

BX-IV Dynamic Library User Manual

Date:2012.07.30

1. Controller parameters setting:

Before you connect with the DLL, please use our LedshowTW2012 software to set up the parameter of the controller, after setting up, you could set the screen parameter ,data and command in the DLL connection system.

Presently, Ethernet communication mode only supports fixed IP mode.

2. Functions:

Every executed function with result back, and users could judge the conditions of the function according to the result, also prompt the users the next step.

The below is the back macro definition of some functions:

RETURN_ERROR_AERETYPE=0xF7;// Area type error when add/delete image area files;

RETURN_ERROR_RA_SCREENNO=0xF8;// The display screen information exist. And if you want to reset it, please delete the screen first;

RETURN_ERROR_NOFIND_AREAFILE =0xF9; // Do not find the valid area file(Image/Text area);

RETURN_ERROR_NOFIND_AREA=0xFA; // Do not find the valid display area; could add area information by AddScreenProgramBmpTextArea;

RETURN_ERROR_NOFIND_PROGRAM=0xFB;// Do not find the valid display screen program; could add program by AddScreenProgram;

RETURN_ERROR_NOFIND_SCREENNO=0xFC;// Do not find the display screen; could add display screen by AddScreen;

RETURN_ERROR_NOW_SENDING = 0xFD; // It's connecting now with this display screen, please try later;

RETURN_ERROR_OTHER =0x FF; // Other errors;

RETURN_ERROR_NOSUPPORT_USB = 0xF6;// This controller does not support USB mode;

RETURN_ERROR_NO_USB_DISK=0xF5;// Cannot fund usb equipment;

RETURN_NOERROR =0x00; // No error;

1) function InitDLLResource(AppHandle: Integer): integer; stdcall;

Function:

Initialize the dynamic library resources, you must call this function before calling other functions.

Parameters:

AppHandle:

2) procedure FreeDLLResource; stdcall;

Function:

Release dynamic library resources, and exit the program before calling the function.

Parameters:No

3) function AddScreen(nControlType, nScreenNo, nWidth, nHeight, nScreenType, nPixelMode: Integer; nDataDA, nDataOE: Integer; nRowOrder, nFreqPar: Integer; pCom: PChar; nBaud: Integer; pSocketIP: PChar; nSocketPort: Integer; pScreenStatusFile: PChar): integer; stdcall;

Function:

Add/Set display screen

Parameter:

nControlType: Controller mode; The below showed with Hexadecimal; And if user need Decimalism, You could exchange it.

BX-5AT = 0x0051;

BX-5A0 = 0x0151;

BX-5A1 = 0x0251;

BX-5A1&WiFi = 0x0651;

BX-5A2 = 0x0351;

BX-5A2&RF = 0x1351;

BX-5A2&WiFi = 0x0751;

BX-5A3 = 0x0451;

BX-5A4 = 0x0551;

BX-5A4&RF = 0x1551;

BX-5A4&WiFi = 0x0851;

BX-5M1 = 0x0052;

BX-5M2 = 0x0252;

BX-5M3 = 0x0352;

BX-5M4 = 0x0452;

BX-5UT = 0x0055;

BX-5U0 = 0x0155;

BX-5U1 = 0x0255;

BX-5U2 = 0x0355;
BX-5U3 = 0x0455;
BX-5U4 = 0x0555;
BX-5E2 = 0x0254;
BX-5E3 = 0x0354;
BX-4T1 = 0x0140;
BX-4T2 = 0x0240;
BX-4T3 = 0x0340;
BX-4A1 = 0x0141;
BX-4A2 = 0x0241;
BX-4A3 = 0x0341;
BX-4AQ = 0x1041;
BX-4A = 0x0041;
BX-4UT = 0x0445;
BX-4U0 = 0x0045;
BX-4U1 = 0x0145;
BX-4U2 = 0x0245;
BX-4U2X = 0x0545;
BX-4U3 = 0x0345;
BX-4M0 = 0x0242;
BX-4M1 = 0x0142;
BX-4M = 0x0042;
BX-4MC = 0x0C42;
BX-4C = 0x0043;
BX-4E1 = 0x0144;
BX-4E = 0x0044;
BX-4EL = 0x0844;
BX-3T = 0x0010;
BX-3A1 = 0x0021;
BX-3A2 = 0x0022;
BX-3A = 0x0020;
BX-3M = 0x0030;

