



# 4.3inch e-Paper User Manual

## Contents

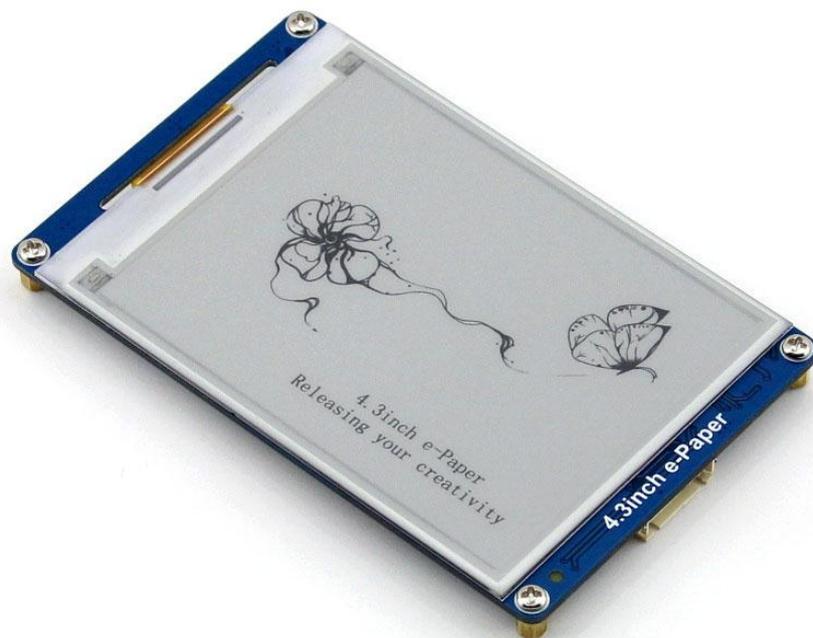
4.3inch e-Paper User Manual.....	1
Overview.....	2
1. Important notices.....	3
2. How to use.....	4
2.1. How to generate bitmap images.....	4
2.2. How to work with PC.....	4
2.3. Arduino.....	11
3. Serial communication protocol.....	13
3.1. Basic rules.....	13
3.1.1. Serial port.....	13
3.1.2. Command frame format.....	13
3.1.3. Transmission sequence.....	13
3.1.4. Command parameters.....	13
3.1.5. Color definition.....	13
3.2. Command set.....	14
3.2.1. Command look-up table.....	14
3.2.2. Command explanations.....	16

## Overview

4.3 inch e-Paper is an e-book display module with serial interface developed by Waveshare. In an easy-to-use design, it enable you to apply different display functions via serial communication, including shapes drawing, text and image displaying, but do not need to know any complex underlying details, making it possible to unleash your creativity.

The 4.3 inch e-Paper has the following features:

- Built-in font libraries, supporting 32, 48 and 64 dots GBK Chinese font and 32, 48 and 64 dots English font;
- Built-in 128MB NandFlash, allowing the font and the image data to be stored to either an external TF card or the internal NandFlash by a serial command;
- 4-level gray display with 800x600 display resolution;
- Adjustable Baud rate, default: 15200;
- Power supply voltage range from 3.3V to 5.5V, logic level compatible;
- Ultra low power consumption, system sleeping current lower than 5mA on 5V power supply;
- Provide with PC software to operate the display, easy to use with PC.



## 1. Important notices

- **System power up**

After powered up, the screen is refreshed to white. At this moment, if no external TF card is in used, the state indicator will flicker 3 times and then keep on. When there is an external TF card in used, the state indicator will light up and keep on without flickering.

- **Storage**

After powered up, the system reads the font and the image data from the internal NandFlash by default.

When using an external TF card, you should format the TF card into a FAT32 system, and place the font libraries provided by Waveshare and the images you want to display to the external TF card before using it.

When using the internal NandFlash, you should import the font libraries and image files you want to display to the internal NandFlash in advance. For more detailed information, please refer to the Section 2.2.

- **State indicator**

The state indicator is on when the system is working properly, and off when the system enters sleep mode. A rising edge on the WAKE\_UP pin from the external MCU can wake up the system. And the state indicator will light up and remain on after resuming from sleep mode.

- **Font libraries**

The system built-in 32, 48 and 64 dots English font is always available without using the TF card or the NandFlash; however, for the 32, 48 and 64 dots Chinese font, you should store the relative library file to the TF card or the NandFlash before using it.

- **Image display**

The system supports 1-bit and 2-bit bitmap image display. For other image formats, you should convert these images into the specified format with the tool on the Demos so as to display them.

The name of the image should be in uppercase English character(s) with the length less than 10 characters, in which the symbol “.” is included.

For example:

PIC7.BMP and PIC789.BMP are correct image names;

PIC7890.BMP is a wrong image name.

## 2. How to use

### 2.1. How to generate bitmap images

1. Make sure the resolution of the image is less than 800x600 pixels. Start the tool **mspaint.exe** on Windows to open the image to be converted;
2. Select the option **24-bit bitmap** in the **Save as Type** list, to save the image as a bmp format file;
3. Start the software tool provided by Waveshare: **uC-GUI-BitmapConvert.exe**;
4. Click File -> Open, and select the bitmap image you want to convert;
5. Click Image -> Convert Into -> Gray4 (2 BPP);
6. Click File -> Save As, and select the option **Windows Bitmap file(\*.bmp)** in the **Save as Type** list, and then enter a correct file name and save the image. Please take a note to the format of the file name.

### 2.2. How to work with PC

We have designed the software for E-Paper to work with PC. With this software, users can easily operate different basic displays on the E-Paper via a PC. In additional, a USB-to-serial module should be applied to build up the communication between the e-Paper and the PC. In here, we will take the serial module CP2102 USB UART Board(mini) as an example to illustrate the application. For more detailed information about this serial module, please refer to Appendix.

