarm or not);1 - liquid level value Dust sensor: 0-pm2.5;1 - PM10
-	updateTime	int	Sensor value update interval (Default5 seconds) : Unit seconds, ranging from 1 second to 12 hours

3.2.11 Database area

Label	attribute	Type	Instruction
<databasepanel>或<database>		-	Database partition
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
<dbUnit>		–	<database> subtag, subunit definition The maximum allowed number is tentatively 32, playing in order
√	order	int	play order： 0~31
-	bgImage	string	background image file path
√	pageStayTime	int	Page retention time, 3~3600 seconds
–	updateEmptyData	bool/int	Query whether the last data is displayed when the database data is empty: 'true' /1 - displays null, 'false' /0 - holds data from last query The Default is "false"
√	dbType	string	Database server mode： 'mysql' - MySQL 'sqlserver' - MS SQL Server 'postgresql' – PostgreSQL（Temporary does not support）
√	host	string	Server address or domain name
√	port	int	Server port
–	instance	string	'sqlserver' optional parameter（Temporary does not support）
√	user	string	User name
√	password	string	password
–	dbName	string	Database name，'mysql' must have this parameter
√	queryCommand	string	**Database query instruction (global)**
√	displayRows	int	Number of lines per page
√	displayColumns	int	Number of columns per page
–	oddLinesBgColor	string	Odd row background color， Default'0x00000000' - Transparent (black)
–	evenLinesBgColor	string	Even row background color， Default'0x00000000' - Transparent (black)
–	oddLinesFontColor	string	Odd line font color, Default '0xFFFFFFFF' - opaque white
–	evenLinesFontColor	string	Even line font color, Default '0xFFFFFFFF' - opaque white

√	fontName	string	font name
√	fontSize	int	font size
–	fontSizeType	string/int	font size unit（Default‘pixel’）： ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
–	fontAttributes	string	Font additional attributes: Include ‘bold’, ‘italic’, ‘normal’, ‘underline’, ‘strikeout’; combination via‘&’,eg.： ‘bold& italic& underline’
–	alignment_H	string	Horizontal alignment（Default‘center’）： ‘left’ - ， ‘center’ - ， ‘right’ -
–	alignment_V	string	（Default‘center’）： ‘top’ - ， ‘center’ - ， ‘bottom’ -
–	autoLF	bool	Auto line wrap flag: 'True' - auto line wrap， Default‘False’
–	rowToColumn	bool	**Line swap flag: 'True' - line swap, Default 'False' This field simply changes the way the data content is displayed, not the column and column definitions in the style**
–	displayFieldName	bool	Field name display or not flag: 'False' - not displayed,Default‘True’
–	paintTable	bool	Table border drawn or not marked: 'False' - not drawn,Default‘True’
–	tableLineWidth	int	Table line width：1~5
–	tableLinear	string	Grid line style: 'solid' - solid line;'dash' - dotted line;'dot' - dotted line;'dashdot' - dotted;'dashdotdot' - double dotted line;The Default 'solid'
–	tableLineColor	string	Table line color， Default‘0xFFFF0000’ - nontransparent red
<specifyRow>		–	<dbUnit>'s sub tag, defined as a row property, to specify the height of a particular row.You can have many or none.Rows that do not specify height with this label are automatically computed
√	row	int	Specifies the index value of the row： 1 ~ displayRows
–	rowHeight	int	Specifies the height of the row, not less than the height of the font in pixels
–	bgColor	string	Specifies the row background color
–	fontColor	string	Cell font color
–	fontName	string	Table font name
–	fontSize	int	Table font size
–	fontSizeType	string/int	Table font size unit（Default‘pixel’）： ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
–	fontAttributes	string	Font additional attributes: Include ‘bold’, ‘italic’, ‘normal’, ‘underline’, ‘strikeout’; combination via‘&’,eg.： ‘bold&italic&underline’
–	alignment_H	string	Horizontal alignment（Default‘center’）： ‘left’ - ， ‘center’ - ， ‘right’ -
–	alignment_V	string	Vertical alignment（Default‘center’）： ‘top’ - ‘center’ - ， ‘bottom’ -
–	autoLF	bool	Auto wrap line mark：‘True’ - auto wrap line， Default‘False’
<specifyColumn>		–	<dbUnit> subtag, column attribute definition, used to