nScreenNo : Start from 1

nWidth: Width of display screen: integral multiple of 16; Minimum: 64;

nHeight: Height of display screen: integral multiple of 16; Minimum:64;

nScreenType: Type of display screen: 1: single color; 2: dual-color; 3: RGB; Note: Only BX-4MC supports RGB.

nPixelMode: Pixels type; 1: R+G 2:G+R; This parameters only support dual-color; Default: 2;

nDataDA: Data polarity: 0x00; Negative:0x01; Positive; Default: 0;

nDataOE: OE polarity; 0x00: OE low; 0x01: OE high; Default: 0;
 nRowOrder: Row order mode; 0:normal; 1: add one line; 2: delete one line; Default:0;
 nFreqPar: Scan frequency: 0~6; Default: 0;
 pCom: Serial port name; Valid in serial port communication mode; Example: COM1
 nBaud: Serial port baud rate: currently with 9600, 57600; Default: 57600; Note: BX-3T only support 9600 baud rate.

pSocketIP: Controller IP address; Valid in Ethernet communication mode; Example: 192.168.0.235

pSocketPort: Controller Ethernet port; Valid in Ethernet communication mode; Example: 5005

pScreenStatusFile: The saved INI file name which used for saving found display screen status parameters.

Return value: Integer

RETURN_NOERROR: Return without error;

RETURN_ERROR_RA_SCREENNO: The information of this screen existed. If users want to reset, please delete this screen by DeleteScreen;

RETURN_ERROR_OTHER: Add parameters error;

4) FunctionAddScreenProgram(nScreenNo,nProgramType:Integer;nPlayLength: Integer;nStartYear, nStartMonth, nStartDay, nEndYear, nEndMonth, nEndDay: Integer; nMonPlay, nTuesPlay, nWedPlay, nThursPlay, bFriPlay, nSatPlay, nSunPlay: integer; nStartHour, nStartMinute, nEndHour, nEndMinute: Integer): Integer; stdcall;

Parameters function:

Add program on the display screen;

Parameters:

nScreenNo: From number one

nProgramType : Program type; 0 normal program

nPlayLength: 0: Automatically play by order, or the length of the playing program; Range: 1~65535; Unit: seconds

nStartYear: The period of program: The year when you play. If it's without limit, then the parameter is 65535; Like 2010

nStartMonth: The period of program: The month when you play. Like 11

nStartDay: The period of program: The date when you play. Like 26

nEndYear : The period of program: The year when you stop playing. Like 2011

nEndMonth: The period of program: The month when you stop playing. Like 11

nEndDay: The period of program: The date when you stop playing. Like 26
nMonPlay: It plays or not on Monday in the period of program; 0: no; 1: yes.
nTuesPlay: It plays or not on Tuesday in the period of program; 0: no; 1: yes.
nWedPlay : It plays or not on Wednesday in the period of program; 0: no; 1: yes.
nThursPlay: It plays or not on Thursday in the period of program; 0: no; 1: yes.
bFriPlay : It plays or not on Friday in the period of program; 0: no; 1: yes.
nSatPlay : It plays or not on Saturday in the period of program; 0: no; 1: yes.
nSunPlay : It plays or not on Sunday in the period of program; 0: no; 1: yes.
nStartHour: The time start to play(hour) at that day. Like: 8
nStartMinute : The time start to play(minute) at that day. Like: 0
nEndHour : The time stop to play(hour) at that day. Like:8
nEndMinute: The time stop to play(minute) at that day. Like:0

Return value: Integer

RETURN_NOERROR: Return without error;
RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in the system;
Then add display screen by AddScreen
RETURN_ERROR_OTHER: Other errors

5) Function AddScreenProgramBmpTextArea(nScreenNo, nProgramOrd:Integer;nX,nY,nWidth,nHeight:integer): Integer; stdcall;

Parameters function:

Add Image&Text area on the specified program of the specified display screen;

Parameters:

nScreenNo: From number one
nProgramOrd: Program number; From number zero
nX: The abscissa of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0
nY: The ordinate of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0
nWidth: The width of area; The maximum value does not bigger than the width of display screen-nX
nHeight: The height of area; The maximum value does not bigger than the height of display screen-nY

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_AERETYPE: Area type error when add/delete image area files. Then, return.

