AGN DISPLAY USER MANUAL

Orient Display

Version	Date	Note
V1.0	2018.3.20	The first release
V1.1	2020.11.30	SysConfig Register Configuration Information (New HDS Functionality)
V1.2	2021.10.26	Adding Center Alignment Feature to Artistic Text Variable
V1.3	2023.7.10	Recommended Value for the Number of Images Stored in Flash

Contents

1   Basic Definitions

1.1 **Words**

Words	Description	Note
AGN Display	TFT LCD Module with UART command/configuration interface	
X	Horizontal direction (X-axis) parameter; two bytes	
Y	Vertical direction (Y-axis) parameter; two bytes	
Xs	Starting point horizontal direction (X-axis) parameter, two bytes	"_Area" is used to represent
Ys	Starting point vertical direction (Y axis) parameter, two bytes	
Xe	Ending point horizontal direction (X-axis) parameter, two bytes	
Ye	Ending point vertical direction (Y axis) parameter, two bytes	
TX	Received data to AGN Display	
RX	Sent data from AGN Display	

Note:
The range of values in the horizontal direction (X-axis) and vertical direction (Y axis) in this document, please refer to the specifications for the relevant model.
AGN Display picture index parameters (PicID) are two bytes (default)
1 byte = 8 bits
2 bytes = 16 bits
For example: AGN Display AGN800480A00 Indicates that the display resolution is 800x480
Parameter range for horizontal (X-axis) direction: 0 ~ 799 (0x00 00 ~ 0x03 1F)
Parameter range for vertical (Y-axis) direction: 0 ~ 479 (0x00 00 ~ 0x01 DF)

1.2 **Data Type**

Data Type	Range
Integer	-32768[0x8000] ~ +32767[0x7FFF]
Unsigned Integer	0[0x0000] ~ 65535[0xFFFF]
Long Integer	-2147483648[0x80000000] ~ +2147483647[0x7FFFFFFF]
Extra Long Integer	-9223372036854775808 ~ +9223372036854775807

Note: Decimal numbers are represented using fixed-point notation with user-defined decimal places. For example, if we have the value 0x4D2 (1234) and specify a decimal place count of 2, it represents the decimal number 12.34.

1.3 **Color Data Format**
The color data has a bit width of 16 bits, which is equivalent to 2 bytes. It can display 216-color levels, or 65,536-color levels in total. The format for representing colors is Red(5 bits) – Green(6 bits) – Blue(5 bits). This means:

The highest 5 bits represent the Red component.
The middle 6 bits represent the Green component.
The lowest 5 bits represent the Blue component.
This format allows for the specification of colors using varying levels of intensity for each of the Red, Green, and Blue color channels, resulting in a wide range of available colors for display.

Data frame structure as below:

Bit	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

Color	RED	Green	Blue

Definition:

Color	RED	Green	Blue	White	Black
Data	0xF800	0x07E0	0x001F	0xFFFF	0x0000

2   UART Interface

The UART interface is used for data transfer and configuration.

2.1 UART Protocol

Asynchronous/Full duplex UART works in mode 8N1. Each byte consists of 10 bits,

➢ 1 start bit
➢ 8 data bits, always LSB (Least Significant Bit) is sent first
➢ 1 stop bit
➢ No parity bit

The communicate data rate can be configured in MicroSD/TF memory and the default value is 115200bps.

All instructions or data are in hexadecimal (HEX) format, and for double-byte or multi-byte data, they are always sent in the high-byte-first, low-byte-second order. For example, if you need to set the background color of the serial screen to red, which is represented in hexadecimal (HEX) as 0xF800, the data sent to the serial screen will be transmitted in the following order: 0xF8 0x00.

2.2 Data Frame

Data Frame is used in communication which consists of Header, Data length, Command, Data and CRC (cyclic redundancy check) bytes. The frame structure is shown as:

Frame	Header	Data length	Command code	Data	CRC value
Length (Byte)	2	1	1	N	2
Note	Defined by R3/RA in configure file (CONFIG.TXT)	The length Include Command, Data and CRC	0x80 ~ 0x84		Enabled/Disabled by R2.4 in configure file (CONFIG.TXT)

The maximum valid data length of a frame is 254 bytes (with CRC) or 252 bytes (without CRC). CRC check does not include header byte and data length byte, only for command byte and data by CRC-16 Algorithm (Polynomial representations is $X^{16}+X^{15}+X^2+1$, initial value is 0xFFFF). When CRC check is enabled (R2.4 and RC.3 are set), AGN Display will automatically return CRC results, the acknowledge frame is as below:

Status	Data Frame

CRC correct	Header+0x02+command code+ 0xFF + CRC value
CRC incorrect	Header+0x02+command code+ 0x00 + CRC value

2.3 **CRC-16 reference code (C language)**

```c
static const uint16_t crc16L [] =
{
    0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241,
    0xC601, 0x06C0, 0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440,
};
static const uint16_t crc16H [] =
{
    0x0000, 0xCC01, 0xD801, 0x1400, 0xF001, 0x3C00, 0x2800, 0xE401,
    0xA001, 0x6C00, 0x7800, 0xB401, 0x5000, 0x9C01, 0x8801, 0x4400,
};
uint16_t CRC16(uint16_t BraekPoint, uint8_t *Buffer, uint32_t Length)
{
    uint16_t CRCValue;
    uint8_t Dat;
    CRCValue = BraekPoint;
    While (Length--)
    {
        Dat = *Buffer++;
        Dat^=CRCValue;
        CRCValue >>=8;
        CRCValue ^= crc16L [Dat & 0x0F];
        CRCValue ^= crc16H[(Dat>>4) & 0x0F];
    }
    return CRCValue;
}
```

3   Command

AGN Display controls functions and status according to the value of the registers. The user can set the value of the register by the commands.

Command list

Command	Code	Data
Write Register	0x80	TX: Address (0x00 ~ 0xFF) + data
Read Register	0x81	TX: Address (0x00 ~ 0xFF) + data length (0x00 ~ 0xFF)
		RX: Address (0x00 ~ 0xFF) + data length (0x00 ~ 0xFF) + data
Write Variable Table	0x82	TX: Address (0x0000 ~ 0x6FFF) + data
	0x83	TX: Address (0x0000 ~ 0x6FFF) + data length (0x00 ~ 0xFF)

Read Variable Table		RX: Address (0x0000 ~ 0x6FFF) + data length (0x00 ~ 0xFF) + data
Write Curve Buffer	0x84	Mode + Data0 + …… + Datan Mode: Channel order definition Parameter Mode is 8-bit data which defines the channel order, each bit corresponds one channel, Mode.0 corresponds to channel 0 Mode.1 corresponds to channel 1 … … Mode.7 corresponds to channel 7 When the bit is set to "1", the data should be written into the corresponding channel. When the bit is set to "0", no data will be written into the corresponding channel. Data is assembled from low to high in order of channel number. For example: When Mode is 0x45(01000101B), the data should be written into channel0, channel2 and channel6, so the data assembling sequence should be (channel0+ channel2+ channel6) + (channel0+ channel2+ channel6) +……+ (channel0+ channel2+ channel6) Datan: 2-byte curve data

Note: AGN Display provides
256 Bytes registers which addressing in Bytes.
A 28K-word (56K Bytes) Variable table which addressing in words.
8K-word buffer to store the data which are up to 8 curves. The data is saved as 16-bit unsigned integers.

3.1 Configuration Register List

Address	Register	R/W	Length (Byte)	Note
0x00	Version	R	1	Version Number, BCD code
0x01	LedLm	R/W	1	LCD backlight brightness, 0x00 ~ 0x3F
0x02	Reserved	W	1	Reserved
0x03	PicIndex	R/W	2	Read: index ID of current display page
				Write: Pre-switch to the index number of the specified page
0x05	TPFlag	R/W	1	Touch panel flag. After reading, touch data will not be updated until this flag is cleared.
				0x5A: update coordinates of touch
				other: not update coordinates of touch
0x06	TPStatus	R	1	0x01: the first tap
				0x02: tap off
				0x03: tap continue
				other: invalid
0x07	TPPosition	R	4	Coordinates of touch: Xh, Xl, Yh, Yl, Xh: high byte of X value in coordinate, Xl: low byte of X value in coordinate, Yh: high byte of Y value in coordinate, Yl: low byte of Y value in coordinate
0x0B	TPCEnable	R/W	1	0x00: Disable Touch function
				other: Enable Touch function, default value is 0xFF
0x0C	RunTime	R	4	Running time after power-on, BCD code, data format: HHHH: MM: SS Example: 9999: 59: 59
0x10	R0 ~ RC	R	13	Mapping of configuration registers of SD card, read only
0x1D	Config_Enable	W	1	Flag of R1~ RC register reset

				0x5A: R1 ~ RC reset and save
				0xA5: R1 ~ RC reset and do not save
0x1E	LedLm_Now	R	1	Current backlight brightness
0x1F	RtcComAdj	W	1	0x5A: RTC setting flag, cleared after RTC is set
0x20	RtcNow	R/W	7	YY: MM: DD: WW: HH: MM: SS
0x27	Reserved	—	25	Reserved
0x40	EnLibOP	R/W	1	0x5A: Enable Font Library operation, cleared after execution
0x41	LibOPMode	W	1	0xA0: copy the specified data in font library space to variable space
0x42	LibID	W	1	Index ID of font library, value range: 0x40 ~ 0x7F
0x43	LiaAddress	W	3	The first address of the data block in font library space, value range: 0x000000 ~ 0x01FFFFF
0x46	VP	W	2	The first address of the data block in variable space, value range: 0x0000 ~ 0x6FFF
0x48	OPLength	W	2	The length of data to be operated, value range: 0x0001 ~ 0x6FFF
0x4A	Timer0	R/W	2	16-bit software timer	Software Timer0~3, Unit: 4ms, automatically reduced to zero and stop
0x4C	Timer1	R/W	1	8-bit software timer	
0x4D	Timer2	R/W	1	8-bit software timer	
0x4E	Timer3	R/W	1	8-bit software timer	
0x4F	KeyCode	W	1	Touch key code for triggering file 13*.bin Value range: 0x01 ~ 0xFF, 0x00 means invalid. This register will be cleared after execution.
0x50	Reserved	W	3	Reserved
0x53	VolumeAdj	W	2	Volume adjustment. Format: 0x5A VOL 0x5A: Enable Volume adjustment VOL: Volume = VOL/64, default value of VOL is 64。
0x55	Reserved	—	1	undefined
0x56	EnDBLOP	R/W	1	0x5A: Enable font library operations, cleared after execution
0x57	OPMode	W	1	0x50: copy data from variable space to database space 0xA0: copy data from database space to variable space
0x58	DBLAddress	W	4	Database address (in word) is 0x00000000 ~ 0x01C1FFFF, maximum database space is 450MW (900MB, depends on Flash in Kernel). The database starts from the physical address of 64MB and coincides with the picture space, with each data occupying 2 bytes of physical storage.
0x5C	VP	W	2	The first address (in word) of the specified data block in variable space, range: 0x0000 ~ 0x6FFF
0x5E	OPLength	W	2	Data length (in word), range 0x0001 ~ 0x6FFF
0x60	Reserved	—	138	Reserved
0xEA	TPCalTrigger	W	1	Write 0x5A to enable touch screen calibration and be cleared after calibration.
0xEB	TrendlineClear	W	1	Writes a specific value to clear the corresponding curve buffer data. 0x55: Clear buffer data for all 8 curves 0x56 ~ 0x5D: Clear curve buffer data for CH0-CH7 channels, respectively When the curve buffer data is cleared, the register is cleared
0xEC-0xED	Reserved	—	2	Reserved
0xEE-0xEF	RstSystem	W	2	Write 0x5AA5 to reset the system once
0xF0-0xFF	Reserved	—	16	Reserved

3.2 **Description of Command**

The default frame header information for this communication protocol is set as 0x5AA5 (this frame header marker can be modified through the R3 and RA settings in the CONFIG.TXT file). The data format is also defaulted to hexadecimal (HEX) format.