			specify the width of a particular column.You can have many or none.Without the mark
√	column	int	Specifies the index value of the column： 1 ~ display Columns
–	columnWidth	int	Specifies the height of the column
–	bgColor	string	Specifies the column background color
–	fontColor	string	Table font color
–	fontName	string	font name
–	fontSize	int	font size
–	fontSizeType	string/int	font size（Default‘pixel’）： ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
–	fontAttributes	string	Font additional attributes: Include ‘bold’, ‘italic’, ‘normal’, ‘underline’, ‘strikeout’; combination via ‘&’,eg.： ‘bold& italic& underline’
–	alignment_H	string	Horizontal alignment（Default‘center’）： ‘left’ - , ‘center’ - , ‘right’ -
–	alignment_V	string	Vertical alignment（Default‘center’）： ‘top’ - ‘center’ - , ‘bottom’ -
–	autoLF	bool	Auto wrap line mark：‘True’ - auto wrap line，Default‘False’
<specifyCell>		–	<dbUnit> subtag, column attribute definition, used to specify the width of a particular column.You can have many or none.Without the mark
√	row	int	Specifies the row index value of the cell： 1 ~ display Rows
√	column	int	Specifies the column index value of the cell： 1 ~ display Columns
–	bgColor	string	Specifies the background color of the cell
–	content	string	Forces the specified content of the display for this cell .Ignore global and local ‘queryCommand’
–	fontColor	string	font color
–	fontName	string	font name
–	fontSize	int	font size
–	fontSizeType	string/int	font size（Default‘pixel’）： ‘pixel’/0 unit is pixel ,"point"/1 unit is Libra
–	fontAttributes	string	Font additional attributes: Include ‘bold’, ‘italic’, ‘normal’, ‘underline’, ‘strikeout’; combination via ‘&’,eg.： ‘bold& italic& underline’
–	alignment_H	string	Horizontal alignment（Default‘center’）： ‘left’ - , ‘center’ - , ‘right’ -
–	alignment_V	string	Vertical alignment（Default‘center’）： ‘top’ - , ‘center’ - , ‘bottom’ -
–	autoLF	bool	Line wrap flag: 'True' - line wrap, Default 'False'

Note: 1. The specially specified priority is: specify cell > specify column > specify row > global

3.2.12 Colorful text partition

The partition is a special text partition, which is composed of a text mask picture of the specified format and a character block picture.By superposition, the text strokes in the text mask are replaced by the colors in the core picture to present colorful text.This partition and dazzle the background area combination, can present more colorful text text.

A text mask image is a pre-rendered text image and must be a file format with alpha channel, such as PNG or BMP.The background must be fully transparent (0x00) and the rendered stroke opaque (0xFF).Character core image is no format restrictions, can be in addition to GIF, any control card support image format.

Label	attribute	Type	Instruction
<colortextpanel>或<colortext>		-	Colorful text partition
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
<maskUnit>			sub label of <colortext> , text mask sub-unit definition The maximum allowed is tentatively 128, playing in order
√	order	int	order：0~127
√	file	string	Text mask image file path
√	stuntType	int	Text mask type index: only support 0, 50~53 of these 5 special effects, other types of index will use 0 instead
√	stuntSpeed	int	Text mask skill speed level, 1~16, Default 1 the fastest
√	stayTime	int	Text mask unit dwell time in seconds
√	waveStuntType	int	Overlay wave effect type index (not in the following range when '0' is processed) 0 - no 58 - horizontal stationary wave;59 - horizontal moving wave 60 - vertical stationary wave;61 - vertical moving wave When the text mask moves (no. 50-53 stunt), the wave direction must be consistent with its direction; otherwise, 0 is used instead. When the text mask is still (no. 0 stunt), there is no such restriction
–	waveCount	int	Number of wave peaks: the minimum number is 1, and the maximum number is related to the number of pixels in the wave direction.Horizontal waves depend on the width of the partition, while vertical waves depend on the height of the partition.When less than 200, is the maximum even number not exceeding N/12;When greater than 1000, is the maximum even number not exceeding N/60;The other is 16
–	waveSpeed	int	Fluctuation velocity grade, 1~16;Moving wave is effective, 1 is the fastest
–	waveAmplitude	int	Peak amplitude ratio: the percentage of peak amplitude