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in the system; Then add display screen by AddScreen

RETURN_ERROR_NOFIND_PROGRAM: Do not find the specified program in this display screen of system, users could add specified program by AddScreenProgram

RETURN_ERROR_OTHER: Other error

6) Function AddScreenProgramAreaBmpTextFile(nScreenNo, nProgramOrd, nAreaOrd: Integer; pFileName: PChar; nShowSingle: Integer; pFontName: PChar; nFontSize, nBold, nFontColor: Integer; nStunt, nRunSpeed, nShowTime: Integer): Integer; stdcall;

Parameter function:

Add file on the specified area in program of the specified display screen;

Parameter:

nScreenNo: From number one

nProgramOrd: Program nnumber; from number zero

nAreaOrd: Area number; from number zero

pFileName: File name: Support *.BMP, *.RTF, *.TXT format "add TXT file". It supports the function of end to end when the display mode is continuous move left or continuous move right.

nShowSingle: Single or multi row display; 1: single row display; 0: multi row display; This parameter is valid when pFileName is txt format.

pFontName: Font name; Support the current installed vector font; This parameter is valid when pFileName is txt format.

nFontSize: Font size; Support the current size; This parameter is valid when pFileName is txt format.

nBold: Font bold; Support 1: bold; 0: normal; This parameter is valid when pFileName is txt format.

nFontColor: Font color; 0: black; 1: red; 2: green; 3: yellow; 4: blue; 5: mauve; 6: light green; 7: white; This parameter is valid when pFileName is txt format.

nStunt: Display mode

0x00: Random

0x01: static

0x02: direct show

0x03: move left
 0x04: continuous move left
 0x05: move up BX-3T do not support
 0x06: continuous move up BX-3T do not support
 0x07: flicker BX-3T do not support
 0x08: snowing
 0x09: bubbling
 0x0A: middle out
 0x0B: move around
 0x0C: horizontal cross move
 0x0D: vertical cross move
 0x0E: scroll closed
 0x0F: scroll opened
 0x10: left stretch
 0x11: right stretch
 0x12: up stretch
 0x13: down stretch BX-3T do not support
 0x14: left laser
 0x15: right laser
 0x16: up laser
 0x17: down laser
 0x18: cross curtain left and right
 0x19: cross curtain up and down 3T、3A、4A、3A1、3A2、4A1、4A2、4A3、4AQ do not support
 0x1A: curtain scattered to the left 3T、3A、4A、3A1、3A2、4A1、4A2、4A3、4AQ do not support
 0x1B: horizontal blinds
 0x1C: vertical blinds 3M、4M、4M1、4MC 类型控制卡无此特技 do not support
 0x1D: Curtain left 3T、3A、4A do not support
 0x1E: Curtain right 3T、3A、4A do not support
 0x1F: Curtain up 3T、3A、4A do not support
 0x20: Curtain down (Above:except 3T/3A/4A) 3T、3A、4A do not support
 0x21: Move to center from left and right 3T do not support
 0x22: Split to left and right 3T do not support
 0x23: Move to center from up and down 3T do not support
 0x24: Split to up and down 3T do not support
 0x25: Move right

0x26: Continuum move right

0x27: Move down 3T do not support

0x28: Continuum move down 3T do not support

nRunSpeed: Run speed; 0-63; Run slow if the value is big

nShowTime: Hold time; 0-65525; unit: 0.5 seconds

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in the system;
Then add display screen by AddScreen

RETURN_ERROR_NOFIND_PROGRAM: Do not find the specified program in this
display screen of system, users could add specified program by AddScreenProgram

RETURN_ERROR_NOFIND_AREA: Do not find the specified area in the program of display
screen.