3.2.1 Write Register (0x80)

Format	TX:	5AA5 Length 80 Addr Data
	RX:	NA
Note	Length	1 Byte, data length, include command code, Addr and Data
	Addr	1 Byte, register address
	Data	Data in byte

Example: to make the buzzer of the PreLet Smart Screen beep for 100ms, you would send the following command to the smart screen: 5AA50380020A.

Data	Note
5AA5	Frame Header
03	Predefined data length (frame data length).
80	Write register command
02	Register address
0A	Data

3.2.2 Read Register (0x81)

Format	Tx	5AA5 Length0 81 Addr Length1
	Rx	5AA5 Length2 81 Addr Length1 Data
Note	Length0	1 byte, data length, include command code, Addr and Data
	Length1	1 byte, required return data length
	Length2	1 byte, data length, include command code, Addr and Data
	Addr	1 Byte, register address
	Data	Data in byte

Example: Get the index number of the currently displayed picture
Send command (5AA5 03 81 03 02) to AGN Display and get the command (5AA5 05 81 03 02 0003) back.

Data	Note
5AA5	Frame Header
03/05	Frame data length (include command code byte and register address byte)
81	Command code
03	Register address
02	Data length
0003	Return data

3.2.3 Write Variable Table (0x82)

Format	Tx	5AA5 Length 82 Addr Data
	Rx	NA
Note	Length	1 byte, data length, include command code, Addr and Data
	Addr	2 Byte, start address in variable table
	Data	Data in byte

Example: write 0x000A into variable table (address: 0x0000), send command: 5AA5 05 82 0000 000A

Data	Note
5AA5	Frame Header
05	Frame data length (include command code byte and register address byte)
82	Command code
0000	Variable table address
000A	Data in byte

3.2.4 Read Variable Table (0x83)

Format	Tx	5AA5 Length0 83 Addr Length1
	Rx	5AA5 Length2 83 Addr Length1 Data
Note	Length0	1 byte, data length, include command code, Addr and Data
	Length1	1 byte, required return data length
	Length2	1 byte, data length, include command code, Addr and Data
	Addr	2 Byte, start address in variable table
	Data	Data in word

Example: Read 1 word from variable table (address: 0x0000)
Send to AGN Display, TX: 5AA5 03 83 0000 01
Receive from AGN Display, RX: 5AA5 06 83 0000 01 000A

Data	Note
5AA5	Frame Header
03/06	Frame data length (include command code byte and register address byte)
83	Command code
0000	Variable table address
01	Return data length
000A	Data in word

3.2.5 Write Curve Buffer (0x84)

Format	Tx	5AA5 Length 84 mode Data
	Rx	NA
Note	Length	1 byte, data length, include command code, mode and Data
	Mode	Mode: Channel order definition (1 byte) Parameter Mode is 8-bit data which defines the channel order, each bit corresponds one channel, Mode.0 corresponds to channel 0 Mode.1 corresponds to channel 1 … … Mode.7 corresponds to channel 7 When the bit is set to "1", the data should be written into the corresponding channel. When the bit is set to "0", no data will be written into the corresponding channel. Data assembled in sequence of lower number channel first.
	Data	Data in word

Example: Write 0x0001、0x0010 into curve buffer channel1, send command (5AA5 06 84 01 0001 0010) to AGN Display.

Data	Note
5AA5	Frame Header
06	Frame data length (include command code byte and mode byte)
84	Command code
01	Mode byte, mode.0=1, channel1 selected
0001 0010	2 words data

3.3 Description of Register

3.3.1 Version register

Address	Register	R/W	Length (Byte)	Note

0x00	Version	R	1	Version Number, BCD code

AGN Display has no handshake command set. This command can be used for System synchronization. User can send command (5AA5 03 81 00 01, read version register (00) and get 1byte data back) to AGN Display to confirm the LCD status. The AGN Display is ready if the system controller receives correct response (5AA5 04 81 00 01 64, "64" is the version number).

3.3.2 Backlight control register

Address	Register	R/W	Length (Byte)	Note
0x01	LedLm	R/W	1	LCD backlight brightness, 0x00 ~ 0x3F
0x1E	LedLm_Now	R	1	Current backlight brightness

User can change AGN Display backlight brightness by change the value of register (0x01), example command: 5AA5 03 80 01 XX (XX is a data in byte, in range of [0x00, 0x3F]). To get the current backlight brightness, user can use commands as: "5AA5 03 81 01 01" or "5AA5 03 81 1E 01"

3.3.3 Buzzer Control Register

Address	Register	R/W	Length (Byte)	Note
0x02	BzTime	W	1	Buzzer Beeping Control Register, in units of 10ms

AGN Display can produce buzzer alert sounds not only through touchscreen interactions but also by sending command frames to control the buzzer for various audio alerts and prompts. Users can send a command frame like "5AA503800201" to enable the buzzer alert sound functionality. The value "01" in the command frame enables the buzzer to sound, and you mentioned that the value can range from "00" to "FF" for different settings, with "00" indicating that the buzzer should not beep. This allows users to control and customize the buzzer's alert behavior using these command frames.

3.3.4 Touch Panel Control Register

Address	Register	R/W	Length (Byte)	Note
0x05	TPFlag	R/W	1	Touch panel flag. After reading, touch panel data will not be updated until this flag is cleared.
				0x5A: coordinate of touch updated
				other: coordinate of touch not updated
0x06	TPStatus	R	1	0x01: the first tap on
				0x02: tap off
				0x03: tap continue
				other: Reserved
0x07	TPPosition	R	4	Coordinate of touch: Xh, Xl, Yh, Yl, Xh: high byte of X value in coordinate, Xl: high byte of X value in coordinate, Yh: high byte of Y value in coordinate, Yh: high byte of Y value in coordinate
0x0B	TPCEnable	R/W	1	0x00: Disable Touch function
				other: Enable Touch function, Power on value is 0xFF

As cleared (set to 0x00) register (0x0B) will disable all operations of touch panel, it should be very carefully to operate on this register. User can get touch status and coordinate of touch point by reading register (0x06) and register (0x07). Register (0x05) should be cleared after each reading, or the AGN Display will not update the coordinate of touch point.

3.3.4.1 Determine if touch functions are supported

Send command frame "5AA5 03 81 0B 01" to AGN Display, returned data frame "5AA5 04 81 0B 01 00" indicates the touch function is disabled. Otherwise, returned data frame "5AA5 04 81 0B 01 XX (XX is a data in byte, in range of [0x01, 0xFF])" indicates the touch function is enabled and pre-defined touch functions are available.

3.3.4.2 Disable/Enable touch function

Disable touch command frame: 5AA5 03 80 0B 00
Enable touch command frame: 5AA5 03 80 0B XX (XX is a data in byte, in range of [0x01, 0xFF])

3.3.4.3 Read touch status and touch point coordinate values

The user can read the value of the register 0x05/0x06x07 (command: 5A A5 03 81 05 06) to get the touch state and contact point coordinate values of the touch panel. The 0x05 register needs to be cleared (commands: 5A A5 03 80 05 00) before the next read of the contact point coordinate values.

3.3.5 Running Time Register

Address	Register	R/W	Length (Byte)	Note
0x0C	RunTime	R	4	Running time after power-on, BCD code, data format: HHHH:MM:SS Example: 9999: 59: 59

Note: Read running time command frame: 5AA5 03 81 0C 04.

3.3.6 Configuration Register Operation

Address	Register	R/W	Length (Byte)	Note
0x10	R0 ~ RC	R	13	Mapping of TF card configuration registers
0x1D	Config_Enable	W	1	Flag of R1~ RC register reset
				0x5A: R1 ~ RC reset and save
				0xA5: R1 ~ RC reset and do not save

Note: The detail configuration of R0~RC refer to the file CONFIG.TXT.

3.3.6.1 Read configure register information

Send command frame "5AA5 03 81 10 0D" to AGN Display, the return frame will be like "5AA5 10 81 10 0D 03 07 0C 5A FF FF 3F 04 03 FF A5 FF 00", the address of the Register determines the meaning of the data.

3.3.6.2 Temporarily change configuration register information

Example: In screensaver mode, command set "5AA5 03 80 18 20 / 5AA5 03 80 1D A5" can be used to change the screensaver wait time temporarily.

3.3.6.3 Change configuration register information

Example: Command set "5AA5 03 80 12 0B / 5AA5 03 80 1D 5A" can be used to configure the refresh frequency of the screen permanently.

3.3.7 RTC (Real Time Clock) Register

Address	Register	R/W	Length (Byte)	Note
0x1F	RtcComAdj	W	1	0x5A: reset real time clock, cleared after execution
0x20	RtcNow	R/W	7	YY: MM: DD: WW: HH: MM: SS

Note: User can read/modify RTC by UART, the RTC information is saved in registers 0x20 ~ 0x26 in BCD format.

3.3.7.1 Read clock information

Read full data (YY: MM: DD: WW: HH: MM: SS): 5AA5 03 81 20 07
Read time only (HH: MM: SS): 5AA5 03 81 24 03

3.3.7.2 Calibrate RTC
Command: 0x80 (Write Register Command)
Register Address: 0x1F
Register Value: 0x5A
Date and Time Information:
Year: 0x16 (2016)
Month: 0x03 (March)
Day: 0x09 (9th)
Hour: 0x16 (16th hour, or 4:00 PM)
Minute: 0x18 (18th minute)
Second: 0x30 (30th second)
The complete command frame to set the smart screen's clock to "2016-03-09 16:18:30" is
"5AA50A801F5A16030900161830". Only the year, month, day, hour, minute, and second need
to be adjusted to calibrate the clock, as the day of the week information will be automatically
corrected by the smart screen.

3.3.8 Font Library Register

Address	Register	R/W	Length (Byte)	Note
0x40	EnLibOP	R/W	1	0x5A: Enable Font Library, cleared after execution
0x41	LibOPMode	W	1	0xA0: copy the specified font library to variable table
0x42	LibID	W	1	Index of font library, value range: 0x40 ~ 0x7F
0x43	LiaAddress	W	3	The first address of the font library to be read, value range: 0x000000 ~ 0x01FFFF
0x46	VP	W	2	The first address of the variable space to be written, value range: 0x0000 ~ 0x6FFF
0x48	OPLength	W	2	The length of data to be copied, value range: 0x0001 ~ 0x6FFF

The font libraries (index ID from 64 to 127, total 64 font libraries, 16MB) can be copied to
variable space by command. And user can read the font library from the variable space by
command 0x83.
Example: Copy font library with index 64, 4KW(0x1000), start address 0x000000 to variable
space with start address 0x1000. The command frame is "5AA5 0C 80 40 5A A0 40 00 00 00 10
00 10 00"
The data to be copied should not exceed the font library space, (LiaAddress + OPLength) ≤
0x020000

3.3.9 Key Code Control Register

Address	Register	R/W	Length (Byte)	Note

0x4F	KeyCode	W	1	Touch key code for triggering file 13.bin. Value range: 0x01 ~ 0xFF, 0x00 is invalid value. This register will be cleared after execution.

AGN Display provide 0x4F (Key code control) Register instead of keyboard interface. User can perform a pre-defined touch process which descripted in (file 13.bin) by writing corresponding key code into register (0x4F).

For example, in touch configure file, key code 0x01 on page 10th means turn to data input interface, when the command frame "5AA5 03 80 4F 01" is sent on page 10th, the AGN Display will turn to data input interface.

Key-code triggers and touch-screen triggers can be triggered in parallel, so they can be together.

3.3.10 Database Register

A database is a continuous storage area in the picture storage area of AGN Display, and the size and location of the space can be defined by the user.