- **Hardware connection**

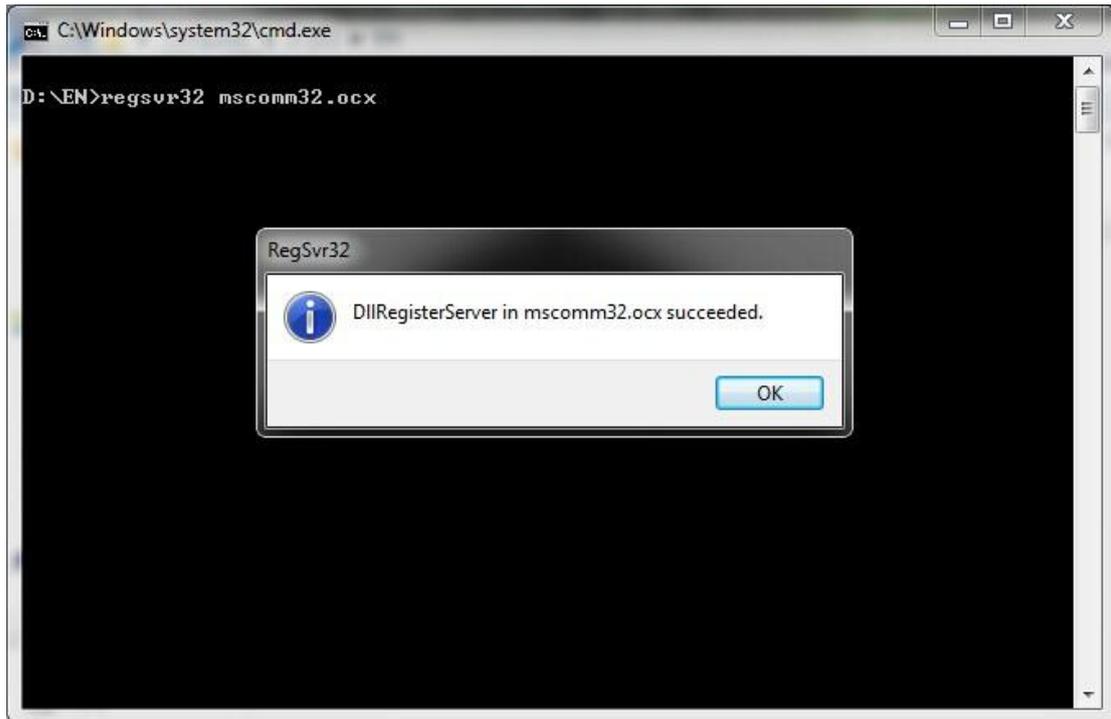
1. Set the pin VCCIO on the CP2102 to 5V;
2. Connect the CP2102 to the E-Paper with jump wire, in which the red wire of the cable is connected to the pin VCCIO on the CP2102, and the black wire to the pin GND, the white wire to the pin RXD, and the green wire to the pin TXD.

CP2102 USB UART Board	4.3inch e-Paper
VCCIO	Red
GND	Black
RXD	White
TXD	Green

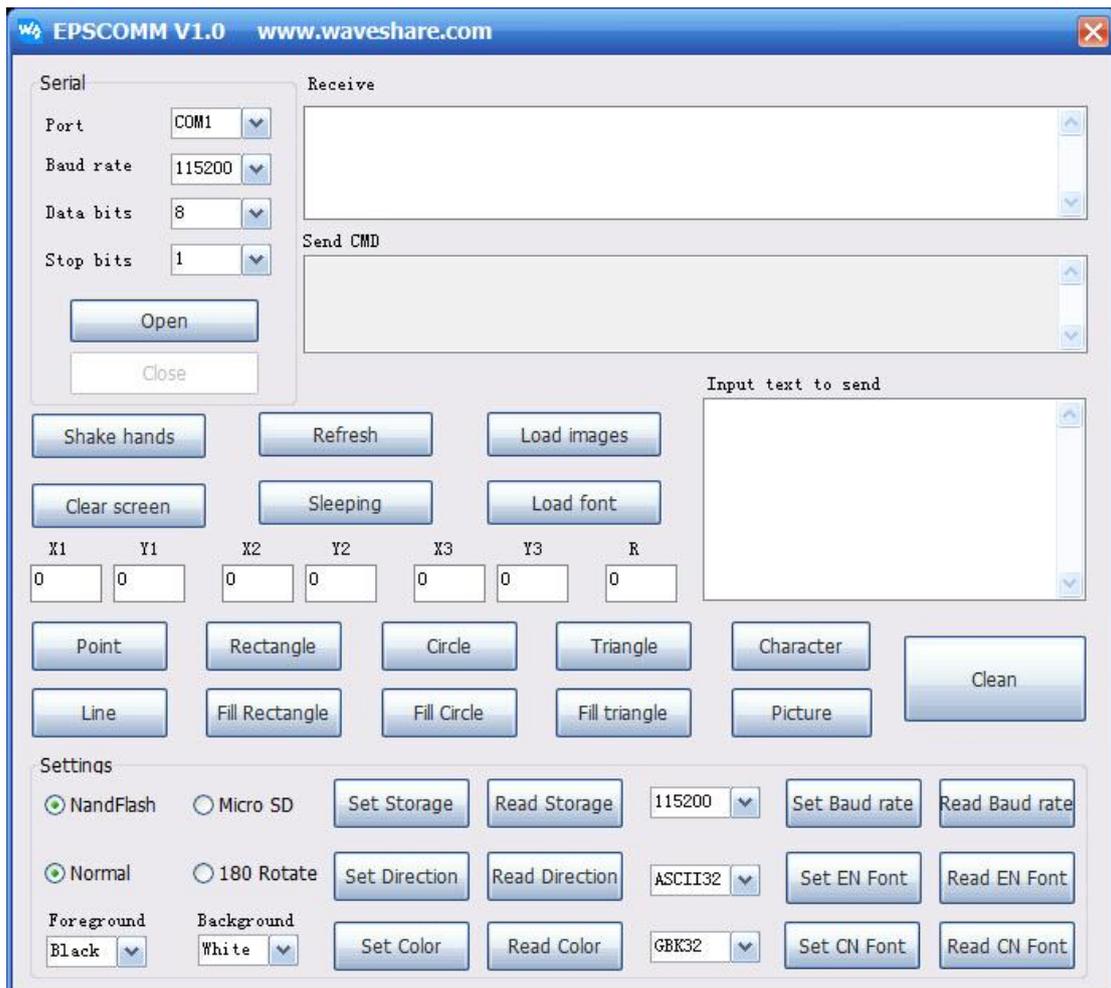
3. Connect the CP2102 to the PC via a USB cable. Now, the E-Paper is powered up, and its state indicator lights up.