RETURN_ERROR_OTHER: Other errors

7) function DeleteScreen(nScreenNo: Integer): Integer; stdcall;

Parameter function:

Delete specified display screen, and delete all the informations there.

Parameters:

nScreenNo: From number one

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_NOFIND_SCREENNO: No screen in the system

RETURN_ERROR_OTHER: Other errors

8) Function DeleteScreenProgram(nScreenNo,nProgramOrd:Integer):Integer; stdcall;

Parameter function:Delete the program, and delete all the information there.

Parameter:nScreenNo: From number one

nProgramOrd: Program number; From number zero

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in the system;

RETURN_ERROR_NOFIND_PROGRAM: No specified program of the screen in the system

RETURN_ERROR_OTHER: Other errors

9) function DeleteScreenProgramArea(nScreenNo, nProgramOrd, nAreaOrd:

Integer):Integer; stdcall;

Parameter function:

Delete the specified area, and delete all the informations there.

Parameters:

nScreenNo: From number one

nProgramOrd: Program number; From number zero

nAreaOrd: Area number; From number zero

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_AERETYPE: Area type error when add/delete the Image/Text area file.

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in the system;

RETURN_ERROR_NOFIND_PROGRAM: Do not find the specified program of the display screen in the system;

RETURN_ERROR_NOFIND_AREA: Do not find the specified area in the program of display screen.

RETURN_ERROR_OTHER: Other errors

10) function DeleteScreenProgramAreaBmpTextFile(nScreenNo, nProgramOrd, nAreaOrd, nFileOrd: Integer): Integer; stdcall;

Parameter function:

Delete the specified area, and delete all the informations there.

Parameter:

nScreenNo: From number one

nProgramOrd: Program number; From number zero

nAreaOrd: Area number; From number zero

nFileOrd: Area files number; from number zero

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in the system;

RETURN_ERROR_NOFIND_PROGRAM: Do not find the specified program of the display screen in the system;

RETURN_ERROR_NOFIND_AREA: Do not find the specified area in the program of display

screen;

RETURN_ERROR_NOFIND_AREAFILE: Do not find the specified file of Area in the program;

RETURN_ERROR_OTHER: Other error

11) function SendScreenInfo(nScreenNo, nSendMode, nSendCmd, nOtherParam1: Integer): Integer; stdcall;

Parameter function:

Send the corresponding command to display screen.

Parameters:

nScreenNo: From number one

nSendMode: Communication mode: serial port: 0; Ethernet mode: 2;

nSendCmd:

SEND_CMD_PARAMETER = 0xA1FF; // Write screen parameters

SEND_CMD_SENALLPROGRAM = 0xA1F0; // Send all the program information

SEND_CMD_POWERON = 0xA2FF; // Forced ON

SEND_CMD_POWEROFF = 0xA2FE; // Forced OFF

SEND_CMD_TIMERPOWERONOFF = 0xA2FD; // Automatically ON/OFF

SEND_CMD_CANCEL_TIMERPOWERONOFF = 0xA2FC; // Delete automatically ON/OFF

SEND_CMD_RESIVETIME = 0xA2FB; // Correction

time

SEND_CMD_ADJUSTLIGHT = 0xA2FA; //Adjust brightness

nOtherParam1: Other parameters

Return value: Integer

RETURN_NOERROR: Return without error

0x01~0x18, 0xFE: Communication error

RETURN_ERROR_NOFIND_AREA: Do not find the valid display area; then users could add the area information by AddScreenProgramBmpTextArea

RETURN_ERROR_NOFIND_PROGRAM: Do not find the valid display screen program; then users could add the specified program by AddScreenProgram

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in system; then users could add display screen by AddScreen

RETURN_ERROR_NOW_SENDING: It's communicating now, please communicate later;

RETURN_ERROR_OTHER: Other errors

- 12) **function SetScreenTimerPowerONOFF(nScreenNo: Integer; nOnHour1, nOnMinute1, nOffHour1, nOffMinute1, nOnHour2, nOnMinute2, nOffHour2, nOffMinute2, nOnHour3, nOnMinute3, nOffHour3, nOffMinute3: Integer): Integer; stdcall;**

Parameter function:

You could set 3 groups ON/OFF time when set automatically ON/OFF. And if users want to send the set value to display screen, you need to send the command of automatically ON/OFF by SendScreenInfo.