			in the width (horizontal wave) or height (vertical wave) of a partition, 1~250 Peak amplitude ratio: the percentage of peak amplitude in the width (horizontal wave) or height (vertical wave) of a partition, 1~250
<hollowUnit>		-	sub- tag of <colortext> , sub- tag unit definition The maximum number is allowed to play in order The core sub-unit is not directly related to the text mask subunit
√	order	int	Play order：0~127
√	file	string	Core image file path
√	stuntType	int	Character core special effects type index: only support 0, 3~6, these 5 special effects type, other types of index will use 0 instead
√	stuntSpeed	int	Character core special effects speed level, 1~16, Default for 1 the fastest
√	stayTime	int	Unit residence time, in seconds

3.2.13 Colorful background area

Label	attribute	Type	Instruction
<animationbg>		-	Colorful background partition
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
-	duration	int	Play time: Play time: In seconds, Default5 seconds;The Default duration for this partition is defined, and the actual duration of the program is the maximum duration for all partitions in the program
√	animationType	int	Animation type selection, temporarily support 0-5 0 - fountain, 1 - circular diffusion, 2 - snowflakes 3 - bubbles, 4 - maple leaves, 5 - block track movement
√	density	int	Density grade, 1 ~ 10 gradually increase; The total density of all the blinding background partitions is limited and will affect each other. When the type is' 1 ', try to use a low density level.
√	size	int	Texture size scale, 1 ~ 10 gradually increase
√	direction	string	Direction, valid when the type is not '1' or '5' :"top","bottom","left","right", "lefttop","leftbottom","righttop", "rightbottom"
√	speed	int	Speed level , 1-10 increase gradually
√	color	string	Texture color (note alpha effect) 0 - represents random color or original texture color For type '4', it is recommended to use 0 (original texture color)
√	taper	int	Taper lever, Valid for type '0', 1-10 increase gradually

3.2.14 Colorful border area

Label	attribute	Type	Instruction
<borderpanel> or <border>		-	Image area
√	xCoord/x	int	Area top left X coordinate
√	yCoord/y	int	Area top left Y coordinate
√	width/w	int	Area width
√	height/h	int	Area height
√	zOrder/z	int	Area layer, 0 represents the bottom layer, the larger the number, the higher the layer Area layer, 0 represents the bottom layer, the larger the number, the higher the layer
-	transparency/t	int	Area transparency (0~100)： 100 means completely opaqu .Default100 Area transparency (0~100)： 100 means completely opaqu .Default100
<unit>			Sub tag of <border>, define image sub unit The maximum number is 128 , play in order.
√	order	int	Play order： 0~127
√	duration	int	Unit play time Unit::second ,default 5 second ,Defines the maximum playback time of the unit.
√	file1	string	Main texture clip image file path: Png, BMP, JPEG and other image formats are supported, and gif format is not supported; PNG or BMP images with transparency are recommended
√	file2	string	Auxiliary texture fragment image file path Support PNG, BMP, JPEG image format, not support GIFformat;PNG or BMP images with transparency are recommended. This texture fragment serves as the background when the border flashes (flickerGrade is not 0);If not specified, the background will be all black.
√	borderWidth	int	Board widht (Unit :pixel)
√	textureWidth	int	Texture fragment width (in pixels) Both the width and border width describe the width and height of the texture fragment image: the texture fragment width section is the width of the image, and the border width is the height of the image
√	stuntType	int	Special effects type index Only support 62, 63, 64 these 3 special effects type, other types of index will use 62 instead; 62 -- stationary;63 -- clockwise rotation;64 -- counterclockwise rotation.
√	stuntSpeed	int	Special effects speed level, 1~ 16,1 is the fastest.
√	flickerGrade	int	Flicker speed level: 0 or 1~ 8,1 is the fastest, 0 is not flicker.

Note：

A colorful border does not normally exist alone, where "duration" specifies a time that only describes the maximum playback time of the unit, not that it needs to be full.For example, the actual running time of the program is 10 seconds, while the "duration" of the colorful border unit is

15, when it reaches 10 seconds, it immediately stops and switches the program (unless there is only one recurring program).