- **Software installation**

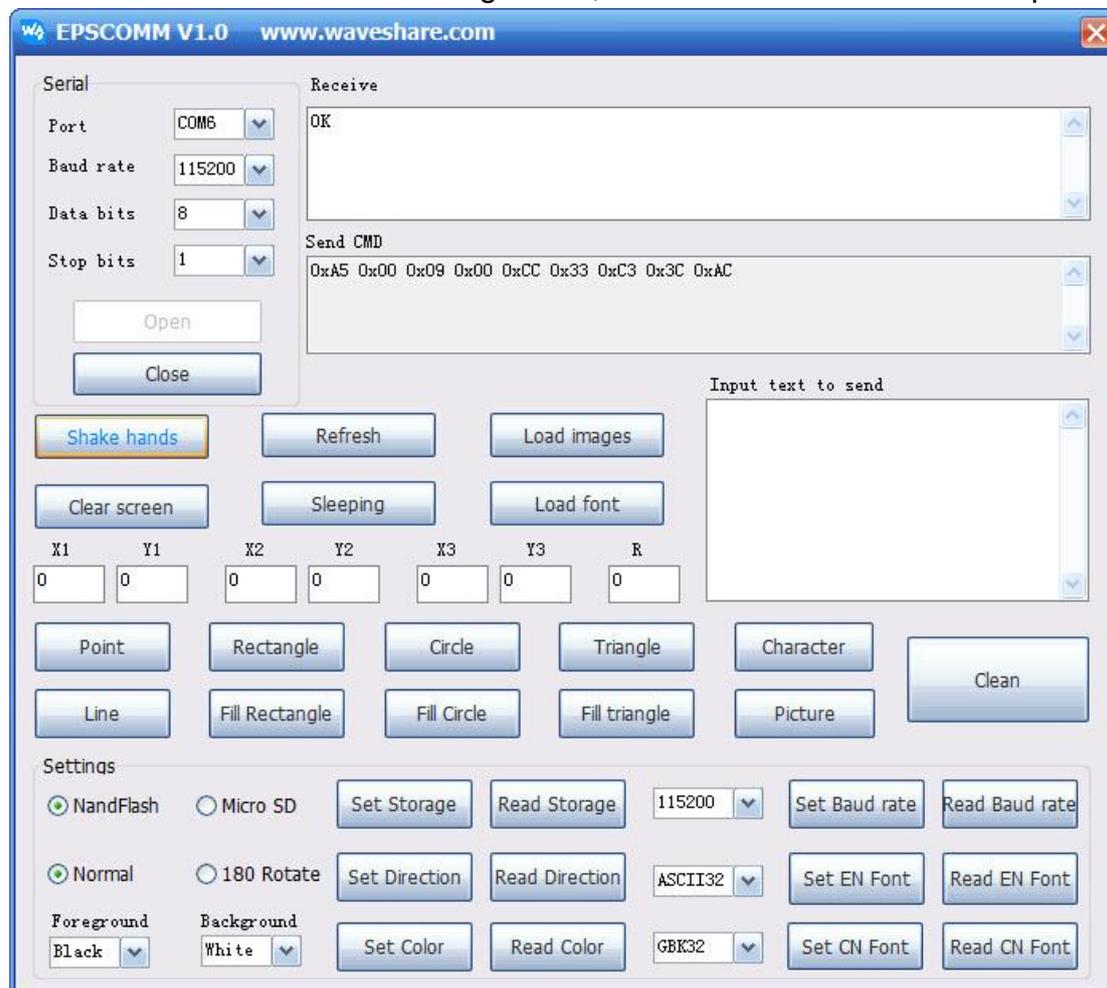
Unzip the software to the installation directory, and then run the file "install.bat" with Administrator . Then, you will see a dialog box will pop up as the figure shows below, if the software is installed successfully.



Double click the **EPSCOMM.exe** to start the software:

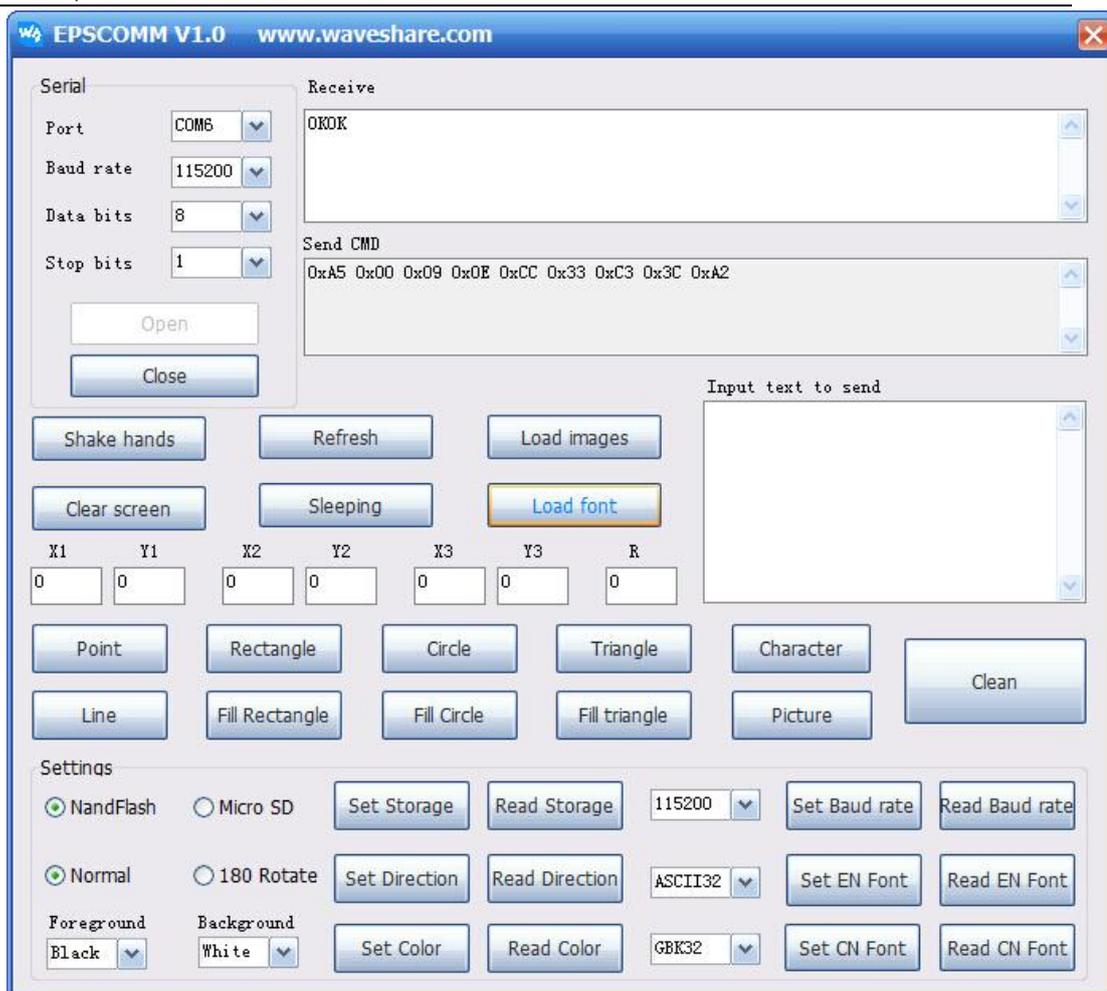


Check the Device Manager, and configure the corresponding serial port. Click the button **Open**, and then the button **Shake hands**. The E-Paper module will return the message “OK”, if the communication is built up.

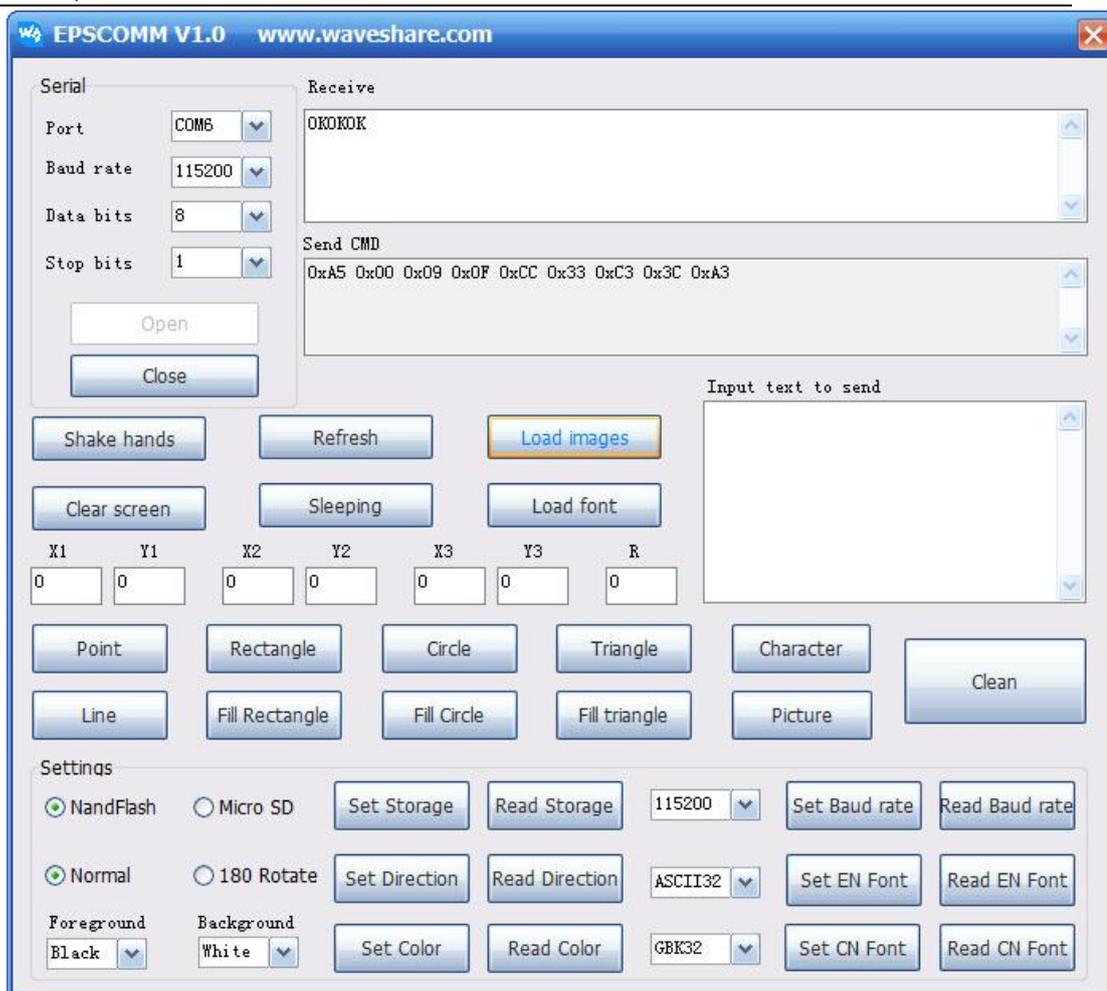


#### ● Font libraries and image importation

1. Prepare a TF card (here we take an 8G TF card as an example), and format the TF card into a FAT32 system with 4096 bytes allocation unit size. Copy the font libraries and image files provided by Waveshare into the TF card. When finished, insert the TF card into the E-Paper.
2. Click the button **Load font**. Then, the state indicator on the e-Paper will flicker 3 times, indicating the importation starts. When finished, the state indicator will flicker 3 times again and the e-Paper module will return the message “OK”, indicating that the importation of font library is completed successfully.