Parameter:

nScreenNo: From number one

nOnHour1: The hour for ON of the first group automatically ON/OFF

nOnMinute1: The minute for ON of the first group automatically ON/OFF

nOffHour1: The hour for OFF of the first group automatically ON/OFF

nOffMinute1: The minute for OFF of the first group automatically ON/OFF

nOnHour2: The hour for ON of the second group automatically ON/OFF

nOnMinute2: The minute for ON of the second group automatically ON/OFF

nOffHour2: The hour for OFF of the second group automatically ON/OFF

nOffMinute2: The hour for OFF of the second group automatically ON/OFF

nOnHour3: The hour for ON of the third group automatically ON/OFF

nOnMinute3: The minute for ON of the third group automatically ON/OFF

nOffHour3: The hour for OFF of the third group automatically ON/OFF

nOffMinute3: The hour for OFF of the third group automatically ON/OFF

Return value: Integer

RETURN_NOERROR: Return without error

0x01~0x18, 0xFE: Communication error

RETURN_ERROR_NOFIND_SCREENNO: Do not find the display screen in system; then users could add display screen by AddScreen

RETURN_ERROR_OTHER: Other errors

- 13) **Function SetScreenAdjustLight(nScreenNo: Integer; nAdjustType, nHandleLight: Integer; nHour1, nMinute1, nLight1, nHour2, nMinute2, nLight2, nHour3, nMinute3, nLight3, nHour4, nMinute4, nLight4: Integer): Integer; stdcall;**

Parameter function:

Set the brightness of the screen. If users want to send the set values to screen, you need

to send automatically adjust brightness by SendScreenInfo.

Parameter:

nScreenNo: From number one

nAdjustType: The type of adjusting brightness; 0: Manually; 1: Automatically

nHandleLight: The brightness value of manually adjust brightness, only valid when nAdjustType=0

nHour: The time(hour) for the first group automatically brightness

nMinute: The time(minute) for the first group automatically brightness

nLight1: The brightness value for the first group automatically brightness

nHour2: The time(hour) for the second group automatically brightness

nMinute2: The time(minute) for the second group automatically brightness

nLight2: The brightness value for the second group automatically brightness

nHour3: The time(hour) for the third group automatically brightness

nMinute3: The time(minute) for the third group automatically brightness

nLight3: The brightness value for the third group automatically brightness

nHour4: The time(hour) for the forth group automatically brightness

nMinute4: The time(minute) for the forth group automatically brightness

nLight4: The brightness value for the forth group automatically brightness

Return value: Integer

RETURN_NOERROR: Return without error

0x01~0x18, 0xFE: Communication error

RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in system; then users could add display screen by AddScreen

RETURN_ERROR_OTHER: Other errors

14) function SaveUSBScreenInfo(nScreenNo: Integer; bCorrectTime, nAdvanceHour, nAdvanceMinute: Integer; pUSBDisk: pChar): Integer; stdcall;

Parameter function:

Save the information to U disk, It's convenient for users to update the screen informations. This parameter is the same with the USB download function in LedshowTW2011.

Supported controller type:

BX-4UT、 BX-4U0、 BX-4U1、 BX-4U2、 BX-4U3 support saving the parameters/correction time parameters and program data to U disk.

Note: Do not support the controllers which did not listed.

