ESL image production

The cloud server cannot directly send standard picture to ESL. Before the image is sent to the ESL, the cloud server should translate the standard picture to ESL json format file. This document tell describe how to translate standard picture file to ESL json file..

In the test phase, you can manually create an image of the electronic label. The method of making it is as follows:

1. Make BMP format pictures

2. Converted BMP format picture to a JSON message.

# Make Black/white Color BMP format pictures

If the ESl only support black/white color, you need to make a pure black and white picture with the right resolutions.

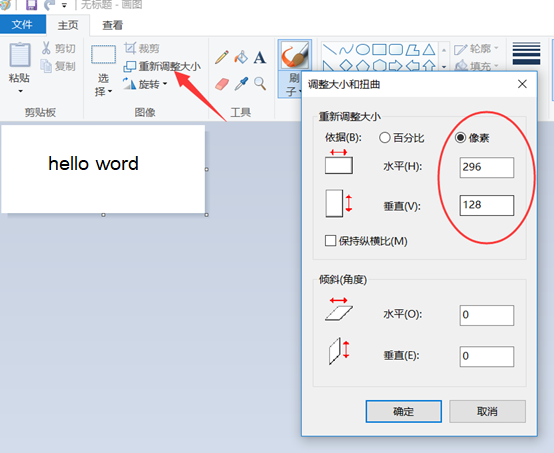
|  |  |
| --- | --- |
| ESL type | resolutions |
| 2.1inch ESL | 212 \* 104 |
| 2.2inch ESL | 250 \* 122 |
| 2.9inch ESL | 296 \* 128 |
| 4.2inch ESL | 400 \* 300 |

The following example uses the windows drawing tool to generate 2.9inch ESL picture:

1, Windows attachment menu = "painting



2. set the resolution to 296\*128



3. You can add text on the picture. For example: hello word

4. Save picture. Please select the single color bitmap to save.



# Make Black/white/red BMP format pictures

This screen does not support gray scale.

If the ESl only support black/white/red color, you can make a three color picture with the right resolutions.

Note: The picture must only include following three color:

* pure red (RGB: 255,0,0);
* pure black (RGB: 0,0,0);
* pure white (255,255,255) ;

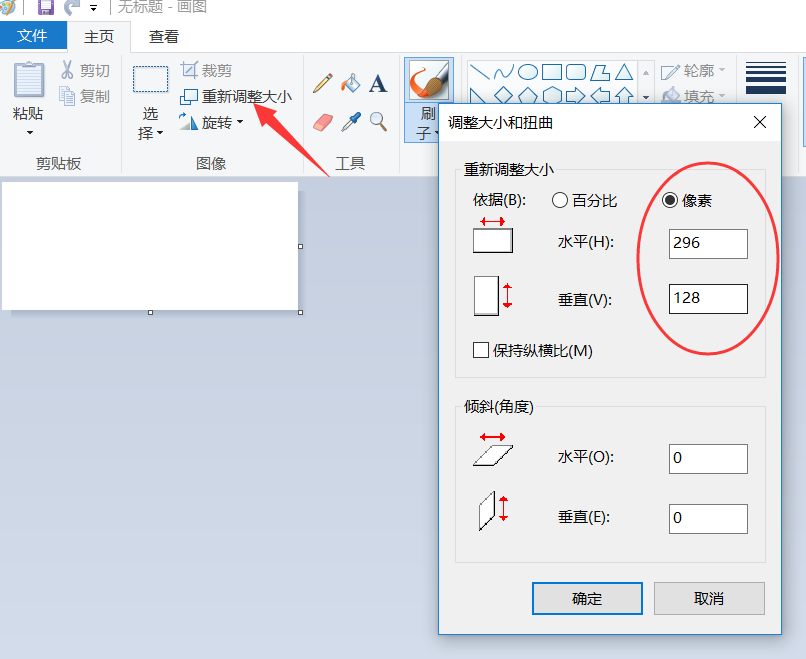
|  |  |
| --- | --- |
| ESL type | resolutions |
| 2.1inch ESL | 212 \* 104 |
| 2.2inch ESL | 250 \* 122 |
| 2.9inch ESL | 296 \* 128 |
| 4.2inch ESL | 400 \* 300 |

The following example uses the windows drawing tool to generate 2.9inch three color ESL picture:

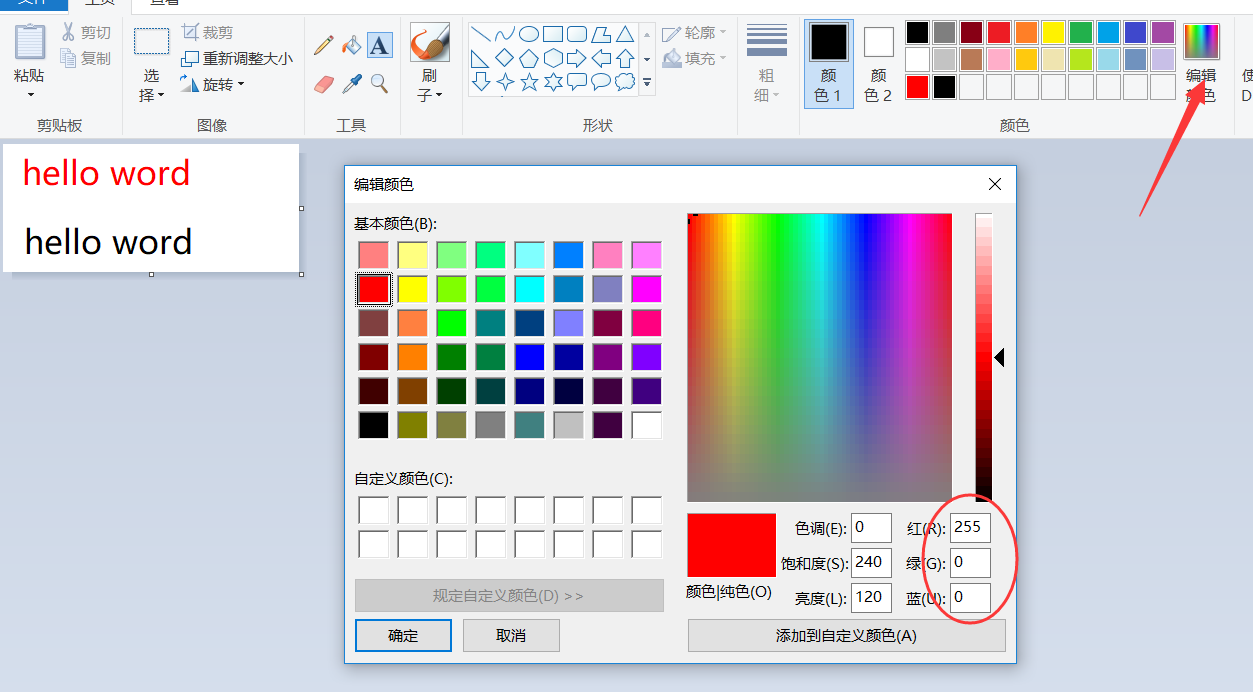
1, Windows attachment menu = "painting



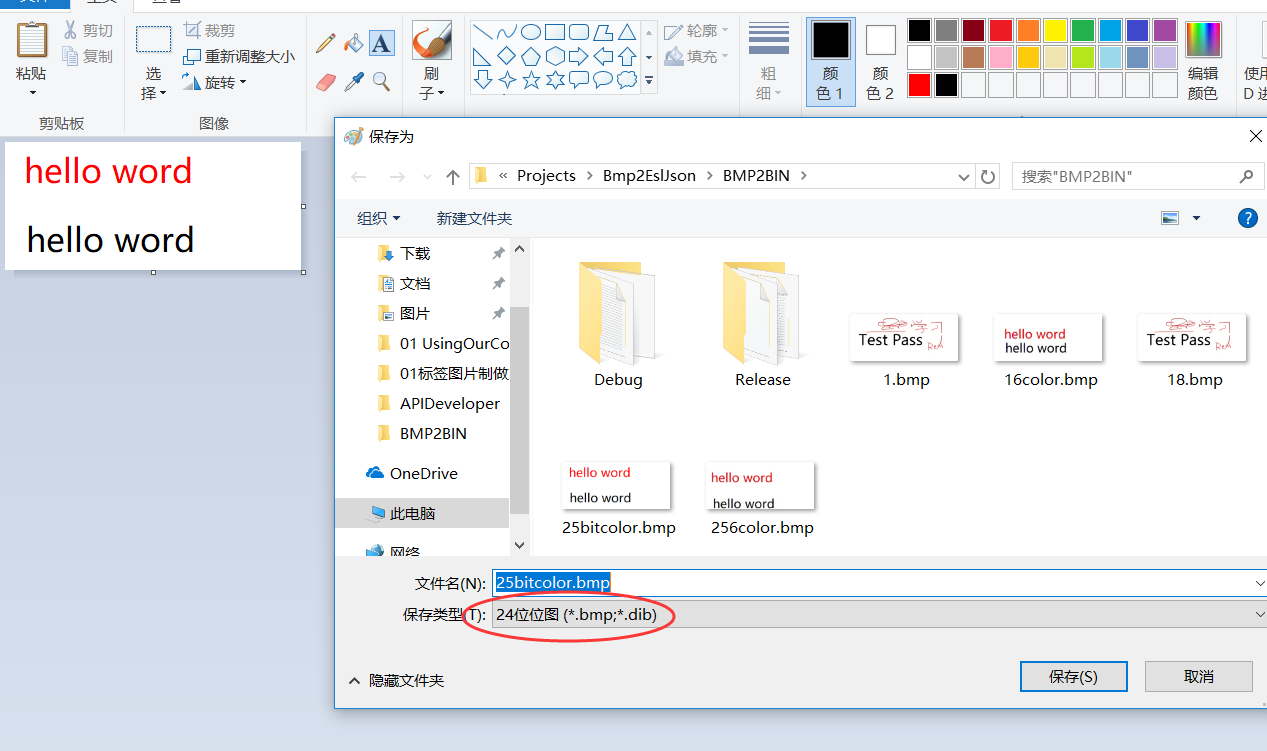
2. set the resolution to 296\*128



3, add text on the picture with a brush, you need to ensure that it is pure white (255, 255, 255); pure red: (255, 0, 0); pure black (0, 0, 0), you can choose in the editing color.



4. Save the file. Please select the single 24 color bitmap to save.



# Converted BMP format picture to a JSON message

## Step By Step to generate JSON message

1. Please put the BMP picture files generated in the previous step and Bmp2EslJson.exe in the same directory.

2, run the Bmp2EslJson.exe tool, the parameters you need to enter are as follows:

1. BIN file name.
2. ESL type:

* e21: 2.1inch ESL
* e29: 2.9inch ESL
* e42: 4.2inch ESL
* e00: partial picture

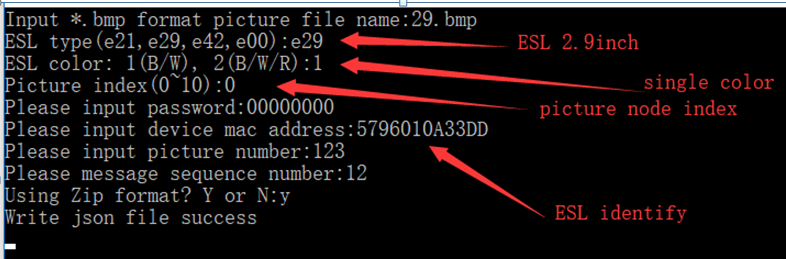
1. The password of the tag, the default is 8 ascii zero. “00000000”;
2. The mac address of the ESL can be viewed through the gateway status monitoring to the 12-bit mac address.
3. The ID of the picture, you can enter a number arbitrarily, this is equivalent to the number of this picture.
4. JSON Message ID: identify about the json message, every download message should be different.
5. Picture Node index: every picture stored in ESL has node index.Please reference <Gateway API Design Specification> for details.
6. Picture number: when cloud server download an picture to ESL, the cloud can assign an picture number for the picture.

e) Whether compression: y

Compressing the image and then sending it will greatly reduce the refresh time.

After the carriage return, you will be prompted to write the JSON file successfully. You can see that a json file is generated in the same directory, which is the image update message.

Bmp2EslJson.exe is based on Visual Studio 2019 written in C/C++ language.



## One command to generate JSON message

1. Please put the BMP picture files generated in the previous step and Bmp2EslJson.exe in the same directory.

2. Execute following command

bmp2esljson -f 21\_example.bmp -t e21 -c 1 -fd 0 -p 00000000 -m A1A2A3A4A5A6 -id 123 -s 148 -z y

After the command is executed successfully, a JSON file will be generated. If the execution fails, a prompt error will be generated.

Paramaters:

-f: file name

-t: esl type: e21, e29, e42

-c: color type; 1: single color 2: three color

-fd: picture node index;

-p: password

-m: mac address

-id: picture id

-s: json message id

-z: compress mode, y means compress, n means not compress