- Click the button **Load image**. Then, the state indicator on the e-Paper will flicker 3 times, indicating the importation starts. When finished, the state indicator will flicker 3 times again and the e-Paper module will return the message "OK", indicating that the importation of image(s) is completed successfully.



Remarks:

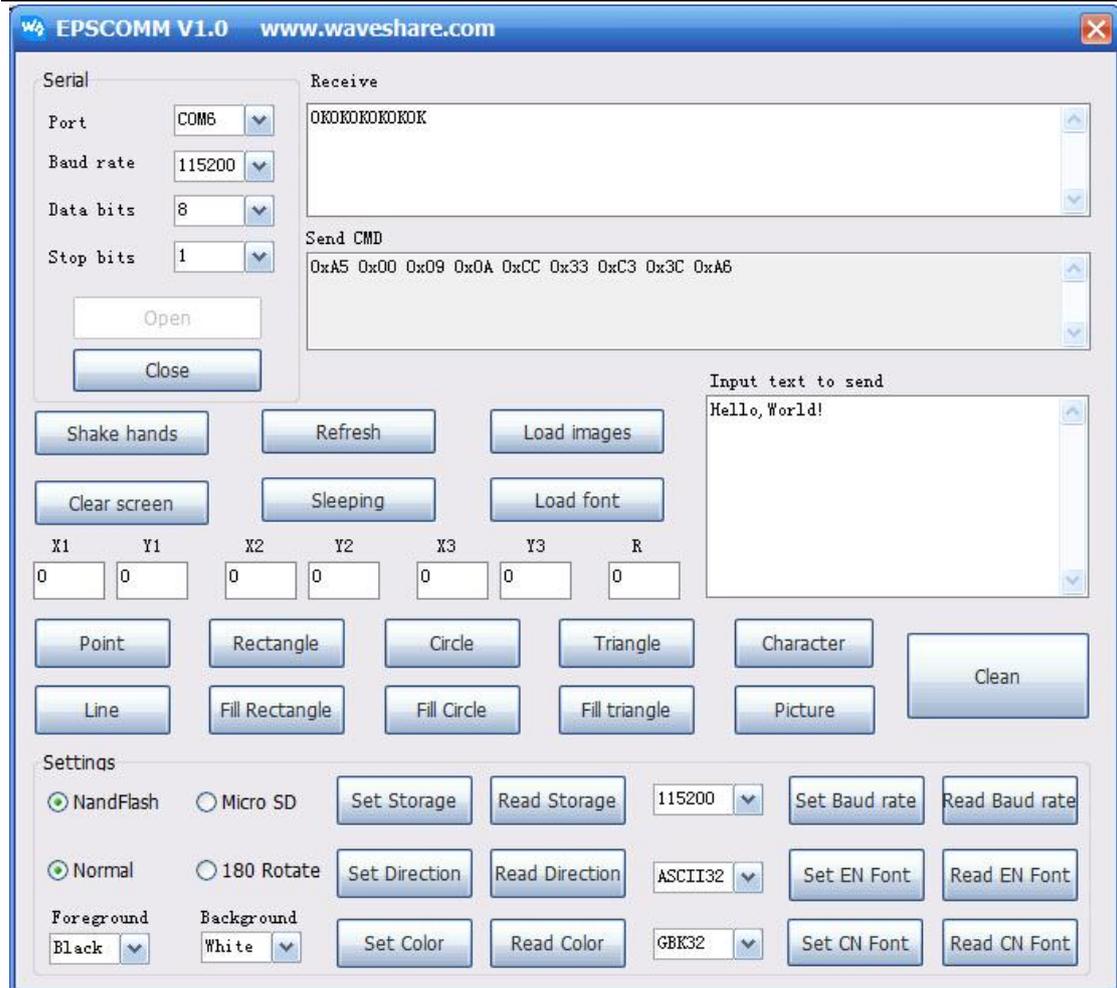
[1] Once you do the operation, please check the state indicator. It flickers when the importation starts. Now, wait the importation finish.

[2] Normally, you only need to import the image(s) you want, but not need to import the font library, since it has been imported by default.