Address	Register	R/W	Length (Byte)	Note
0x56	EnDBLOP	R/W	1	0x5A: Enable font library operations, cleared after execution
0x57	OPMode	W	1	0x50: copy variable table to data table 0xA0: copy data table to variable table
0x58	DBLAddress	W	4	Database address is 0x00000000 ~ 0x01C1FFFF, maximum database space is 450MW (900MB, depends on Flash in Kernel). The database starts from the physical address of 64MB and coincides with the image memory space, with each data occupying 2 bytes of physical memory.
0x5C	VP	W	2	Specifies the first (word) address of the database operation for the variable memory space, range: 0x0000 ~ 0x6FFF
0x5E	OPLength	W	2	Operation Data length, range 0x0001 ~ 0x6FFF

With different flash storage space, the picture space and the available database space has different sizes which listed in the table below:

Picture storage (MB)	Database storage (MB)	Max picture number with different flash					
		320x240	480x272	640x480	800x480	800x600	1024x600
Version A = 128	51.75	596	351	149	119	95	74
Version B = 256	179.5	1415	832	353	283	226	176

The user database is physically composed of several database pages of 64KW (128KB) in size, but the address in the read/write operation is continuous, the system automatically handles paging situations. Each page writes a lifetime of 100, 000 times, that is, each write operation reduces the number of times once.

As the database space shared the storage space with picture storage space, the address offset should be calculated to avoid data overlap. Image index number corresponding to the physical address (64MB) of the database space as below:

Resolution		320x480	480x272	640x480	800x480	800x600	1024x600	1024x768
Adjustment coefficient (K)	VersionA	2	2	5	6	8	10	12
	VersionB	1	1	3	3	4	5	6
	VersionA	126	126	50 ~ 51	42	31 ~ 32	25 ~ 26	21

The number of pictures	VersionB	124	124	41 ~ 42	41 ~ 42	31	24 ~ 25	20 ~ 21

Note:

"124" represents the number of reserved images in an unused database area when the database address is set to 0. This value also indicates that starting from this image, any operations in the database area may damage image information.

"50 ~ 51" represents that when the data operation offset address is set to 0, performing operations in this database area may damage image information.

When using the database for data access, it's important to avoid overlapping with image pages or to skip image page numbers within the database storage area.

The database offset address calculation formula is shown below (in words):

$A = (Q*K-M) *T*C$ when $(Q*K) > M$

$A = 0$ when $(Q*K) ≤M$

A: Represents a relative offset address (in word), that is, the length of the address that needs to be offset from the start address of the database

Q: The total number of pre-saved pictures

M: Picture quantity adjustment coefficient (constant value). Adjustment coefficient of Version A is 252, Adjustment coefficient of Version B is 124.

K: Adjustment coefficient

T: Correction coefficient, the correction coefficient of version A is 1, the correction coefficient of version B is 2.

C: Constant, 64*1024

Example 1: 100 pictures with resolution 800x480 saved in AGN Display (Part No. AGN800480A00-7.0N12NSM-R), system version is A, Q = 100, M = 252, K = 6, T = 1, then address offset of the database operation is (100*6-252) *1*64*1024 = 22806528[0x015C0000]

Example 2: 200 pictures with resolution 800x600 saved in AGN Display (Part No. AGN800480A00-7.0N12NSM-R), system version is B, Q = 200, M = 124, K = 4, T = 2, then address offset of the database operation is (200*4-124) *2*6481024 = 88604672[0x05480000]

4 MicroSD/TF Card Download Description

All parameter settings and data of AGN Display need to be downloaded by MicroSD /TF card which support FAT32 file system. AGN Display SD Card interface only supports MicroSD card, that is, TF card. To prevent mis-operation, the AGN Display has strict naming and formatting requirements for the MICROSD/TF card configure file.

File Type	Naming Rule	Sample	Note
Picture File	Index number+ [file name].bmp	00 Boot screen.bmp	24-bit color BMP file with the same resolution as the AGN Display
Font Library File	Font Library Index+ [file name].bin /hzk/dzk/ico	32_GBK12 hanziku.dzk 13 touch control.bin 0_ODASCII.hzk	32MB for font library space, 256KB for each font library, total 128 font libraries. If the font library needs to occupy 3.1 font space, it is recommended that the font index number be extended
Configure File	CONFIG.TXT		

4.1 File download operation

1. Create a folder named as OD_GUI under the card root directory
2. Copy pictures, font libraries, and configuration files that need to be downloaded into OD_GUI folder
3. Power off the AGN Display and insert the MicroSD (TF) card;
4. Power on the AGN Display and it will automatically load the contents of the OD_GUI folder, and save to the AGN Display device;

5. The user can power off the AGN Display and unplug the MicroSD (TF) card and re-power on the AGN Display to go back to normal mode.

4.2 **Picture File**

24-bit color BMP file with the same resolution as the AGN Display and file name start by index number.
Example: To update 15th picture through TF card. Picture file name can be "15.BMP"/ "15_xx.BMP"/ "15 xx.BMP"/ "015 xx.BMP", but it can not be named as "xx 15.BMP". Copy this file into TF card and update it by steps described in 4.1 File download operation.
Note: When the smart screen power on, the picture (ID=0) is displayed first, so it is recommended that the picture arrangement order starts from 0.

4.3 **Font Library File**

AGN Display support total 32MB font library space and divided into 128 fixed-size sub-space equally with 256KB for each. File name should begin with the index number (0 ~127) also. A font library file set includes font library file, configure file and icon file.

File Type	Naming Rule	Example
Font library file	Location index + file name[optional].BIN/HZK/DZK	36 GBK12.DZK
Icon file	Location index + file name[optional].ICO	25 Iconlib.ICO
ASCII font library file	0*.HZK	0_OD_ASC.HZK
Touch control configure file	13*.BIN	13 touchcontrol.BIN
Variable configure file	14*.BIN	14 varconfig.BIN
Variable initial file	22*.BIN	22 varini.BIN
Index numbers (0~22) are system reserved, index number which greater than or equal to 23 is available.		

4.4 **Configure File (CONFIG.TXT)**

The configure file (CONFIG.TXT) configures a parameter on each line that is not written if it is unused.

Register	Value Range	Note
R0	LcdID	LCD ID number, system reserved
R1	0x00 – 0x12	UART communication rate setting, refer to Appendix II
R5	0x00 ~ 0xFF	R5: high byte of communication rate setting
R9	0x00 ~ 0xFF	R9: low byte of communication rate setting when R1=0xFE, R5: R9 is used to configure UART Baud rate (the frequency division of 6250000bps)
R2	0x00 ~ 0xFF	SysConfig register, refer to Appendix III
RC	0x00 ~ 0xFF	AuxConfig register, refer to Appendix IV
R3	0x00 ~ 0xFF	UART_SYNC_H, data frame high byte
RA	0x00 ~ 0xFF	UART_SYNC_L, data frame low byte
R6	0x00 ~ 0x3F	After backlight control function enabled, the brightness of the backlight that can be lighted by touch
R7	0x00 ~ 0x3F	After backlight control function enabled, the brightness of the backlight that shut down after a period of time during which there is not any touch operation.
R8	0x01 ~ 0xFF	After backlight control function enabled, lighting time of the backlight, unit: 1s
RD	0x7F/0xFF	Hardware configuration register
Note1: The parameter in configure file is one-byte hexadecimal number, for example: "0C" means "12" in decimal		
Note2: The parameter in configure file should be a two-digit number, for example: zero should be "00" and not "0"		

4.4.1 LCD ID Number Register

AGN Display support multi-resolution from 320x240 to 1024x768, which selected by setting register R0. The parameter of R0 is set in factory which need not configure again and improper configuration may cause display error.

R0 Register value (Index)	LCD Resolution	Note
28	240x320	2.8′ ,
20	320x240	3.5′ ,
22	480x272	4.3′ 、5.0′
01	640x480	5.6′ ,
02	800x480	5.0′ , 7.0′
03	800x600	8.0′ , 10.4′
0B	1024x600	10.1′ ,

4.4.2 Communication Rate Configuration register

Register	Value Range	Note	
R1	0x00 – 0x12	UART communication rate setting, refer to Appendix II	
R5	0x00 ~ 0xFF	R5: high byte of Baud rate setting	when R1=0xFE, R5: R9 is used to configure UART Baud rate (the frequency division of 6250000bps)
R9	0x00 ~ 0xFF	R9: low byte of Baud rate setting	

AGN Display supports three modes of UART communication rate configuration.

When the R1 value is between 0x00 and 0x12, each value corresponds to a fixed UART Baud rate, refer to Appendix II.

When R1=0xFE, UART Baud rate will decided by values in registers R5 and R9. Baud rate = $[6250000 \div (R5 * 256 + R9)]\ bps$. If R5=0, the value of R9 should greater than 0. R9 can select any value that include 0, when R5 is set non-zero value.

The default UART Baud rate is 115200bps, when R1 value is not equal to 0x00 ~ 0x12、0xFE.

4.4.3 Screensaver Register

When R2.5 is set, backlight brightness is controlled by touch operation (Note: the first touch operation does not trigger any action when backlight standby).

Register	Range	Note
R6	0x00 – 0x3F	After backlight control function enabled, the brightness of the backlight after wakeup
R7	0x00 – 0x3F	After backlight control function enabled, the brightness of the backlight that standby after a period of time during which there is not any touch operation.
R8	0x01 – 0xFF	After backlight control function enabled, the time waiting to standby, unit: seconds

For example: when R2.5=1, R6=0x3F, R7=0x10, R8=0x1E, after 30 seconds during which there is not any touch operation, the backlight brightness turns to 0x10 automatically. When any touch operation occurred, the backlight brightness turns to 0x3F automatically.

4.4.4 Frame Header Register

Register	Range	Note
R3	0x00 ~ 0xFF	UART_SYNC_H, data frame high byte
RA	0x00 ~ 0xFF	UART_SYNC_L, data frame low byte

The default data frame header is set as 0x5AA5 according to R3(0x5A): RA(0xA5) configured in file CONFIG.TXT which can be loaded from MicroSD/TF card.

4.4.5 Hardware configuration register (0xRD)

AGN Display supports both resistive / capacitive touch panels and they are optional.
The default hardware setting is system reserved and it is not recommended to be changed by user. It is recommended to set the touch panel type to resistive panel if there is no touch panel assembled.

Register Bit	Weight	Description	Note
Bit 7	0x80	R/C TP	1: resistive touch panel
			0: capacitive touch panel
Bit 6	0x40	Reserved	Recommend to set "1"
Bit 5	0x20	Reserved	Recommend to set "1"
Bit 4	0x10	Reserved	Recommend to set "1"
Bit 3	0x08	Reserved	Recommend to set "1"
Bit 2	0x04	Reserved	Recommend to set "1"
Bit 1	0x02	Reserved	Recommend to set "1"
Bit 0	0x01	Reserved	Recommend to set "1"

4.5 Software Update

Note: Our company does not recommend that users use this feature to update the AGN Display software, if you need to update the AGN Display software, please contact our technical support staff.

AGN Display can update software though MicroSD/TF, the workflow as below,
1) Create folder /OD_GUI/BOOT in root folder of MicroSD/TF
2) Copy the files (*.bin) that need to be updated to folder BOOT
3) Power off AGN Display and plug MicroSD/TF card into AGN Display
4) Power on AGN Display again and it will search and update the files in MicroSD/TF. If AGN Display do not find any files, update operation will not be performed

During the software update process, user can get system information of update process by UART interface. The default Baud rate of UART is 115200bps, 8n1 mode (10bits, 1start bit, 8data bits, 1stop bit for each word). Do not shut down power or plug/unplug SD card during the update process.

4.6 Touch Panel Calibration

This function is available only when the touch panel is already assembled.
There are 3 methods to enable the touch panel calibration.

Method 1:
Step1: In 3 seconds, continuous click on the non-touch area more than 20 times
Step2: Get into calibration mode and click the specified location of the touch panel which indicated by the cross lines
Step3: Calibration finished and return back.

Method 2:
Add "TP_CORRECT" into file CONFIG.TXT, AGN Display will start one calibration process. AGN Display automatically verifies that the calibration is valid, the configure will not be set when the verification failed.

Method 3:

In normal mode, writing 0x5A into register (0xEA) though UART will start one calibration process. AGN Display automatically verifies that the calibration is valid. The system will perform calibration operations until the calibration test is effective.

Please be noted that multiple invalid calibrations may cause physical damage to the touch panel, such as line breakage, touchpad damage ...

5   Touch/Keying Features Description

Touch/keying features are configured with touch profiles 13*.bin. The file consists of one or more instructions described in accordance with the touch/key control function, each of which is fixed in 16, 32, or 48 bytes of space; A touch instruction consists of 6 parts.