Parameter:

nScreenNo: From number one

bCorrectTime: Correcting time or not, 1: correcting time; 0: not correcting time. This parameter only supports BX-4UT、BX-4U0、BX-4U1、BX-4U2、BX-4U3.

nAdvanceHour: The advanced hour of the correction time with the current computer. Range: 0-23; And only valid when bCorrectTime=1.

nAdvanceMinute: The advanced minute of the correction time with the current computer. Range: 0-59; And only valid when bCorrectTime=1.

pUSBDisk: USB equipment route name; Format is "xx:\", like: "F:\"

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_NOSUPPORT_USB: These controllers do not support USB mode;

RETURN_ERROR_NO_USB_DISK: Cannot find USB equipment route;

15) AddScreenProgramTimeArea(nScreenNo,nProgramOrd: Integer; nX, nY, nWidth, nHeight: integer): Integer; stdcall;

Parameter function: Add time area on the program of display screen;

Parameter:

nScreenNo: From number one

nProgramOrd: Program number; From number zero

nX: The abscissa of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0

nY: The ordinate of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0

nWidth: The width of area; The maximum value does not bigger than the width of display screen-nX

nHeight: The height of area; The maximum value does not bigger than the height of display screen-nY

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_AERETYPE: Area type error when add/delete time area file.

RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in system; then users could add screen by AddScreen

RETURN_ERROR_NOFIND_PROGRAM: Do not find the program in system; then users could add the program by AddScreenProgram

RETURN_ERROR_OTHER: Other errors

16) AddScreenProgramTimeAreaFile(nScreenNo, nProgramOrd, nAreaOrd:Integer;pInputtxt,pFontName:PChar;nSingal,nAlign,nFontSize,nBold,nItalic,nUnderline:Integer;nUsetxt,nTxtcolor,nUseymd,nYmdstyle,nYmdcolor,nUseweek,nWeekstyle,nWeekcolor,nUsehns,nHnsstyle, nHnscolor, nAutoset: Integer) : Integer; stdcall;

Parameter function:

Add time file on the time area of program in display screen;

Parameters:

nScreenNo: From number one

nProgramOrd: Program number; from number zero

nAreaOrd: Area number; from number zero

pInputtxt: Fixed words

pFontName: words' font

nSingal: Single row or multiple rows, 0: single row 1: multi rows, and nAlign does not worked in single row

nAlign: Text alignment mode is valid to multi rows, 0: left 1: middle 2: right

nFontSize: Words' size

nBold: Bold or not, 0: no 1: yes

nItalic: Italic or not, 0: no 1: yes

nUnderline: Underscore or not, 0: no 1: yes

nUsetxt: Used fixed words or not, 0: no 1: yes

nTxtcolor: Fixed words color, decimalism of transferring color: red: 255 green: 65280 yellow 65535

nUseymd: Use year/month/day or not, 0: no 1: yes

nYmdstyle: Format of year/month/day, listed below

nYmdcolor: Color of year/month/day, decimalism of transferring color

nUseweek: Use week or not, 0: no 1: yes

nWeekstyle: Week format, listed below

nWeekcolor: Week color, decimalism of transferring color

nUsehns: Used hour/minute/seconds or not, 0: no 1: yes

nHnsstyle: Hour/minute/seconds format, listed below
nHnscolor: Hour/minute/seconds color, decimalism of transferring color
nAutoset: Automatically set corresponding width or not, 0: no 1: yes(default: no)

0	2000年12月30号	0	星期一	0	20时59分59秒
1	00年12月30号	1	Monday	1	20:59:59
2	12/30/2000	2	Mon	2	20 59 59
3	2000/12/30	3		3	8 59 59
4	00-12-30	4		4	20时59分
5	00.12.30	5		5	20:59
6	12月30号	6		6	上午
7	12.30.2000	7		7	AM
8	2000-12-30	8		8	20(时)
9	30 12 2000	9		9	59(分)
10	30:12:2000	10		10	59(秒)
11	30:12:00	11		11	上午 8:59
12	2000	12		12	AM 8:59
13	12	13		13	8:59 AM
14	30	14		14	AM 8 59
		15		15	8 59 AM
		16		16	8 59 上午

Return value:
RETURN_NOERROR: Return without error
RETURN_ERROR_AERETYPE: Area type error when add/delete lunar area files
RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in system; then users could add screen by AddScreen
RETURN_ERROR_NOFIND_PROGRAM: Do not find the program in system; then users could add the program by AddScreenProgram
RETURN_ERROR_OTHER: Other errors