- **Display testing**

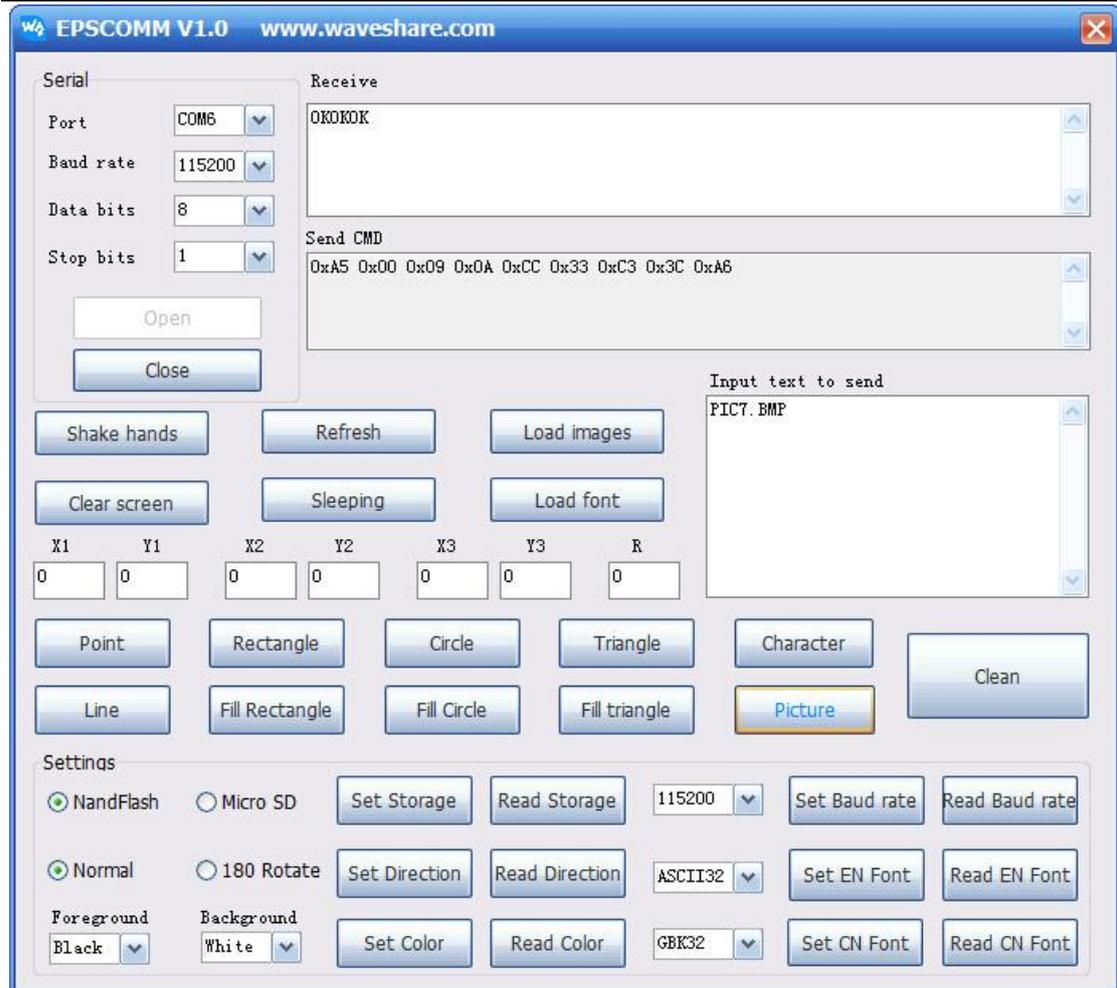
1. Display text

Enter the text "Hello, World!" into the Input box, and set the coordinate options **X1** and **Y1** to 100. Then, click the button **Character** and then the button **Refresh**. Now, you can see the text inputted shows in the screen.



## 2. Display image

Enter the name of the image you want to display into the Input box, and set the coordinate options **X1** and **Y1** to 0. Then, click the button **Picture**, and then the button **Refresh**. Now, you can see the corresponding image shows in the screen.



### 3. Display shape

To draw shapes, you may need to input the parameters listed in the table below:

	Coordinate(X1, Y1)	Coordinate(X2,Y2)	Coordinate(X3, Y3)	Radius (R)
Point	●			
Line	●	●		
Rectangle	●	●		
Filled rectangle	●	●		
Circle	●			●
Filled circle	●			●
Tri-angle	●	●	●	
Filled tri-angle	●	●	●	

### 4. Other settings

- Set the storage area, select the storage locations of font library and images, default location: NandFlash;
- Set the display direction;
- Set the drawing color, the option **Foreground color** for the text is set to Black and the option **Background color** for the background is set to White by default;

- Set Baud rate, default: 15200; it is not recommended to modify this option frequently;
- Set the font size of the English character;
- Set the font size of the Chinese character.

Remarks:

- [1] When finished each command, there will be corresponding response. Normally, it will return "OK";
- [2] For drawing point(s) or line(s), or displaying text(s) or image(s), you should click the button **Refresh** to update the display on the E-Paper screen whenever you finish each drawing.
- [3] You may need to wait the response for a while after clicking the button **Display image**, since the image data is large.

## 2.3. Arduino

### 1. Hardware connection

Here, we take Arduino UNO R3 development board as an example to illustrate the application. Connect the development board to the e-Paper with a serial cable as shown in the follow table. For more information about this development board, please refer to the Appendix.

Connection between Arduino UNO R3 and 4.3inch e-Paper:

Arduino UNO R3	4.3inch e-Paper
5V	Red
GND	Black
RX/D0	White
TX/D1	Green
D2	Yellow
Reset, you can keep it unconnected	Blue

### 2. Program compiling and uploading

Open the Arduino program on the Demo. Then, compile and upload it. Please take a note that the white cable should be removed on program uploading.



### 3. Serial communication protocol

#### 3.1. Basic rules

##### 3.1.1. Serial port

This e-Paper module adopts full-duplex asynchronous UART communication with adjustable Baud rate.

##### 3.1.2. Command frame format

Command	0xA5	0xFF XX	0xFF	0xFF...	0xCC 33 C3 3C	0xFF
<b>Meaning (Description)</b>	Frame header (1 byte)	Frame length <sup>[1]</sup> (2 bytes)	Command Type (1 byte)	Parameter or data (0~1024 bytes)	Frame end (fixed 4 bytes)	Parity (1 byte) <sup>[2]</sup>

[1] The frame length is calculated by the length begin with the frame header and end with the parity byte.

[2] Parity byte is the final result of the XOR operation on the bytes from the frame header to the frame end.

##### 3.1.3. Transmission sequence

The data of the module is transmitted in the Network byte sequence, which means higher byte is sent at first, then lower byte following. For example, a parameter, 0x1234, is transmitted in two parts: 0x12 is sent at first, and then 0x34 following.

##### 3.1.4. Command parameters

- Byte: 8 bits, single byte;
- Short: 16 bits, double bytes;
- Dword: 32 bits, four bytes;
- String: the character string end with "0".

##### 3.1.5. Color definition

This e-Paper module supports 4-level gray display, and 2 bits for each pixel.