NO.	Name	Length (Byte)	Note
1	TPID	2	Index ID of function picture. When TPID = 0xFFFF, touch function ended.
2	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye). When Xs = 0xFFFF, the function is triggered by writing key code into register 0x4F and tap effect should be disabled, then Ys_H is key code value and Ys_L / Xe / Ye are undefined.
3	TraID	2	Switch picture index ID；0xFFxx: no switching
4	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be displayed
5	Code	2	Key code 0xFFxx: invalid key code 0xFExx (0xFDxx): function key code. For example, 0xFE00: enable variable screen input function Any change of 0xFExx function key can be upload automatically by setting the R2.3. 0xFDxx function keys are not upload automatically. 0x00xx: key code in ASCII. For example, 0x0031: key "1"
6	TPFun	16/32/48	Description of function key when Code has value of 0xFExx/0xFDxx

Function List Table

No.	Code	Function	Note
1	0x00	Data Input	Input data (integer, decimal, etc.) to variable space.
2	0x01	Menu Action	Click to trigger a pop-up menu and return key code of the menu
3	0x02	Incremental Input	Click the button to make a self-added (++) or self-subtraction (- -) operation on the specified variable. The step length and upper/lower limits can be set. Setup loop adjustment with range 0~1 to achieve column check box function.
4	0x03	Drag Input	Data input by the way of dragging and the scale range is configurable.
5	0x04	Set RTC	Set Real time clock by soft keyboard, data of a full calendar day (year, month, date, hour, minute, second) is needed.
6	0x05	Return Key code	Click the button to return the key value directly to the variable, and support returning variable bit
7	0x06	Text Input	Input text in format of ASCII
8	0x08	Return touch data	Click the touch panel and return pre-set data to variable space or UART.

5.1 Number input (0xFE00/0xFD00)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID；0xFFxx: no switching

0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE00/0xFD00
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data
0x13	VType	1	Return data type 0x00: 2 bytes, Integer: -32768~+32767 unsigned integer: 0~65535 0x01: 4 bytes, long integer -2147483648 ~ +2147483647 unsigned long integer 0 ~ 4294967295 0x02: *VP high byte, unsigned integer, 0 ~ 255 0x03: *VP low byte, unsigned integer, 0 ~ 255 0x04: 8 bytes, super long integer: -9223372036854775808 ~ +9223372036854775807
0x14	NInt	1	Number of integer bits of input data. For data 1234.56, NInt is set 0x04
0x15	NDot	1	Number of decimal bits of input data. For data 1234.56, NDot is set 0x02
0x16	x, y	4	Display location of input: right alignment, (x, y) is the upper-right coordinate of the last character of the string
0x1A	Color	2	Color of the input characters
0x1C	LibID	1	ASCII Font Library index ID of characters displayed
0x1D	FontHor	1	Font size, number of matrix dot in X-direction
0x1E	CusorColor	1	Cursor Color, 0: Black; other: White
0x1F	HideEn	1	0x00: Hide Input, displayed as "*" other: display input character
0x20	0xFE	1	0xFE
0x21	KBSource	1	0x00: Keyboard on the current page other: Keyboard is not in the current page
0x22	KBPicID	2	The index ID of the page on which the keyboard resides. Valid when KBsource is not 0x00
0x24	KBArea	8	Keyboard area: Upper-left coordinate (xs, ys), lower-right coordinate (xe, ye). Valid when KBsource is not 0x00
0x2C	KBPosition	4	The keyboard position on the current page, the upper-left coordinates, and is valid when KBsource is not 0x00
0x30	0xFE	1	0xFE
0x31	LimitEn	1	0xFF: Enable input range limit, overflow input invalid (equivalent cancellation) Other: no input range limit
0x32	Vmin	4	Input lower limit, 4 bytes (Long integer or unsigned long integer)
0x36	Vmax	4	Input upper Limit, 4 bytes (Long integer or unsigned long integer)
0x3A	Reserved	6	Recommend to set "0"

The valid key codes:
0x0030(0)~0x0039(9), 0x002E(.), 0x00F0(cancel), 0x00F1 (enter), 0x00F2 (BackSpace)

5.2 Pop-up Menu (0xFE01/0xFD01)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE01/0xFD01
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data. The returning data is determined by VType

0x13	VType	1	0x00: Write the key code 0x00** to the VP address (integer number)
			0x01: Write the Key code 0x** to the high-byte of the VP address (VP_H)
			0x02: Write the Key code 0x** to the low-byte of the VP address (VP_L)
			0x10-0x1F: Writes the lowest bit (1bit) of the key code 0x** to the specified bit of the data in VP address (0x10 modifies VP.0, 0x1F modifies the VP. F)
0x14	KBPicID	2	The index ID of the page where the keyboard resides
0x16	KBArea	8	Keyboard area: upper-left coordinate (Xs, Ys), lower-right coordinate (Xe, Ye)
0x1E	KBPositionX	2	The keyboard position on the current page, the upper-left coordinates
0x20	0xFE	1	0xFE
0x21	KBPositionY	2	The keyboard position on the current page, the upper-left coordinates
0x23	Reserved	13	Recommend to set "0"

Valid Key code: 0x0000 ~ 0x00FF, where 0x00FF means cancel (return directly without parameter).

5.3 Incremental Input (0XFE02/0XFD02)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE02/0xFD02
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data. The returning data is determined by VType
0x13	VType	1	0x00: Adjust VP address (integer)
			0x01: Adjust the high-byte address of the VP word address (1byte unsigned integer, VP_H)
			0x02: Adjust the low-byte address of the VP word address (1byte unsigned integer, VP_L)
			0x10-0x1F: Adjust the specified bit of the VP address (0x10 corresponds to VP.0, 0x1F corresponding to the VP. F), the adjustment range must be set to 0 ~ 1.
0x14	AdjMode	1	Adjust mode	0x00: --
				other: ++
0x15	ReturnMode	1	Overflow processing	0x00: stop
				other: cyclic adjustment
0x16	AdjStep	2	Step length adjustment, 0x0000 ~ 0x7FFF
0x18	VMin	2	Lower limit: 2 bytes integer (only low bytes valid when VPmode is 0x01 or 0x02)
0x1A	VMax	2	Upper limit: 2 bytes integer (only low bytes valid when VPmode is 0x01 or 0x02)
0x1C	KeyMode	1	0x00: Continuous adjustment when holding down the button
			0x01: Adjust only once when holding down the button
0x1D	NULL	3	Recommend to set "0"

5.4 Drag Input(0xFE03/0xFD03)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).

0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE03/0xFD03
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data. The return data is determined by VType
0x13	VType	1	High 4 bits define the data return format — 0x0*: Adjust data in VP address (integer); 0x1*: Adjust the high-byte of the VP Word address (1byte unsigned integer, VP_H); 0x2*: Adjust the low-byte of the VP Word address (1byte unsigned integer, VP_L). Low 4 bits define the drag mode — 0x*0: Drag horizontally; 0x*1: Drag vertically
0x14	AreaAdj	8	Adjusting area: (Xs, Ys), (Xe, Ye); must be consistent with TPArea (touch area)
0x1C	VBegain	2	Return value corresponding to the start position, integer
0x1E	VEnd	2	Return value corresponding to the end position, integer

5.5 RTC Control (0xFE04/0xFD04)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE04/0xFD04
0x10	0xFE	1	0xFE
0x11	NULL	3	0x00 00 00
0x14	x, y	4	Input display location: Right alignment, (x, y) is the upper-right coordinate of the last character of the string
0x18	Color	2	Color of the input characters
0x1A	LibID	1	ASCII Font Library index ID of characters displayed
0x1B	FontHor	1	Font size, number of matrix dot in X-direction
0x1C	Reserved	1	Recommend to set "0"
0x1D	KBSource	1	0x00: Keyboard on the current page; other: Keyboard is not on the current page
0x1E	KBPicID	2	The index ID of the page on which the keyboard resides which is valid when KBsource is not 0x00
0x20	0xFE	1	0xFE
0x21	KBArea	8	Keyboard area: upper-left coordinate (Xs, Ys), lower-right coordinate (Xe, Ye) which is valid when KBsource is not 0x00
0x29	KBPosition	4	The keyboard position on the current page, the upper-left coordinates which is valid when KBsource is not 0x00
0x2D	NULL	3	Recommend to set "0"

5.6 Key Value (0XFE05/0XFD05)

Address	Definition	Length (byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE05/0xFD05
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data.

0x13	VType	1	The return data is determined by VType
			0x00: Key code saved in VP address (integer)
			0x01: The low byte of the key code is saved at the high-byte of the VP word address (VP_H)
			0x02: The low byte of the key code is saved at the low-byte of the VP word address (VP_L)
0x14	Code	2	The key code returned
0x16	NULL	10	Recommend to set "0"

5.7 Text Input(0xFE06/0xFD06)

The typical text key code definition is shown as below:

Key code	Lower case	Capital	Key code	Lower case	Capital	Key code	Lower case	Capital	Key code	Lower case	Capital
7E60	`	~	5171	q	Q	4161	a	A	5A7A	z	Z
2131	1	!	5777	w	W	5373	s	S	5878	x	X
4032	2	@	4565	e	E	4464	d	D	4363	c	C
2333	3	#	5272	r	R	4666	f	F	5676	v	V
2434	4	$	5474	t	T	4767	g	G	4262	b	B
2535	5	%	5979	y	Y	4868	h	H	4E6E	n	N
5E36	6	^	5575	u	U	4A6A	j	J	4D6D	m	M
2637	7	&	4969	i	I	4B6B	k	K	3C2C	,	<
2A38	8	*	4F6F	o	O	4C6C	l	L	3E2E	.	>
2839	9	(	5070	p	P	3A3B	;	:	3F2F	/	?
2930	0	)	7B5B	[	{	2227	'	"	2020	SP	SP
5F2D	-	_	7D5D	]	}	0D0D	Enter	Enter			
2B3D	=	+	7C5C	\								

Note: For the key code in table, the low byte of two-byte key code represents a lowercase key code, and the high byte represents a capital key code. The key code of a text keyboard must be less than 0x80 (ASCII code). 0x0D Key code entry will be automatically converted to 0x0D 0x0A; Key codes 0x00 and 0xFF are disabled.

Special features keyboard key code definition

Key Code	Definition	Note
0x00F0	Cancel	Cancel input and return, do not change variables
0x00F1	Return	Confirm input and return, input text is saved to the specified variable position
0x00F2	Backspace	Delete 1character forward
0x00F3	Delete	Delete 1 character backward
0x00F4	CapsLock	Uppercase lock. If enabled, the corresponding button must define the effect of button click
0x00F7	Left	The cursor moves a character forward and is also used for paging in GBK Chinese character entry
0x00F8	Right	The cursor moves a character backward and is also used for paging in GBK Chinese character entry

When using the keyboard (the key code saved in the register 0x4F) to do text entry, define the animation area of the button in the area where you need to prompt "CapsLock", so that when you send the CapsLock key, the corresponding position of the screen will automatically display the "CapsLock" area icon prompt.