The presence of multiple cells in a partition does not mean that all of them must play, only the actual broadcast time of the program.

3.2.15 Background music area

Label	attribute	Type	Instruction
<muzak>		-	One program just support one background music area; if there are multiple areas, only the first one will be selected; as long as there are background music areas , all other areas with music will be muted.
<unit>		-	Sub tag of < muzak >,sub units definition The max is 128 ,display by order
√	order	int	playback order：0~127
√	volume	int	Unit volume: 1 ~ 100
√	file	string	Music file path Mp3 format is temporarily supported

Note:

This section is not controlled as the actual duration of the program, and other sections will be counted as finished when the time is full.

This function is only formally regarded as a partition.

3.3 Dynamic area

The dynamic area program is temporarily stored in the memory and will not be stored in the local media. Dynamic area can be played independently as a global program, or can be played in association with normal programs. The dynamic area is always displayed on the top of all programs and does not affect the playback of other programs. You can balance other programs by setting the transparency of the dynamic area.

- ✓ When displayed as a global program (unrelated program), the dynamic area will play immediately.
- ✓ When the associated program is played, the playback time of the dynamic zone will depend on or be subordinate to the associated program.
- ✓ Dynamic area time limit: The dynamic area of unrelated programs is the global dynamic zone, and time limit is always valid. The dynamic area of the associated program, the time limit is based on the time limit of the associated program. Only dynamic areas within the aging range can be played.
- ✓ If the dynamic area file is saved, the controller restart will automatically play the saved dynamic zone according to the association and time limit.
- ✓ The dynamic area program is played using the "Dynamic Zone Update" command, while the Stop Dynamic area uses the "Clear Dynamic area " command.

The dynamic zone program is not recommended to be sent in the xml format. This document does not describe the specific format of the xml file. To send a dynamic zone program, please use the dynamic zone management class command in the communication protocol

3.4 Bulletin area program

The bulletin area can be created by command, or uploaded XML format file to to the special directory of the bulletin area directly . To facilitate unified maintenance, when using the upload XML file method, the bulletin file must start with "bulletin" and bulletin number (order field + 1), with ". XML" as the suffix.

If users need to add additional information to the file name, users must use the underscore "_" after the bulletin number. For example, if the order is "0", the bulletin program file is named "bulletin0. XML", or bulletin0_ [MD5]. XML ", and so on. If the name of the bulletin file does not conform to the command rules, it may cause the normal use of the functions related to the management of the bulletin areas.

Tentative: the maximum number of bulletin is 100 (order field: range 0-99).

- ✓ eg：

```
<?xml version="1.0" encoding="utf-8"?>
<bulletin order="0" name="公 告-1" layoutMode="2"
  x="0" y="0" w="1280" h="48" t="100"
  bgColor="#00000000"
  fontSize="18" fontSizeType="point" fontName="SimSun"
  fontColor="0xFFFF0000" fontAttributes="normal"
  stuntType="51" stuntSpeed="50" stayTime="0"
  startDate="" stopDate="" startTime="" stopTime="" weekFlag="127"
  content="Test"/>
```

- ✓ Bulletin area program file labels and properties

Label	attribute	Type	Instruction
<bulletin>		-	bulletin area
	order / z	int	bulletin no. (tentative 0-99)
	name	string	Custom name
	layoutMode	string/int	Layout Mode 'top'/0 - Top ， 'bottom'/0 -Bottom， 'custom'/0 - Custom
	xCoord/x	int	Custom layout top left X coordinate
	yCoord/y	int	Custom layout top left Y coordinate
	width/w	int	Custom layout width
	height/h	int	Custom layout height
	startDate	string	Start date， format：'yyyy-MM-dd',Year month data
	startTime	string	Start time， format：'hh:mm:ss', Hour minute second
	stopDate	string	Stop date，format：'yyyy-MM-dd' ,Year month data
	stopTime	string	Stop time，format：'hh:mm:ss' ,Hour minute second
	weekFlag	int	Week valid attribute: the 1st to 7th digits represent Monday to Sunday respectively, for example: '1' means it only plays on Mondays, '127' means it plays from Monday to Sunday
	transparency/t	int	Overall transparency of partitions (0~100) : Default100 is completely opaque
	bgColor	int	Bullet area background color， Default'0xFF000000' - Opaque black
	fontColor	string	Font color ， Default'0xFFFFFFFF' - Opaque white
	fontName	string	Font name
	fontSize	int	Font size
	fontSizeType	string/int	Font size unit（Default 'pixel'）： 'pixel'/0 In pixels，'point'/1 in point
	fontAttributes	string/int	font additional properties including :'bold','italic','normal','underline', 'strikeout'; Can be combined by '&' ,for example ： 'bold& italic& underline'
	fontAlignment	string	(to be determined)
	stuntType	int	stunt type It is recommended to use stunt types of 0, 50-57 only
	stuntSpeed	int	Stunt speed level, 1-16, 1 is the fastest
	stayTime	int	Stunt stay time in seconds (Provisional)
	content	string	text content