17) AddScreenProgramClockArea(nScreenNo,nProgramOrd: Integer; nX, nY, nWidth, nHeight: Integer): Integer; stdcall;

Parameter function: Add dial area on the program;

Parameter:

nScreenNo: From number one

nProgramOrd: Program number; from number zero

nX: The abscissa of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0

nY: The ordinate of area; On the top left corner of the display screen, the abscissa is 0; minimum value is 0

nWidth: The width of area; The maximum value does not bigger than the width of display screen-nX

nHeight: The height of area; The maximum value does not bigger than the height of display screen-nY

Return value: Integer

RETURN_NOERROR: Return without error

RETURN_ERROR_AERETYPE: Area type error when add/delete dial area file.

RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in system; then users could add screen by AddScreen

RETURN_ERROR_NOFIND_PROGRAM: Do not find the program in system; then users could add the program by AddScreenProgram

RETURN_ERROR_OTHER: Other errors

18) AddScreenProgramClockAreaFile(nScreenNo, nProgramOrd, nAreaOrd, nusetxt, nusetime, nuseweek, ntimeStyle, nWeekStyle, ntxtfontsize, ntxtfontcolor, ntxtbold, ntxtitalic, ntxtunderline, txtleft, txttop, ntimefontsize, ntimefontcolor, ntimebold, ntimeitalic, ntimeunderline, timeleft, timetop, nweekfontsize, nweekfontcolor, nweekbold, nweekitalic, nweekunderline, weekleft, weektop, nclockfontsize, nclockfontcolor, nclockbold, nclockitalic, nclockunderline, clockcentersize, clockcentercolor, mhrdotstyle, mhrdotsize, mhrdotcolor, hrdotstyle, hrdotsize, hrdotcolor, mindotstyle, mindotsize, mindotcolor, hrhandsize, hrhandcolor, minhandsize, minhandcolor, sechandsize, sechandcolor, nAutoset: integer; btxtcontent, btxtfontname, btimefontname, bweekfontname, bclockfontname: pchar): Integer; stdcall;

Parameter function:

Add dial file on dial area of program in the display screen; and if it added, well, modify;

Parameter:

nScreenNo: From number one

nProgramOrd: Program number; From number zero

nAreaOrd: Area number; From number zero

Nusetxt: Use fixed words or not, 0: no 1: yes

Nusetime: Use year/month/day or not, 0: no 1: yes

Nuseweek: Use week or not, 0: no 1: yes

ntimeStyle: Year/month,day format, showed as the list

nWeekStyle: Week format, showed as the list

ntxtfontsize: The size of the fixed words

ntxtfontcolor: The color of the fixed words; Decimalism for transferring color Red: 255
Green:65280 Yellow:65535

ntxtbold: Fixed words: Bold or not, 0: no 1: yes

ntxtitalic: Fixed words: Italic or not, 0: no 1: yes

ntxtunderline: Fixed words: Underline or not, 0: no 1: yes

txtleft: Fixed words on the X coordinate of dial area

txttop: Fixed words on the Y coordinate of dial area

ntimefontsize: Words size of Year/month/day

ntimefontcolor: Words color of Year/month/day; Decimalism of transferring color

ntimebold: Year/month/day: Bold or not, 0: no 1: yes

ntimeitalic: Year/month/day: Italic or not, 0: no 1: yes

ntimeunderline: Year/month/day: Underline or not, 0: no 1:yes

timeleft: Year/month/day on the X coordinate of dial area

timetop: Year/month/day on the Y coordinate of dial area

nweekfontsize: The size of Week words

nweekfontcolor: Color of Week words; Decimalism for transferring color

nweekbold: Week words: Bold or not, 0: no 1: yes

nweekitalic: Week words: Italic or not, 0: no 1: yes

nweekunderline: Week words: Underline or not, 0: no 1: yes

weekleft: Week words on the X coordinate of dial area

weektop: Week words on the Y coordinate of dial area

nclockfontsize: Size of dial words

nclockfontcolor: Color of dial words; Decimalism for transferring color

nclockbold: Week words: Bold or not, 0: no 1: yes

nclockitalic: Week words: Italic or not, 0: no 1: yes

nclockunderline: Week words: Underline or not, 0: no 1: yes

clockcentercolor: Color of dial center; Decimalism of transferring color

mhrdotstyle: 3/6/9 time point type, 0: linear 1: roundness 2: square 3: number 4: Rome