0x00: Black

0x01: Dark gray

0x02: Light gray

0x03: White

## 3.2. Command set

### 3.2.1. Command look-up table

Functions	Frame length	Command	Command parameter	Descriptions
System control	0x0009	0x00	Null	Handshake command. If the module is ready, it will return an "OK".
	0x000D	0x01	Dword: Set the serial Baud rate	Set the serial Baud rate
	0x0009	0x02	Null	Return the current Baud rate value in ASCII format
	0x0009	0x06	Null	Return the information about the currently used storage area. 0: NandFlash 1: MicroSD
	0x000A	0x07	Byte: Set the storage area	0x00: NandFlash 0x01: MicroSD
	0x0009	0x08	Null	System enters sleep mode
	0x0009	0x0A	Null	Refresh and update the display at once
	0x0009	0x0C	Null	Return the current display direction 0: Normal 1: 180° rotation
	0x000A	0x0D	Byte: Set the display direction	0x00: Normal 0x01: 180° rotation
	0x0009	0x0E	Null	Import font library: 48MB
	0x0009	0x0F	Null	Import image: 80MB
	Display parameter configuration	0x000B	0x10	Byte0: foreground color Byte1: background color
0x0009		0x11	Null	Return the current drawing color
0x0009		0x1C	Null	Read the English font size 1: 32 dots matrix 2: 48 dots matrix 3: 64 dots matrix
0x0009		0x1D	Null	Read the Chinese font size 1: 32 dots matrix 2: 48 dots matrix 3: 64 dots matrix
0x000A		0x1E	Byte0: Set the English	Set the English font size

			font size	0x01: 32 dots matrix 0x02: 48 dots matrix 0x03: 64 dots matrix
	0x000A	0x1F	Byte0: Set the Chinese font size	Set the Chinese font size 0x01: 32 dots matrix 0x02: 48 dots matrix 0x03: 64 dots matrix
Basic drawings	0x000D	0x20	Short0: coordinate x Short1: coordinate y	Display a point on a given coordination Default: foreground color
	0x0011	0x22	Short0: coordinate x0 Short1: coordinate y0 Short2: coordinate x1 Short3: coordinate y1	Draw a line
	0x0011	0x24	Short0: coordinate x0 Short1: coordinate y0 Short2: coordinate x1 Short3: coordinate y1	Fill a rectangle
	0x0011	0x25	Short0: coordinate x0 Short1: coordinate y0 Short2: coordinate x1 Short3: coordinate y1	Draw a rectangle
	0x000F	0x26	Short0: coordinate x0 Short1: coordinate y0 Short2: r radius	Draw a circle
	0x000F	0x27	Short0: coordinate x0 Short1: coordinate y0 Short2: r radius	Fill a circle
	0x0015	0x28	Short0: coordinate x0 Short1: coordinate y0 Short2: coordinate x1 Short3: coordinate y1 Short4: coordinate x2 Short5: coordinate y2	Draw a tri-angle
	0x0015	0x29	Short0: coordinate x0 Short1: coordinate y0 Short2: coordinate x1 Short3: coordinate y1 Short4: coordinate x2 Short5: coordinate y2	Fill a tri-angle
	0x0009	0x2E	Null	Clear the screen with background color
Display text	0x0D + string length	0x30	Short0: coordinate x Short1: coordinate y	1. Support English and Chinese fonts mixed display

			String: bitmap name	2. Maximum string length = 1024 – 4
Display image (bitmap Only)	0x0D + string length	0x70	Short0: coordinate x0 Short1: coordinate y0 String: .BMP file name	Support bitmap file: 1bpp/2bpp

Remarks:

- [1] The string length is the length of the character string end with “0”, in which the ending “0” is included.
- [2] The name of the image should be in uppercase English character(s) with the length less than 10 characters (the symbol “.” is included). And the string length of the image name should be less than 11 characters, in which the ending “0” is included.

### 3.2.2. Command explanations

- **System control**

1. Handshake command (0x00)

After powered up, you can send this command to the terminal to make sure whether the terminal is ready and able to receive commands or data.

**Example:** A5 00 09 00 CC 33 C3 3C AC

**Descriptions:** Handshake command

2. Set the Baud rate (0x01)

After powered up, the default Baud rate is 115200. This command is used to set the Baud rate. You may need to wait 100ms for the module to return the result after sending this command, since the host may take a period of time to change its Baud rate.

**Example:** A5 00 0D 01 00 00 25 80 CC 33 C3 3C 0C

**Descriptions:** 0x2580: set the Baud rate to 9600. The parameter of Baud rate is consisted of 4 bytes, so you need to complete the rest by yourself.

3. Read the current Baud rate (0x02)

**Example:** A5 00 09 02 CC 33 C3 3C AE

**Descriptions:** Read the current Baud rate of the system.

4. Return the information about the currently used storage area (0x06)

**Example:** A5 00 09 06 CC 33 C3 3C AA

**Descriptions:** When returns 0x00, the currently used storage area is internal NandFlash; when returns 0x01, the currently used storage area is external TF card.

5. Set the storage area (0x07)

Set the storage area to select the storage locations of font library and images, either the external TF card or the internal NandFlash is available.

**Example:** A5 00 0A 07 00 CC 33 C3 3C A8

**Descriptions:** 0x00: set the currently used storage area as NandFlash.

## 6. Sleep mode (0x08)

The system will enter the sleep mode and reduce system power consumption by this command. Under sleep mode, the state indicator is off, and the system does not respond any commands. Only the rising edge on the pin WAKE\_UP can wake up the system.

**Example:** A5 00 09 08 CC 33 C3 3C A4

**Descriptions:** Switch the system to sleep mode.

## 7. Refresh (0x0A)

Refresh and update the display on the screen. After the host send out the drawing command, the system will only write the relative data into the internal video memory, but not update the display on the screen. So you should operate this command whenever you finish each drawing.

**Example:** A5 00 09 0A CC 33 C3 3C A6

**Descriptions:** Refresh and update the display on the screen

## 8. Get the display direction (0x0C)

**Example:** A5 00 09 0C CC 33 C3 3C A0

**Descriptions:** When returns 0, the screen is in normal display; when returns 1, the display on the screen will be rotated by 180°.

## 9. Set the display direction (0x0D)

Set the display direction, only 180° rotation display supported.

**Example:** A5 00 0A 0D 01 CC 33 C3 3C A3

**Descriptions:** 0x01: The display on the screen will be switch from normal display to 180° rotation display.

## 10. Import font library (0x0E)

Import the font library files from the TF card to the internal NandFlash. The font library files include GBK32.FON/GBK48.FON/GBK64.FON. The state indicator will flicker 3 times when the importation is start and ending.

**Example:** A5 00 09 0E CC 33 C3 3C A2

**Descriptions:** Import the font library

## 11. Import image (0x0F)

Import the image files from the TF card to the internal NandFlash. The state indicator will flicker 3 times when the importation is start and ending.