5.7.1 ASCII Input

Address	Definition	Length (Byte)	Note
0x00	TPID	2	Index ID of function picture

0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE00/0xFD00
0x10	0xFE	1	0xFE
0x11	*VP	2	Variable address pointer which corresponds to the input data.
0x13	VPLenMax	1	The maximum length of the text variable, the range of the number of words: 0x01 ~ 0x7B. When the text is saved to the specified address, 0xFFFF is added at the end of the text automatically. The input text variable may actually occupy the maximum variable space as VPLenMax +1.
0x14	ScanMode0	1	Input Mode control
			0x00	Re-input
			0x01	Open the original text and modify it
0x15	LibID	1	Display the current ASCII font library location, 0x00 = default font library
0x16	FontHor	1	Font size, number of matrix dot in X-direction
0x17	FontVer	1	Font size, number of matrix dot in Y-direction (when LibID =0x00, number of matrix dot in Y-direction must be doubled)
0x18	CusorColor	1	Cursor Color
			0	Black
			Other	White
0x19	Color	2	Text Display color
0x1B	ScanAreaStart	4	Input text display area left upper corner coordinates (Xs, Ys)
0x1F	ScanReturnMode	1	0X55: Save the input end tag and the valid data length at the * (VP-1) Location	* (VP-1) High byte, 0x5A: end tag, 0x00: processing tag
				* (VP-1) Low byte, valid input data lengths, one byte
			0x00: Do not return input end tags and lengths	
0x20	0xFE	1	0xFE
0x21	ScanAreaEnd	4	Input text display area lower-right corner coordinates (Xe, Ye)
0x25	KBSource	1	0x00: Keyboard on the current page
			Other: Keyboard is not on the current page
0x26	KBPicID	2	The index ID of the page on which the keyboard resides. Valid when KBSource is not 0x00
0x28	KBArea	8	Keyboard area: Upper-left coordinate (Xs, Ys), lower-right coordinate (Xe, Ye). Valid when KBSource is not 0x00
0x30	0xFE	1	0xFE
0x31	KBPosition	4	The keyboard position on the current page, the upper-left coordinates, and is valid when KBSource is not 0x00
0x35	DisplayEn	1	0x00: The input displayed normally
			0x01: Display "*" for password input
0x36	Reserved	9	Recommend to set "0"
0x3F	ScanMode1	1	0x00 (ASCII)

Note: AGN Display pre-installed font library (ID=0) which contains 4*8 ~ 64*128 dot matrix of all ASCII pattern.

5.7.2 GBK Input

Address	Definition	Length (Byte)	Note
0x00	TPID	2	Index ID of function picture
0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID; 0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be displayed
0x0E	Code	2	0xFE00/0xFD00
0x10	0xFE	1	0xFE

Address	Definition	Length (Byte)	Note			
0x11	*VP	2	Variable address pointer which corresponds to the input data.			
0x13	VPLenMax	1	The maximum length of the text variable, the range of the number of words: 0x01 ~ 0x7B. When the text is saved to the specified address, 0xFFFF is added at the end of the text automatically. The input text variable may actually occupy the maximum variable space as VPLenMax +1.			
0x14	ScanMode0	1	Input Mode control 0x00: Re-input 0x01: Open an existing text and modify it			
0x15	LibGbk0	1	The GBK font library index ID, default ASCII Font ID is 0x00			
0x16	LibGbk1	1	The index ID of GBK font library used in input process			
0x17	FontDot0	1	LibGbk0: Font size, number of matrix dot			
0x18	FontDot1	1	LibGbk1: Font size, number of matrix dot			
0x19	CusorColor	1	Color of cursor. 0x00: Black Other: White			
0x1A	Color0	2	Color of text entered			
0x1C	Color1	2	Color of text in the input			
0x1E	PyDisMode	1	Pinyin hints and the display mode of Chinese characters			
			0x00	Pinyin tips on the top, Chinese characters show another line below. Pinyin tips and Chinese characters show left alignment, line spacing is ScanDis.		
			0x01	Pinyin tips on the left, Chinese characters on the right; Chinese characters begin to show X position in Scan1AreaStart + 3×FontDot1 +ScanDis		
0x1F	ScanReturnMode	1	0xAA: Save the input end tag an the valid data length at the *(VP-1) Location	*(VP-1) Low bytes, valid input data length, in byte.		
				*(VP-1) High bytes, entry status flag	0x5A	Input end
					0x00	In input process
			0xFF: Do not return input end tags and lengths			
0x20	0xFE	1	0xFE			
0x21	Scan0AreaStart	4	Input text display area upper-left corner coordinates (Xs, Ys)			
0x25	Scan0AreaEnd	4	Input text display area lower-right corner coordinates (Xe, Ye)			
0x29	Scan1AreaStart	4	The upper-left coordinate of the text display area of the voice prompt during the input process			
0x2D	ScanDis	1	The spacing of Chinese characters is shown in the input process, up to 8 Chinese characters per line			
0x2E	Reserved	1	Recommend to set "0x00"			
0x2F	KbSource	1	Keyboard location selection. 0x00: Keyboard on the current page; other: Keyboard not on current page			
0x30	0xFE	1	0xFE			
0x31	PicKb	2	The following data is valid only if Kbsource is not 0x00. The index ID of the page where the keyboard resides			
0x33	AreaKb	8	Keyboard area coordinates: Upper-left corner (Xs, Ys) Lower-right corner (Xe, Ye)			
0x3B	AreaKbPosition	4	the location of keyboard area which pasted on the current page, the upper-left coordinates.			
0x3F	ScanMode1	1	0x02(Pinyin Input Method)			

Note: Pinyin "bd" corresponds to all GBK encoded fullwidth punctuation marks
AGN Display pre-installed font library (ID=0) which contains 4*8 ~ 64*128 dot matrix of all ASCII pattern.

5.8 **Touch Synchronous Data Return (0xFE08/0xFD08)**

Address	Definition	Length (Byte)	Note
0x00	TPID	2	Index ID of function picture

0x02	TPArea	8	Valid area of touch: Upper-left corner coordinates (Xs, Ys), Lower-right corner coordinates (Xe, Ye).
0x0A	TraID	2	Switch picture index ID；0xFFxx: no switching
0x0C	AniID	2	Tap effect picture index ID; 0xFFxx: no effect to be display
0x0E	Code	2	0xFE08/0xFD08
0x10	0xFE	1	0xFE
0x11	TP_OnMode	1	Data return mode for the first tap: 0x00: No State 0x01: Copy data which pointed by VP1S with the length of LEN1 to the variable space pointed by VP1D 0x02: Copy data which pointed by VP1S with the length of LEN1 to UART interface 0x03: Copy data which pointed by VP1S with the length of LEN1 to the register space pointed by VP1D Other: reserved
0x12	VP1S	2	Address of data to be read for the first tap
0x14	VP1D	2	Address of data to be written for the first tap
0x16	NULL	1	0x00
0x17	LEN1	1	Length of returning data in byte. When data returning mode is 0x01, the value must be even.
0x18	0xFE	1	0xFE
0x19	TP_ContinueMode	1	Data returning mode for the long tap: 0x00: No State 0x01: Copy data which pointed by VP1S with the length of LEN2 to the variable space pointed by VP1D 0x02: Copy data which pointed by VP1S with the length of LEN2 to UART interface 0x03: Copy data which pointed by VP1S with the length of LEN2 to the register space pointed by VP1D Other: reserved
0x1A	VP1S	2	Address of data to be read for the long tap
0x1C	VP1D	2	Address of data to be written for the long tap
0x1E	NULL	1	0x00
0x1F	LEN2	1	Length of returning data in byte. When Data return mode is 0x01, the value must be even.
0x20	0xFE	1	0xFE
0x21	TP_OffMode	1	Data return mode for tap off: 0x00: No State 0x01: Copy data which pointed by VP1S with the length of LEN3 to the variable space pointed by VP1D 0x02: Copy data which pointed by VP1S with the length of LEN3 to UART interface 0x03: Copy data which pointed by VP1S with the length of LEN3 to the register space pointed by VP1D Other: reserved
0x22	VP1S	2	Address of data to be read for tap off
0x24	VP1D	2	Address of data to be written for tap off
0x26	NULL	1	0x00
0x27	LEN3	1	Length of returning data in byte. When Data return mode is 0x01, the value must be even.
0x28	0x00	8	0x00

The three states of the Tap-On:

---

6   Variable Display Configuration File

Variable display function configured by file 14*.bin which consists of one or more instructions that describes the function of a variable display.

Each instruction is fixed to occupy 32 bytes.

Each page is fixed with a 2KB or 4KB (0x1000 or 0x0800) variable storage space, and each page can be configured to display 64 or 128 variable functions (64/128 are selected by RC.4 in file CONFIG.TXT).

Variable display configuration file (14*.bin) is up to 2MB, configurable up to 1024 pages (Up to 512 pages in 128 display functions mode).

For variables of the same type, the lower the storage location, the higher the display priority.

A display variable instruction consists of 6 parts, refer to the table below:

No.	Definition	Length (Byte)	Note
1	0x5A	1	Fixed value
2	Type	1	Variable type
3	*SP	2	Destination address pointer for Variable description file which loaded from flash to data space. 0xFFFF: Do not load to data space.
4	Len_Dsc	2	The length of variable description
5	*VP	2	Variable address, 0x0000 ~ 0x6FFF, 0x0000 is used for the variable which need not specifying an address. This instruction will be canceled when the variable address has high bytes of 0xFF.
6	Description	N	Variable description

Variable Display Function List

No.	Code	Function	Note
01	0x00	Variable icon display	Linearly corresponds the range of a data variable to a set of icons. When the variable changes, the icon also automatically switches accordingly, mostly for fine dashboards, progress bars
02	0x01	Animation icon display	A constant data variable corresponds to 3 different icons indicating the status: do not display, display a icon, and display an animation icon. Mostly used for alarm tips.
03	0x02	Progress Bar display	The range of a data variable is represented by the position of an icon (slider). Mostly used for liquid level, dial, schedule indication
04	0x03	Artistic character display	Replace the font library with icons to display variable data
05	0x04	Animation display	Play a set of full-screen pictures at the specified speed; Mostly for the boot interface or screensaver
06	0x05	Icon rotation display	The range of a data variable is linearly corresponding to the angle data, and then an icon is rotated according to the corresponding angle data, which is mostly used for display of pointer meter, etc.
07	0x06	Variable Bit display	The state of each bit of a data variable corresponds to two of the 8 different display scenarios, which are displayed with icons (or icon animations). Mostly used for switch status display, such as fan operation (animation), stop (still icon).
08	0x10	Numeric Variable display	Display a numeric variable in the specified format (integer, decimal, unit) with a number of the specified font and size
09	0x11	Text display	Displays the string in the specified format (determined by the selected font library), displayed in the specified area
10	0x12_00	Text Format RTC display	Display RTC in text with customize format
11	0x12_01	Dial format RTC display	Rotate the icon and display the RTC in the form of a pointer dial
12	0x13	HEX data display	Separate variable data by byte (HEX) and display with specified ASCII characters. Mostly for timing display, such as showing 1234 as 12: 34

13	0x14	Scroll text display	Displays the specified content in the specified area using scrolling
14	0x20	Curve display	Auto Match command 0x84 write curve buffer data to display real-time curves (trend graph which can specify display area, center axis coordinates, display scale (zoom in/out).
15	0x21	Drawing	Draw Basic Graphics, drawing command list as below.
16	0x22	List display	Display the data which defined by the two-dimensional array separately in tables.
17	0x25	2D QR code display	Displays the specified content in the specified location by means of 2D QR code

Drawing command list

Command	Function	Description
0x0001	Set point	Set point (x, y, color)
0x0002	Draw line	Multi-point Connection (color, (x0, y0), … (xn, yn))
0x0003	Draw a rectangular box	Display a rectangular box with customize size, area, location
0x0004	Draw a rectangular area	Populates the specified area with the specified color
0x0005	Draw a circle box	Display a circle box with customize size, area, location
0x0006	Picture clipping display	Cut an area from the specified page to the current page
0x**07	Icon Display	Gets the icon from the specified icon library and displays
0x0008	Monochrome Area Fill	Specify the color fill in the specified area
0x0009	Draw a vertical line	Display vertical lines based on variable data, with customize colors, locations
0x000A	Draw Segments	Customize colors, locations, and display connections based on variable data
0x000B	Draw Arcs	Display arcs with customize radius, color, starting and ending angles
0x000C	Character display	Display characters based on variable data
0x000D	Reverse color display in rectangular area	Reverse color display of the specified area
0x000E	Two-color bitmap display	Display the appropriate color by bit based on variable data
0x000F	Bitmap display	Statement variable data display

6.1 Icon Display

6.1.1 Variable Icon Display (0x00)

Address		Definition	Length (byte)	Note	
0x00		0x5A00	2	Fixed value	
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file	
0x04		0x0008	2	Fixed value	
0x06	0x00	*VP	2	Variable pointer, variable is integer	
0x08	0x01	x, y	4	Variable display position, upper-left coordinates of icons	
0x0C	0x03	Vmin	2	Variable Lower limit, overflow data is not displayed	
0x0E	0x04	Vmax	2	Variable Upper limit, overflow data is not displayed	
0x10	0x05	IconMin	2	Icon ID corresponding to Vmin	
0x12	0x06	IconMax	2	Icon ID corresponding to Vmax	
0x14	0x07: H	IconLib	1	Icon Library location	
0x15	0x07: L	Mode	1	Icon Display mode	0x00: Transparent (no background displayed)
					other: Display icon Background

6.1.2 Animation Icon Display (0x01)

Address		Definition	Length (byte)	Note
0x00		0x5A01	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000A	2	Fixed value
0x06	0x00	*VP	2	Icon variable pointer, variable in double word, low word is reserved, high word is unsigned number (0x0000-0x0FFFF). Animation icon display is controlled by user data.
0x08	0x01	x, y	4	Variable display location, upper-left coordinates of icons
0x0C	0x03	0x0000	2	Fixed value
0x0E	0x04	VStop	2	Stop animation display when the variable equals this value
0x10	0x05	VStart	2	Automatically display the animation icon when the variable equals this value
0x12	0x06	IconStop	2	The icon corresponding to VStop
0x14	0x07	IconStart	2	When the variable equals VStart value, the icon is automatically
0x16	0x08	IconEnd	2	displayed from IconStart to IconEnd to form an animation
0x18	0x09: H	IconLib	1	Icon Library location
0x19	0x09: L	Mode	1	Icon Display Mode: 0x00: Transparent (no background displayed) other: Display icon background

Note: (VP +1) is a reserved bit that cannot be used. Icons or animations will not be displayed when variables are not equal to Vstop or Vstart.