Appendix 1 Image/ text area display effect table

Label	attribute	Type
0	Quick type	
1	Random Display	Y Series only random show 128 or above new stunts
2	Static	merged with quick type
3	Move up	The old image move out, the new image moves in Speed level 16 (slowest) is point by point movement
4	Move down	
5	Move left	
6	Move right	
7	slide down	The old image does not move, the new image moves in Speed level 16 (slowest) is point by point movement
8	slide up	
9	slide right	
10	slide left	
11	build up from up	Laser effect
12	build up from down	
13	build up from left	
14	build up from right	
15	open up	Image not move , new image replace old one gradually
16	open down	
17	open left	
18	open right	
19	open top left	
20	open top right	
21	open down left	
22	open downright	
23	corner to center (rectangle)	
24	corner to center （cross）	
25	Center to corner （cross）	
26	horizontal cross open	
27	vertical cross open	
28	vertical shutter open	
29	Horizontal louver curtain	
30	closed(up and down)	
31	open((up and down)	
32	open(left and right)	
33	close (left and right)	
34	small to bigger(center)	
35	Mosaic	
36	Fade in ,fade out	
38	Strip erase	
39	Pull the curtain to the corner (rectangle)	Image not move , new image replace old one gradually
40	center reduction	
41	center (trailor)	
42	Stretch to the left	
43	Stretch to the right	
50	Smooth push upwards	Push: the old image moves out, the new image moves in Move in: the old image does not move, the new image moves in
51	Smooth push down	
52	Smooth push left	

53	Smooth push right	The eight stunts are optimized mobile stunts, corresponding to 3-8 stunts. Speed level 1,2:5 points / 1 frame; speed level 3,4:4 points / 1 frame; Speed level 5,6:3 points / 1 frame; speed level 7,8:2 points / 1 frame; Speed level 9,10:1 point / 1 frame (point by point); Speed level 11,12:1 point / 2 frame; speed level 13,14:1 point / 2 frame; Speed class 15,16:1 point / 3 frames
54	Smooth move down	
55	Smooth move up	
56	Smooth move right	
57	Smooth move left	
58 | 61	More details pls checkClorful text partition	Parameterized special effects ,only applicable to the special effects that are superimposed in the colorful word area; other area do not support the parameterization of special effects, and the effect is unknown, so it is not recommended to use
62-64	More details pls check colorful boarder area	Stunt type index for colorful borders only
Random stunts (1) will only show the following stunts		
128	Cross shutter	
129	Roll left	
130	scroll expanding	
131	Roll both sides open	
132	Rotary curtain	
133	Rotary enlarge	
134	Quick type,rotary reduction	
135	flip over	
136	page left	
137	page right	
138	page up	
139	page down	
140	Left symmetrical open	
141	Left symmetrical close	
142	Shutter open	
143	pull from center to around （X）	
144	pull from around to center （X）	
145	center spreads toward the four corners（cross）	
146	open door(push)	
147	Ripples fade in and out	
148	Center diffusion fade in and out	
149	Center accumulation fade in and out	
150	Multi point diffusion fade in and out	
151	Broke screen	

152	Rotary Speaker	
153	Louver flip	
154	Four point diffusion	
155	Split and fall	
156	Lamination	
157	Mosaic fade in and out	
158	Sphere falling expansion	
159	Cube rotation near	

Appendix 2 sensor address table

The sensor address uses the unsigned short shaping: 0x0000 ~ 0xFFFF, in which the high 8 bits are used to identify the sensor bus interface; the low 8 bits are used to identify the device address on different buses