mhrdotsize: 3/6/9 time point size, 0-8

mhrdotcolor: 3/6/9 time point color; Decimalism of transferring color

hrdotstyle: The time point type except 3/6/9, 0: linear 1: roundness 2: square 3: number 4: Rome

hrdotsize: The time point type except 3/6/9, 0-8

hrdotcolor: The time point color except 3/6/9; Decimalism for transferring color

Mindotstyle: The type of minute point, 0: linear 1: roundness 2: square

mindotsize: The size of minute point, 0-1

mindotcolor: The color of minute point; Decimalism for transferring color

hrhandsize: The size of hour point, 0-8

hrhandcolor: The color of hour point; Decimalism for transferring color

minhandsize: The size of minute point, 0-8

minhandcolor: The color of minute point; Decimalism for transferring color

sechandsize: The size of seconds point, 0-8

sechandcolor: The color of seconds point; Decimalism for transferring color

nAutoset: Place set of self-adapt, 0: no 1: yes And if yes, then the coordinate set of txtleft/txttop/ weekleft/weektop/timeleft/timetop should be by yourselves

btxtcontent: Fixed words information

btxtfontname: Fixed words font
 btimefontname: Time words font
 bweekfontname: Week words font
 bclockfontname: Dial words font

Return value: Integer

RETURN_NOERROR: Return without error
 RETURN_ERROR_AERETYPE: Area type error when add/delete lunar area files
 RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen in system; then users could add screen by AddScreen
 RETURN_ERROR_NOFIND_PROGRAM: Do not find the program in system; then users could add the program by AddScreenProgram
 RETURN_ERROR_OTHER: Other errors

19) Function GetScreenStatus (nScreenNo, nSendMode:integer): Integer; stdcall;

Parameter function:

Check the current display screen status, and save the parameters in INI pScreenStatusFile of AddScreen.

nScreenNo: From number one

nSendMode: Communication mode: serial port: 0; Ethernet port: 2;

Note: pScreenStatusFile: files construction

Name	Description	Value
[1]	The value in "[1]" is the display screen number	Value
On/Off	On/Off status of current screen	Power off Power on
Timer for switch state	Automatically On/Off of screen	Invalid Valid
Brightness adjust mode	Brightness adjust mode on screen	Timer adjustment Auto adjustment Manual adjustment

Current brightness level	Current brightness level on screen	Value
Programs number	Quantity of the programs on screen	Value
Current program	Current playing program on screen	Default by system
Lock screen status	Lock screen status	Invalid Valid
Lock program status	Lock program status	Invalid Valid
RTC	Clock status	Abnormal Normal
RTC datetime	Current time	2011-12-02 15:36:31,Fri.
Temperature sensor 1	Temperature sensor 1 value	If controller supports: Temperature value+°C; If do not support: None
Temperature sensor 2	Temperature sensor 2 value	If controller supports: Temperature value+°C; If do not support: None
Humidity sensor	Humidity sensor value	If controller supports: Humidity value+%RH; If do not support: None
Noise sensor	Noise sensor value	If controller supports: Noise value+%dB; If do not support: None

Return value: Integer

RETURN_NOERROR: Return without error

0x01~0x18, 0xFE: Communication error

RETURN_ERROR_NOFIND_AREA: If users do not find valid area, then could add area information by AddScreenProgramBmpTextArea

RETURN_ERROR_NOFIND_PROGRAM: Do not find valid display screen program; then could add the program by AddScreenProgram

RETURN_ERROR_NOFIND_SCREENNO: Do not find the screen, then could add screen by AddScreen

RETURN_ERROR_NOW_SENDING: It's communicating now, please try again later

RETURN_ERROR_OTHER: Other errors