6.1.3 Progress Bar Display(0x02)

Address		Definition	Length (byte)	Note	
0x00		0x5A02	2	Fixed value	
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file	
0x04		0x000A	2	Fixed value	
0x06	0x00	*VP	2	Icon variable pointer, variable in double word, low word is reserved, high word is unsigned number (0x0000-0x0FFFF). Animation icon display is controlled by user data.	
0x08	0x01	VBegain	2	Variable values corresponding to the begin scale	
0x0A	0x02	VEnd	2	Variable values corresponding to the end scale	
0x0C	0x03	XBegain	2	Start scale X coordinates (Y coordinates in vertical)	
0x0E	0x04	XEnd	2	End scale X coordinates (Y coordinates in vertical)	
0x10	0x05	IconID	2	Icon ID of the progress bar	
0x12	0x06	Y	2	The Y coordinates of the icon (X coordinates in vertical)	
0x14	0x07: H	XAdj	1	The X coordinates offset of the icon (Y coordinates in vertical)	
0x15	0x07: L	Mode	1	Scale mode	0x00: Horizontal Bar
					0x01: Vertical Bar
0x16	0x08: H	IconLib	1	Icon Library location	
0x17	0x08: L	DisplayMode	1	Icon Display mode	0x00: Transparent (no background displayed)
					other: Display icon background
0x18	0x09: H	DataMode	1	0x00: *VP points to an integer variable	
				0x01: *VP points to high-byte data of an integer variable	
				0x02: *VP points to low-byte data of an integer variable	

6.1.4 Artistic Character Display (0x03)

Address		Definition	Length (byte)	Note
0x00		0x5A03	2	Fixed value

0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file	
0x04		0x0007	2	Fixed value	
0x06	0x00	*VP	2	Variable pointer	
0x08	0x01	X, Y	4	Start position for display	Right alignment mode, upper-right coordinate of the string.
					Left alignment mode, upper-left coordinate of the string.
0x0C	0x03	Icon0	2	The IconID corresponds to 0, the order is 0123456789 … …	
0x0E	0x04: H	IconID	1	Icon library index ID	
0x0F	0x04: L	IconMode	1	Icon Display mode	0x00: Transparent (no background displayed)
					other: Display icon background
0x10	0x05: H	Nint	1	Number of integer bits of data to be displayed.	
0x11	0x05: L	NDot	1	Number of decimal bits of data to be displayed.	
0x12	0x06: H	VType	1	0x00: integer (2 bytes): -32768 ~ +32767	
				0x01: long integer (4 bytes): -2147483647 ~ +2147483647	
				0x02: *VP high byte, unsigned integer, 0 ~ 255	
				0x03: *VP low byte, unsigned integer, 0 ~ 255	
				0x04: super long integer (8 byte): -9223372036854775808 ~ +9223372036854775807	
				0x05: unsigned integer (2 bytes), 0 ~ 65535	
				0x06: unsigned long integer (4 byte), 0 ~ 4294967295	
0x13	0x06: L	Mode	1	0x00: Left alignment	
				0x01: Right alignment	
				0x02: Center alignment	

6.1.5 Picture Animation Display (0x04)

Address		Definition	Length (byte)	Note
0x00		0x5A04	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x0004	2	Fixed value
0x06	0x00	0x0000	2	Fixed value
0x08	0x01	PicBegin	2	Index ID of the first picture
0x0A	0x02	PicEnd	2	Index ID of the last picture
0x0C	0x03: H	FrameTime	1	Display time of one frame, unit: 8ms

Note: The Begin picture index ID must be less than the End picture index ID. Picture loop display can be achieved by setting the picture animation variable of the end page. During the image animation process, the picture animation can be ended by switching the page through the 0x80 command or touch instruction.

6.1.6 Icon Rotation Display (0x05)

Address		Definition	Length (byte)	Note
0x00		0x5A05	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000C	2	Fixed value
0x06	0x00	*VP	2	Variable pointer, variable format determined by VPMode
0x08	0x01	IconID	2	The specified icon index ID
0x0A	0x02	IconXc	2	Rotation center position of the icon: X coordinates
0x0C	0x03	IconYc	2	Rotation center position of the icon: Y coordinates
0x0E	0x04	Xc	2	Rotation center position of the page: X coordinates
0x10	0x05	Yc	2	Rotation center position of the page: Y coordinates
0x12	0x06	VBegain	2	Variable values corresponding to the begin rotation angle, in integer, overflow data is not displayed.

0x14	0x07	VEnd	2	Variable values corresponding to the end rotation angle, in integer, overflow data is not displayed.
0x16	0x08	ALBegain	2	Begin rotation angle, 0 ~ 720(0x000-0x2D0), Unit: 0.5°
0x18	0x09	ALEnd	2	End rotation angle, 0 ~ 720(0x000-0x2D0), Unit: 0.5°
0x1A	0x0A: H	VPMode	1	0x00: *VP point to an integer variable
				0x01: *VP point to high-byte data of an integer variable
				0x02: *VP point to low-byte data of an integer variable
0x1B	0x0A: L	LibID	1	Icon library index ID
0x1C	0x0B: H	Mode	1	Icon Display mode	0x00: Transparent (no background displayed)
					other: Display icon background

Note: Rotation is always assumed to be "clockwise" rotation, that is, ALEnd must be greater than ALBegain.
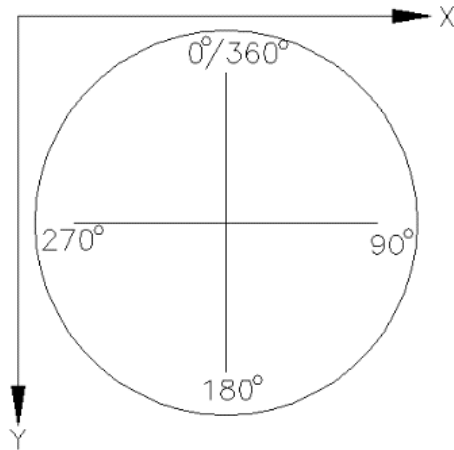
6.1.7 Variable Bits Display (0x06)

Address		Definition	Length (byte)	Note
0x00		0x5A06	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000C	2	Fixed value
0x06	0x00	*VP	2	Variable pointer, variable format determined by VPMode
0x08	0x01	*VpAux	2	Auxiliary variable pointer, double byte, cannot be accessed by user software
0x0A	0x02	ActBitSet	2	Bits that equals 1 indicate the corresponding bits of the *VP needs to be displayed.
0x0C	0x03: H	DisplayMode	1	When DisplayMode is 0x02, the Icon0S icon is displayed when the corresponding variable (*VP) has a certain bit of "0"

When DisplayMode is 0x02, the Icon0S icon is displayed when the corresponding variable (*VP) has a certain bit of "0"

DisplayMode	Variable Bit Value	
	0	1
0x00	Icon0S	Icon1S
0x01	Icon0S	not display
0x02	Icon0S	Icon0S ~ Icon0E Animation
0x03	not display	Icon1S
0x04	not display	Icon0S ~ Icon0E Animation
0x05	Icon0S~ Icon0E Animation	not display
0x06	Icon0S~ Icon0E Animation	not display
0x07	Icon0S~ Icon0E Animation	Icon0S ~ Icon0E Animation

				Bitmap Icon Arrangement
0x0D	0x03: L	MoveMode	1	0x00: X++, do not keep DisMov position for the bit which specified non-displayed by ActBitSet.
				0x01: Y++, do not keep DisMov position for the bit which specified non-displayed by ActBitSet.
				0x02: X++, keep DisMov position for the bit which specified non-displayed by ActBitSet.
				0x03: Y++, keep DisMov position for the bit which specified non-displayed by ActBitSet.
0x0E	0x04: H	IconMode	1	Icon Display mode — 0x00: Transparent (no background displayed) / other: Display icon background
0x0F	0x04: L	IconLib	1	Icon library index ID
0x10	0x05	Icon0S	2	Not display animation mode, bit 0, Icon index ID. Display animation mode, bit 0, start Icon index ID of icon animation.
0x12	0x06	Icon0E	2	Display animation mode, bit 0, end Icon index ID of icon animation.
0x14	0x07	Icon1S	2	Not display animation mode, bit 1, Icon index ID. Display animation mode, bit 1, begin start Icon index ID of icon animation.
0x16	0x08	Icon1E	2	Display animation mode, bit 1, end Icon index ID of icon animation.
0x18	0x09	x, y	4	Starting bit variable display position, upper-left coordinate of icon
0x1C	0x0B: H	DisMov	2	Coordinate interval to the next icon
0x1E		Reserved	2	Recommend to set "0"

6.2 Text Display

6.2.1 Numeric Variable Display (0x10)

Address		Definition	Length (byte)	Note	
0x00		0x5A10	2	Fixed value	
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file	
0x04		0x000D	2	Fixed value	
0x06	0x00	*VP	2	Variable pointer	
0x08	0x01	x, y	4	Display position, upper-left coordinates of strings	
0x0C	0x03	Color	2	Display color	
0x0E	0x04: H	LibID	1	ASCII Font location	
0x0F	0x04: L	FontSize	1	Number of matrix dot in X-direction of Character	
0x10	0x05: H	Mode	1	0x00: Left alignment 0x01: Right alignment 0x02: Center	
0x11	0x05: L	NInt	1	Number of integer bits displayed	The sum of NInt and NDot cannot exceed 20
0x12	0x06: H	NDot	1	Number of decimal bits displayed	
0x13	0x06: L	VType	1	0x00: integer (2 bytes), -32768 ~ +32767 0x01: long integer (4 bytes), -2147483647 ~ +2147483647 0x02: *VP high byte, unsigned integer, 0 ~ 255 0x03: *VP low byte, unsigned integer, 0 ~ 255 0x04: super long integer (8 bytes), -9223372036854775808 ~ +9223372036854775807 0x05: unsigned integer (2 bytes), 0 ~ 65535 0x06: unsigned long integer (4 bytes), 0 ~ 4294967295	

| 0x14 | 0x07: H | Len | 1 | The length of the unit (for fixed string), 0x00: no unit to be display |
| 0x15 | 0x07: L | StringUnit | Max11 | Unit string (ASCII) |