Address	Description	Note
0x0000~0x00FF	Reserved	
0x0100~0x07FF	（One Wire） On board single bus (one wire)interface	Maximum 7 onboard single bus interfaces are supported; bx-y series only has No. 1, which is temporarily unavailable.
0x0800~0x0FFF	On board I^2C bus interface	Maximum 8 onboard single bus interfaces are supported; bx-y series only has No. 1
0x1000~0x17FF	On board serial 232 interface	Maximum 8 onboard RS232 interfaces are supported; bx-y series only has No. 1
0x1800~0x1FFF	On board serial RS232 interface	Maximum 8 onboard RS485 interfaces are supported; bx-y series only has No. 1 and share with RS232
0x2000~0x27FF	USB to serial RS232 interface	Maximum 8 onboard USB to RS232 interfaces are supported; bx-y series can support no 1 and no 2, the signal USB is no 1, if there are two USB ports ,the down one is no 1.
0x2800~0x2FFF	USB to serial RS485 interface	Maximum 8 onboard USB to RS232 interfaces are supported; bx-y series can support no 1 and no 2, the signal USB is no 1, if there are two USB ports ,the down one is no 1.
0x3000~0x7FFF	Reserved	
0x8000~0x80FF	Lan 1 BX-VMF address range	The address of the first multi-function board connected in series under LAN 1 is 0x8000, the address of LAN2 is 0x8001, and so on
0x8100~0x81FF	Lan 2 BX-VMF address range	
0x8F00~0x8FFF	Lan 16 BX-VMF address range	
0x9000~0xFFFF	Reserved	

Sensor address table

Address	Description
0	No sensor
1	Air quality sensor PM2.5（unitType=0）或 PM10（unitType=1）
2	Wind speed sensor
3	Wind direction sensor
4	Noise sensor
5	temp sensor（sequence=261）humidity sensor（sequence=262）in the integrate sensor
6	Noise sensor in environment sensor
7	PM2.5（unitType=0）or PM10（unitType=1）
8	Atmospheric pressure sensor in environment sensor
9	Light intensity sensor in integrate sensor

Appendix 3 Sensor function serial umber table

The sensor function sequence number is used to distinguish the sensor type on the multifunction card, and other sensor types are uniquely determined by the sensor address (8-bit bus address + 8-bit device address).

Item	description
0	No sensor, or unknown sensor
The sensors implemented on BX-VMF (also applicable to on-board sensors, only# 1 bus) ,if not specified , all the sensors are made by Onbon	
1	I^2C bus brightness sensor
2	Temperature sensor, when the sensor address is in the range of the multi-functional card, it corresponds to the temperature sensor connected on the # 1 single bus on the multi-functional card; if there is a humidity sensor on the bus, it corresponds to the temperature sensor in the temperature &humidity sensor
3	Temperature sensor, only when the sensor address is in the range of the multi-functional card, it corresponds to the temperature sensor connected on the # 2 single bus on the multi-functional card; if there is a humidity sensor on the bus, it corresponds to the temperature sensor in the temperature &humidity sensor
4	The humidity sensor in the temperature &humidity sensor, when the sensor address is the range of the multi-functional card, corresponds to the humidity sensor in the temperature &humidity sensor connected on the # 1 single bus on the multi-functional card(no separate humidity sensor)
5	The temperature sensor in the temperature &humidity sensor, when the sensor address is the range of the multi-functional card, corresponds to the humidity sensor in the temperature &humidity sensor)
6	Smoke sensor (alarm)
7~255	Reserved , same as 0
256-512:Sensors supporting Modbus Protocol (mainly meteorological sensors at present) ,if not specified , all the sensors are made by Onbon	
256	Air quality sensor(dusty sensor PM2.5)
257	Wind speed sensor
258	Wind direction sensor
259	Noise sensor
260	Air quality sensor(dusty sensor PM10)
261	Temperature sensor
262	humidity sensor
263	Noise sensor
264	Air quality sensor PM2.5
265	Air quality sensor PM10
266	Atmospheric pressure sensor
267	Light sensor
268~512	Reserved , deal as 0
512~：Special customized sensor	
512~	Reserved , Deal as 0