**Example:** A5 00 09 0F CC 33 C3 3C A3

**Descriptions:** Import the images

### ● Display parameter configuration

#### 1. Set the pallet (0x10)

Set the foreground color and the background color on drawing, in which the foreground color can be used to display the basic drawings and text, while the background color is used to clear the screen.

**Example:** A5 00 0B 10 00 03 CC 33 C3 3C BD

**Descriptions:** 0x00: The foreground color is set to Black  
 0x03: The background color is set to White

#### 2. Get the pallet data (0x11)

**Example:** A5 00 09 11 CC 33 C3 3C BD

**Descriptions:** For example, when returns “03”, “0” means the foreground color is Black and “3” means the background color is White.

3. Read the English font size (0x1C)

**Example:** A5 00 09 1C CC 33 C3 3C B0

**Descriptions:** Return the index of currently used English font size.

4. Read the Chinese font size (0x1D)

**Example:** A5 00 09 1D CC 33 C3 3C B1

**Descriptions:** Return the index of currently used Chinese font size.

5. Set the English font size (0x1E)

Set the English font size, currently support 32/48/64 dots matrix.

**Example:** A5 00 0A 1E 01 CC 33 C3 3C B0

**Descriptions:** 0x01: Set the English font size to 32 dots matrix

6. Set the Chinese font size (0x1F)

Set the Chinese font size, currently support 32/48/64 dots matrix.

**Example:** A5 00 0A 1F 01 CC 33 C3 3C B1

**Descriptions:** 0x01: Set the Chinese font size to 32 dots matrix

● **Basic drawings**

1. Draw a point (0x20)

Display a point on a specified coordination position with foreground color.

**Example:** A5 00 0D 20 00 0A 00 0A CC 33 C3 3C 88

**Descriptions:** Display a point on the position (0x0A, 0x0A) with foreground color.

2. Draw a line (0x22)

Draw a line between two specified point coordinates with foreground color.

**Example:** A5 00 11 22 00 0A 00 0A 00 FF 00 FF CC 33 C3 3C 96

**Descriptions:** Draw a line between the point coordinates (0x0A, 0x0A) and (0xFF, 0xFF) with foreground color.

3. Fill a rectangle (0x24)

Fill a rectangle according to two point coordinates with foreground color, in which these two points serve as the diagonal points of the rectangle.

**Example:** A5 00 11 24 00 0A 00 0A 00 FF 00 FF CC 33 C3 3C 90

**Descriptions:** Fill a rectangle according to the point coordinates (0x0A, 0x0A) and (0xFF, 0xFF), in which these two points serve as the diagonal points of the rectangle.

4. Draw a rectangle (0x25)

Draw a rectangle according to two point coordinates with foreground color, in which these two points serve as the diagonal points of the rectangle.

**Example:** A5 00 11 25 00 0A 00 0A 00 FF 00 FF CC 33 C3 3C 91

**Descriptions:** Draw a rectangle according to the point coordinates (0x0A, 0x0A) and (0xFF, 0xFF), in which these two points serve as the diagonal points of the rectangle.

5. Draw a circle (0x26)

Draw a circle based on the given center coordination and radius.

**Example:** A5 00 0F 26 00 FF 00 FF 00 80 CC 33 C3 3C 0C

**Descriptions:** Draw a circle with the center coordination (0xFF, 0xFF), and radius 0x80.

6. Fill a circle (0x27)

Fill a circle based on the given center coordination and radius.

**Example:** A5 00 0F 27 00 FF 00 FF 00 80 CC 33 C3 3C 0D

**Descriptions:** Fill a circle whose center coordination is (0xFF, 0xFF) and radius is 0x80.

7. Draw a tri-angle (0x28)

Draw a tri-angle according to three given point coordinates.

**Example:** A5 00 15 28 00 0A 00 0A 00 20 00 80 00 80 00 FF CC 33 C3 3C 47

**Descriptions:** The three point coordinates of the tri-angle are (0x0A, 0x0A), (0x20, 0x80) and (0x80, 0xFF).

8. Fill a tri-angle (0x29)

Fill a tri-angle according to three given point coordinates.

**Example:** A5 00 15 29 00 0A 00 0A 00 20 00 80 00 80 00 FF CC 33 C3 3C 46

**Descriptions:** The three point coordinates of the tri-angle are (0x0A, 0x0A), (0x20, 0x80), (0x80, 0xFF)

9. Clear the screen (0x2E)

Clear the screen with the background color.

**Example:** A5 00 09 2E CC 33 C3 3C 82

**Descriptions:** Clear the screen

- **Display text**

1. Display character string (0x30)

Display a character string on a specified coordination position. Chinese and English mixed display is supported.

**Example:** A5 00 17 30 00 0A 00 0A C4 E3 BA C3 57 6F 72 6C 64 00 CC 33 C3 3C 9E

**Descriptions:** 0x0A, 0x0A: This is the given coordination position

0xC4E3:你

0xBAC3:好

0x57 6F 72 6C 64:World

After executed this command, the screen will show the text:  
你好 World

Each character string should be end with a "0". So, you should add a "00" at the end of the string C4 E3 BA C3 57 6F 72 6C 64.

- **Display image**

1. Display bitmap (0x70)

Before executing this command, please make sure the bitmap file you want to display is stored in the storage area (either TF card or internal NandFlash).

**Example:** A5 00 16 70 00 00 00 00 50 49 43 37 2E 42 4D 50 00 CC 33 C3  
3C DF

**Descriptions:** Image start coordination position: (0x00, 0x00)

0x50 49 43 37 2E 42 4D 50: Bitmap name: PIC7.BMP

Each character string should be end with a "0". So, you should add a "00" at the end of the string 50 49 43 37 2E 42 4D 50.

The name of the bitmap file should be in uppercase English character(s). And the string length of the bitmap name should be less than 11 characters, in which the ending "0" is included. For example, PIC7.BMP and PIC789.BMP are correct bitmap names, while PIC7890.BMP is a wrong bitmap namem.