6.2.2 Text Display(0x11)

Address		Definition	Length (byte)	Note
0x00		0x5A11	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000D	2	Fixed value
0x06	0x00	*VP	2	Variable pointer
0x08	0x01	X, Y	4	Display position, upper-left coordinates of string
0x0C	0x03	Color	2	Display color
0x0E	0x04	Area	8	Text box (Xs, Ys) (Xe, Ye)
0x16	0x08	Textlength	2	Text length in byte. Text display ends at data 0xFFFF/0x0000 or Textbox tail
0x18	0x09: H	Font0ID	1	The index ID of ASCII font library when encoding mode is 0x01~0x04
0x19	0x09: L	Font1ID	1	The index ID of non-ASCII font library when encoding mode is 0x01~0x04 and the index ID of font library when encoding mode is 0x00/0x05.
0x1A	0x0A: H	FontXDots	1	Number of matrix dot in X-direction of font (in mode 0x01 ~ 0x04, the X value of the ASCII character is calculated according to X/2)
0x1B	0x0A: L	FontYDots	1	Number of matrix dot in Y-direction of the font
0x1C	0x0B: H	Encode	1	[6: 0]: encoding mode <table><tr><td>Value</td><td>Mode</td></tr><tr><td>0</td><td>8bit encoding</td></tr><tr><td>1</td><td>GB2312</td></tr><tr><td>2</td><td>GBK</td></tr><tr><td>3</td><td>BIG5</td></tr><tr><td>4</td><td>SJIS</td></tr><tr><td>5</td><td>UNICODE</td></tr></table>[7]: character spacing adjustment <table><tr><td>Value</td><td>Note</td></tr><tr><td>0</td><td>Auto adjust</td></tr><tr><td>1</td><td>not auto adjust</td></tr></table>
0x1D	0x0B: L	HorDis	1	Character Horizontal interval
0x1E	0x0C: H	VerDis	1	Character Vertical interval
0x1F	0x0C: L	Reserved	1	Recommend to set "0"

Note: The Y-direction matrix dot of the character must be an even number. The pre-installed font library (ID=0) includes 4*8 ~ 64*128 dot matrix ASCII characters.

6.2.3 Real Time Clock (RTC) Display(0x12)

6.2.3.1 Text mode

Address		Definition	Length (byte)	Note
0x00		0x5A12	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000D	2	Fixed value
0x06	0x00	0x0000	2	Fixed value
0x08	0x01	X, Y	4	Display position, upper-left coordinates of string

0x0C	0x03	Color	2	Display color of font
0x0E	0x04: H	LibID	1	Font library index ID
0x0F	0x04: L	FontSize	1	Number of matrix dot in X-direction of the font
0x10	0x05	StringCode	MAX16	String that displayed is defined by RTC encoding rules. For example: current time is 2014-03-22 16: 18: 50 Saturday, then: Mode1: Y-M-D H: M: S 0x00 display as: 2014-03-22 16: 18: 50 Mode2: M-D W H: M 0x00 display as: 03-22 SAT 16: 18

RTC encoding rules

Information	Code	Display Format
Year	Y	2000 ~ 2099
Month	M	01 ~ 12
Day	D	01 ~ 31
Hour	H	00 ~ 23
Minute	Q	00 ~ 59
Second	S	00 ~ 59
Week	W	SUN MON TUE WED THU FRI SAT
End sign	0x00	

6.2.3.2 Dial mode

Address		Definition	Length (byte)	Note
0x00		0x5A13	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000D	2	Fixed value
0x06	0x00	0x0001	2	Fixed value
0x08	0x01	x, y	4	The center of the dial
0x0C	0x03	IconHour	2	The index ID of the hour hand Icon, 0xFFFF: do not display
0x0E	0x04	IconHourCentral	4	Rotation center position of the hour hand Icon
0x12	0x06	IconMinute	2	The index ID of the minute hand Icon, 0xFFFF: do not display
0x14	0x07	IconMinuteCentral	4	Rotation center position of the minute hand Icon
0x18	0x09	IconSecond	2	The index ID of the second hand Icon, 0xFFFF: do not display
0x1A	0x0A	IconSecondCentral	4	Rotation center position of the second hand Icon
0x1E	0x0C: H	IconLib	1	The index ID of Icon library
0x1F		Reserved	1	Recommend to set "0"

6.2.4 Time Variable Display (0x13)

Address		Definition	Length (byte)	Note
0x00		0x5A13	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000D	2	Fixed value
0x06	0x00	*VP	2	Variable pointer of start Address of data string. Variable is saved as BCD(HEX) code. For example: Data 0x32 displays as 32, Data 0xBF displays BF
0x08	0x01	X, Y	4	Display position, upper-left coordinates of strings
0x0C	0x03	Color	2	Font color

0x0E	0x04: H	ByteNum	1	Byte number showed in high byte of *VP pointer, 0x01 ~ 0x0F
0x0F	0x04: L	LibID	1	Font library index ID, halfwidth mode for Chinese character. If LibID is not 0, the font library should be 8-bit coding
0x10	0x05: H	Fontx	1	Number of matrix dot in X-direction of the font
0x11	0x05: L	StringCode	MAX15	Encode strings, combined with time variables, to get the display format required. Special characters are defined as follows: 0x00: Invalid, not display 0x0D: Line breaks, X=Xs, Y=Y+FontX*2

6.2.5 Text Scrolling Display (0x14)

Address		Definition	Length (byte)	Note
0x00		0x5A13	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000B	2	Fixed value
0x06	0x00	*VP	2	Text pointer, the first three bytes are reserved and the text start at (VP+3), stop at 0xFF/0x00
0x08	0x01: H	RollingMode	1	Rolling mode, from right to left
0x09	0x01: L	RollingDis	1	Scroll interval, number of pixels for each text scrolling cycle
0x0A	0x02: H	AdjustMode	1	Display Adjustment mode: 0x00: Left alignment 0x01: Center 0x02: Right alignment
0x0B	0x02: L	RunControl	1	Scrolling control: 0x00: Run 0x01: Pause 0x02: Stop 0x03: Initialization (static display)
0x0C	0x03	Color	2	Text color
0x0E	0x04	(Xs, Ys) ~ (Xe, Ye)	8	Text box
0x16	0x08: H	Font0ID	1	Font library of ASCII character. The default font library index ID is 0 when coding mode is 0x00/0x05.
0x17	0x08: L	Font1ID	1	Font library index ID for Non-ASCII characters
0x18	0x09: H	FontXDots	1	Number of matrix dot in X-direction of Character, automatically calculated as X/2 when it is ASCII characters
0x19	0x09: L	FontYDots	1	Number of matrix dot in Y-direction of Character,
0x1A	0x0A: H	EncodeMode	1	[6: 0]: encode mode 0x00: 8Bits 0x01: GB2312 0x02: GBK 0x03: BIG5 0x04: SJUS 0x05: UNICODE [7]: character spacing adjustment 0: auto adjust 1: not auto adjust
0x1B	0x0A: L	TextDis	1	Character interval
0x1C	0x0B	Reserved	4	Recommend to set "0"

Note: The number of matrix dot in Y-direction of the character must be even number. The pre-installed font library (ID=0) includes 4*8 ~ 64*128 dot matrix ASCII characters.

6.3 Graph Display

6.3.1 Curve Display(0x20)

Address		Definition	Length (byte)	Note
0x00		0x5A20	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000A	2	Fixed value
0x06	0x00	0x0000	2	undefined
0x08	0x01	Area	8	Curve window: Upper-left corner coordinates (Xs, Ys) Lower-right corner coordinates (Xe, Ye) The overflow data will not be displayed
0x10	0x05	YCentral	2	Curve Center Axis location
0x12	0x06	VDCentral	2	The value of curve center axis: generally, it is average of maximum and minimum data
0x14	0x07	Color	2	Curve color
0x16	0x08	MulY	2	Amplification Multiples of Y axis, Unit: 1/256, 0x0000 ~ 0x7FFF
0x18	0x09: H	Channel	1	Data source channel: 0x00 ~ 0x07
0x19	0x09: L	DisHor	1	Horizontal axis Interval: 0x01 ~ 0xFF

6.3.2 Basic Graphical Display (0x21)

Address		Definition	Length (byte)	Note
0x00		0x5A21	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x0008	2	Fixed value
0x06	0x00	*VP	2	Variable data pointer
0x08	0x01	Area	8	Display area: from upper-left coordinates to lower-right corner coordinates and the overflow data will not be displayed. Available for command 0x0001 ~ 0x0005, 0x0009 ~ 0x000B.
0x10	0x05	Reserved	18	Recommend to set "0"

Note: A "drawing board" is defined in file 14*.bin firstly, the specific drawing operation is determined by the content in the variable space pointed to by *VP. Users implement different drawing functions by sending different data frames.

Variable data frame structure in variable space

Address	Definition	Note
VP	CMD	Drawing command
VP+1	Data_Pack_Num_Max	Maximum number of data packages, connection command (0x0002), number of connection lines is defined as (number of vertex -1)
VP+2	Data_Pack	Data package

Drawing command format

Command (CMD)	Function	Command Frame structure			
		Address (Relative)	Length (Byte)	Definition	Note

0x0001	Draw dot	0x00	2	(x, y)	Coordinate of Dot, high byte of x parameter is condition.
		0x02	1	color	Dot color
0x0002	Draw line	0x00	1	color	Line color
		0x01	2	(x, y)0	Coordinate of endpoint0, high byte of x parameter is condition*.
		0x03	2	(x, y)1	Coordinate of endpoint1, high byte of x parameter is condition*.
		...	...	...	...
		0x01+2*n	2	(x, y) n	Coordinate of endpointn, high byte of x parameter is condition*.
0x0003	Draw Rectangular box	0x00	2	(x, y) s	Upper-left corner coordinates of the box, high byte of x parameter is condition*.
		0x02	2	(x, y) e	Lower-right corner coordinates
		0x04	1	color	Rectangular box color
0x0004	Draw Rectangular Area	0x00	2	(x, y) s	Upper-left corner coordinates, high byte of x parameter is condition*.
		0x02	2	(x, y) e	Lower-right corner coordinates
		0x04	1	color	Rectangular area color
0x0005	Draw circle	0x00	2	(x, y)	Center point coordinates, high byte of x parameter is condition*.
		0x02	1	rad	Radius
		0x03	1	color	Circle color
0x0006	Cut picture area	0x00	1	Pic_Index	Picture index ID, high byte of x parameter is condition*.
		0x01	2	(x, y) s	Upper-left corner coordinates of cut area
		0x03	2	(x, y) e	Lower-right corner coordinates of cut area
		0x05	2	(x, y)	Upper-left corner coordinates of the target area to be pasted in current page.
0x**07	Icon display	0x00	2	(x, y)	Location of Icon, high byte of x parameter is condition*.
		0x02	1	Icon_Index	Icon index ID. The icon library position is indicated by the high byte of the command. Icon is displayed without background color.
0x0008	Area fill	0x00	2	(x, y)	Seed point coordinates, high byte of x parameter is condition*.
		0x02	1	color	Filled color
0x0009	Spectrum display	0x00	1	color	Connect (x0, y0s) (x0, y0e) with color lines, high byte of x0 is condition*.
		0x01	3	x0, y0s, y0e	
0x000A	Line segment display	0x00	1	color	Connect (x, y) s and (x, y) e with color lines, high byte of xs is condition*
		0x01	2	(x, y) s	
		0x03	2	(x, y) e	
0x000B	Arc display	0x00	1	color0	Arc color
		0x01	2	(x, y)0	Center point coordinates, high byte of x parameter is condition*.
		0x03	1	rad0	Radius
		0x04	1	Deg_s0	Starting angle, unit 0.5°, 0 ~ 720
		0x05	1	Deg_e0	Termination angle, unit 0.5°, 0 ~ 720
0x000C		0x00	1	color0	Character color

		0x01	2	(x, y)0	Starting position, high byte of x parameter is condition*.
		0x03H	0.5	Lib_Index	Font library Index ID
	Character display	0x03L	0.5	mode	Encoding mode: 0: 8bit 1: GB2312 2: GBK 3: BIG5 4: SJIS 5: UNICODE
		0x04H	0.5	x_dots	Number of matrix dot in X-direction of Character
		0x04L	0.5	y_dots	Number of matrix dot in Y-direction of Character
		0x05	1	text0	Character data
0x000D	Reverse color display in rectangular area	0x00	2	(x, y) s	Upper-left corner coordinates, high byte of x parameter is condition*.
		0x02	2	(x, y) e	Lower-right corner coordinates
		0x04	1	color	Color to be displayed, 0xFFFF: reverse current color.
0x000E	Two-color bitmap display	0x00	2	(x, y) s	Upper-left corner coordinates of the rectangular area, high byte of x parameter is condition*.
		0x02	1	x_dots	Number of matrix dot in X-direction
		0x03	1	y_dots	Number of matrix dot in Y-direction
		0x04	1	color0	Color corresponding to bit "0"
		0x05	1	color1	Color corresponding to bit "1"
		0x06	N	DataPack	Data to be displayed, MSB mode, word alignment: each line should start at a new word.
0x000F	Bitmap display	0x00	2	(x, y) s	Upper-left corner coordinates of the rectangular area, high byte of x parameter is condition*.	
		0x02	1	x_dots	Number of matrix dot in X-direction	Limited by variable space, the maximum display bitmap is 196x146(4:3)/226x126(16:9)
		0x03	1	y_dots	Number of matrix dot in Y-direction	
		0x04	N	DataPack	Data to be displayed. Data format: MSB, 5R6G5B	

*Condition:
0xFF: the end of drawing operation
0xFE: ignore this operation

6.3.3 Table Display(0x22)

Address		Definition	Length (byte)	Note
0x00		0x5A22	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x000C	2	Fixed value

0x06	0x00	*VP	2	Variable data pointer
0x08	0x01: H	Tab_X_Num	1	Number of columns, 0x01 ~ 0xFF
0x09	0x01: L	Tab_Y_Num	1	Number of rows, 0x01 ~ 0xFF
0x0A	0x02: H	Tab_X_Start	1	Position of the first column, 0x00 ~ 0xFF
0x0B	0x02: L	Tab_Y_Start	1	Position of the first row, 0x00 ~ 0xFF
0x0C	0x03: H	Unit_Data_Num	1	0x01 ~ 0x7F: Data length for cell (in word), all cells have the same data length.
				0x00: The variable space pointed by the *VP pointer defines the data length (in word) of different column cells: table data content storage location corresponding deferred (UNIT_DATA_NUM/2) up to the entire word address. For example: When *VP0=x1000, and Tab_X_Num = 7, 0x1000 ~ 0x1003 stores the table data length of column 0 to 6 in turn, Where the 0x1003 low bytes are not used; The 0x1004 address begins to store the table contents.
0x0D	0x03: L	Encode_Mode	1	[7]: Character interval adjustment 0: auto adjust 1: not auto adjust
				[6]: Cell format 0: text format 1: defined by the first two words of the data
				[5]: Border display setting 0: display border 1: not display border
				[4]: undefined, recommend to set "0"
				[3: 0]: text encoding mode 0: 8bit 1: GB2312 2: GBK 3: BIG5 4: SJIS 5: UNICODE
0x0E	0x04	Area	8	Table area definition: Upper-left corner coordinates and Lower-right corner coordinates The table is always displayed from upper-left corner and ended when overflow.
0x16	0x08	color_line	2	Border line color
0x18	0x09	color_text	2	Text color
0x1A	0x0A: H	fontID0	1	ASCII font library index ID when encoding mode is 0x01 ~ 0x04
0x1B	0x0A: L	fontID1	1	Non-ASCII font library index ID when encoding mode is 0x00/0x05 or 0x01 ~ 0x04
0x1C	0x0B: H	Font_X_dots	1	Number of matrix dot in X-direction of Character, automatically calculated as X/2 when it is ASCII characters
0x1D	0x0B: L	Font_Y_dots	1	Number of matrix dot in Y-direction of Character,
0x1E	0x0C: H	Tab_X_Adj_Mod	1	When Tab_X_Start not equal 0x00, 0x00: not display first column 0x01: display first column

0x1F	0x0C: L	Tab_Y_Adj_Mod	1	When Tab_Y_Start not equal 0x00, 0x00: not display first row 0x01: display first row

When Encode_Mode.6 is set to "1", the cell format is defined by the first two words of the cell data as below

Function	Location	value	Definition	Note
Data type	High bytes of the first word	0x00	Integer, 2 bytes	-32768 ~ 32767
		0x01	Long integer, 4 bytes	-2147483648 ~ 2147483647
		0x02	High byte of *VP, unsigned integer	0 ~ 255
		0x03	Low byte of *VP, unsigned integer	0 ~ 255
		0x04	Super long integer, 8 bytes	-9223372036854775808 ~ 9223372036854775807
		0x05	Unsigned integer, 2 bytes	0 ~ 65536
		0x06	Unsigned long integer, 4 bytes	0 ~ 4294967295
		0x10	Time format 1	BCD string as 12: 34: 56
		0x11	时间格式 2	BCD string as 12-34-56
		0x12	时间格式 3	BCD string as YYYER-MM-DD HH: MM: SS
		0xFF	Text	Text format
Data format	Low bytes of the first word	Data format of the decimal (data type is 0x00 ~ 0x06)		The high 4 bits is the bit number of integer and the low 4 bits is the bit number of the decimal.
		Time BCD string, (data type is 0x10 ~ 0x11)		Length of the BCD string
		others	undefined	
Text color	The second word	The color of the text displayed		

Note: If the actual length of the table is shorter than the length specified by unit_data_num, use 0xFFFF as the cell text terminator. For particularly large tables, through the touch screen operation to modify the Tab_x_start, Tab_y_start value can be very convenient to achieve the table positioning or dragging.

6.3.4 2D QR Code Display(0x25)

Address		Definition	Length (byte)	Note
0x00		0x5A12	2	Fixed value
0x02		*SP	2	Variable description pointer 0xFFFF: loaded by configuration file
0x04		0x0004	2	Fixed value
0x06	0x00	*VP	2	2D code data pointer, data length up to 458 bytes, with 0x0000/0xFFFF as the end flag

|				Greater than or equal to 1 byte and less than 155 bytes	45x45 unit pixels
				Greater than or equal to 155 bytes and less than or equal to 458 bytes	73x73 unit pixels
0x08	0x01	X, Y	4	The upper-left position coordinates of the area	
0x0C	0x03	Size	2	Physical pixel dot size for each QR code unit pixel: 0x01 ~ 0x07。 For example: when Size=2, each unit pixel will be displayed as a 2x2 dot matrix	
0x0D		Reserved	18	Recommend to set "0"	

7   Appendix I Introduction to Font Library

7.1 Pattern Extract

In a font library, the pattern of each character is stored by row.
  ➢ Horizontal (X-axis) direction: encoding from left to right, one word per row.
  ➢ Vertical (Y-axis) direction: encoded from upper down

A sample (16*16 dot matrix font) as below:
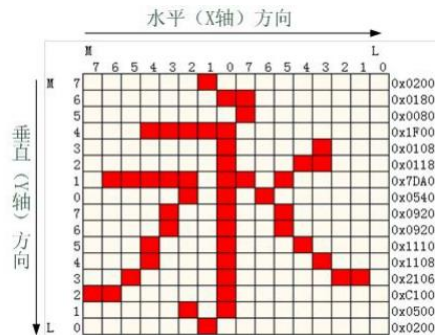
7.2 AGN Display default font library

7.2.1   ASCII Character Set

ASCII is the acronym for the **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange. It is a code for representing 128 English characters as numbers, with each letter assigned a number from 0 to 127. For example, the ASCII code for uppercase M is 77. Most computers use ASCII codes to represent text, which makes it possible to transfer data from one computer to another.

7.2.2   GB2312 Character Set

GB2312 is the registered internet name for a key official character set of the People's Republic of China, used for simplified Chinese characters. GB abbreviates Guojia Biaozhun (国家标准), which means national standard in Chinese. GB2312 (1980) has been superseded by GBK and GB18030, which include additional characters, but GB2312 is nonetheless still in widespread use.

7.2.3   GBK Character Set

GBK is an extension of the GB2312 character set for simplified Chinese characters, used in the People's Republic of China. It includes all unified CJK characters found in GB13000.1-93, i.e. ISO/IEC 10646: 1993, or Unicode 1.1. Since its initial release in 1993, GBK has been extended by Microsoft in Code page 936/1386, which was then extended into GBK 1.0. GBK is also the IANA-registered internet name for the Microsoft mapping, [1] which differs from other implementations primarily by the single-byte euro sign at 0x80.

7.2.4 Unicode Universal Character Set

Unicode is a computing industry standard for the consistent encoding, representation, and handling of text expressed in most of the world's writing systems. The standard is maintained by the Unicode Consortium, and as of June 2018 the most recent version, Unicode 11.0, contains a repertoire of 137, 439 characters covering 146 modern and historic scripts, as well as multiple symbol sets and emoji. The character repertoire of the Unicode Standard is synchronized with ISO/IEC 10646, and both are code-for-code identical.

7.2.5 BIG-5 Traditional Character Set

Big-5 or Big5 is a Chinese character encoding method used in Taiwan, Hong Kong, and Macau for traditional Chinese characters. The People's Republic of China (PRC), which uses simplified Chinese characters, uses the GB character set instead. Big5 gets its name from the consortium of five companies in Taiwan that developed it.

8 Appendix II UART Data Rate Index Table

Baud ID	Rate(bps)	Baud ID	Rate(bps)	Baud ID	Rate(bps)
0x00	1200	0x08	28800	0x10	625000
0x01	2400	0x09	76800	0x11	691200
0x02	4800	0x0A	62500	0x12	921600
0x03	9600	0x0B	125000	0x12-0xFD	Reserved
0x04	19200	0x0C	250000	0xFE	custom
0x05	38400	0x0D	230400	0xFF	Reserved
0x06	57600	0x0E	345600		
0x07	115200	0x0F	460800		

Note: The data rate can be configured by MicroSD/TF card, for example:

1   Create/open file CONFIG.TXT in folder OD_GUI, add a new line of code (or edit existing code) as R1=07 (it will set the rate to 115200bps) and save the file。

2   In the shutdown state, insert the card into the MICROSD/TF card slot and power it on. Then UART data rate of AGN Display is set to 115200bps.

9 Appendix III SysConfig Register

BIT	Weight	Definition	Note
.7	0x80	VDS	"00": Normal Display (0 degrees orientation)
.6	0x40	Reserved	"10": Rotate Display by 90 degrees "01": Rotate Display by 180 degrees "11": Rotate Display by 270 degrees These two bits are likely used as configuration settings to adjust the orientation of the display or screen to one of these four angles, depending on the desired viewing orientation.
.5	0x20	TPLed	0: disable backlight energy saving mode 1: enable backlight energy saving mode, parameters are set in R6/R7/R8 in file config.txt

.4	0x10	FCRC	0: disable CRC-16 verification in UART communication 1: enable CRC-16 verification in UART communication				
.3	0x08	TPSAUTO	0: Disable automatic upload touch/keying input parameters, requiring user access 1: Enable automatic upload touch/keying input parameters				
.2	0x04	L22_InitEn	0: Set all variable storage space to be "0" in initialization 1: Set variable storage space according to font library file 22 in initialization.				
.1	0x02	FRS1	Scanning period, the smaller the scanning period, the more device sensitive, but the lower the processing capacity. The scan period affects the animation speed displayed by the animation icon.				
.0	0x01	FRS0	Scanning period	80ms	120ms	160ms	200ms
			FRS1	1	1	0	0
			FRS0	1	0	1	0

10   Appendix IV AuxConfig Register

BIT	Weight	Definition	Note
.7	0x80	Reserved	Recommend to set "0"
.6	0x40	RunOSEnable	0: Disable running OS 1: Enable running OS
.5	0x20	Reserved	Reserved
.4	0x10	Page128Enable	0: 64 variables displayed per page 1: 128 variables displayed per page
.3	0x08	CRCAckEnable	0: Disable automatic acknowledge of CRC results 1: Enable automatic acknowledge of CRC results
.2	0x04	Reserved	Recommend to set "0"
.1	0x02	Reserved	Recommend to set "0"
.0	0x01	Reserved	Recommend to set "0"