

2022/7/8

2022年7月8日 15:34

完成ESP32-pico-d4最小系统测试板绘制并投板
淘宝购买芯片工具等，花费约450

2022/7/11

2022年7月12日 18:38

查询到免费天气api: [天气免费 API | 天气API接口 | 天气JSON API | 天气XML API | JSON在线免费 API \(sojson.com\)](#)

成都天气查询url为:

<http://t.weather.sojson.com/api/weather/city/101270101>

来自 <<https://www.sojson.com/api/weather.html>>

101270101

来自 <<https://github.com/baichengzhou/weather.api/blob/master/src/main/resources/citycode-2019-08-23.json>>

查询到免费ip获取api: [免费开源的国外ip获取国家城市接口 - 简书 \(jianshu.com\)](#)

2022/7/12

2022年7月12日 18:43

焊接最小系统，目前发现CP2102为非官方VIP和PID，通过修改驱动后可以识别，未测试是否工作正常

截止23:47发现 GPIO12 GPIO4 虚焊，复位键无用，怀疑1K电阻不足以拉低

目前最小系统烧录程序时connecting超时，目前不清楚具体是什么原因导致的

- 问题解决，使用默认波特率460200，任意修改都可能卡住

```
Flash: [===== ] 52.4% (used 168160 bytes from 327680 bytes)
Flash: [===== ] 57.2% (used 1797952 bytes from 3145728 bytes)
Configuring upload protocol...
AVAILABLE: esp-prog, espota, esptool, iot-bus-jtag, jlink, minipro, olimex-arm-usb-ocd, olimex-arm-usb-jtag
CURRENT: upload_protocol = esptool
Looking for upload port...
Auto-detected: COM7
Uploading .pio\build\HoloCubic AIO\firmware.bin
esptool.py v3.1
Serial port COM7
Connecting.....

A fatal error occurred: Failed to connect to ESP32: Timed out waiting for packet header
*** [upload] Error 2
===== [FAILED] Took 48.00s
Default (only _lvgltest_V1.1_ usedtest)  Auto  tabnine starter
```

2022/7/13

2022年7月13日 19:30

企图焊好虚焊点结果拆下来又烧不进去了 ￣へ￣

换了个芯片有焊了一天，重新获得了以下结论：

- 新芯片基本引脚焊好了，设置波特率115200是能获得报文的，如下图

```
rst:0x10 (RTCWDT_RTC_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
flash read err, 1000
ets_main.c 371
ets Jun  8 2016 00:22:57
```

- 如果没焊好，可能以16进制显示回汇报，目前发现可能会回报：
- 0xFF：不知道啥意思意思
- 0x00：不知道意思意思

同时QFN封装直接在pcb上上一堆锡和助焊剂然后吹就行

第一块esp32确认报废，猜测在夹具上的时候上电导致所以IO电压都不正常，也无法识别

目前焊好两块最小系统，焊接成功率非常高(￣▽￣)~

之前的ESP8266_WebServer程序移植到ESP32，测试成功，WiFi与天线无问题

```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 188777542, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1184
load:0x40078000,len:12784
load:0x40080400,len:3032
entry 0x400805e4

.....
Connected tossid
IP address:192.168.1.19
MDNS responder started
HTTP server started
successful || system_time = 5.00s
successful || system_time = 10.00s
```

2022/7/14

2022年7月14日 11:42

[10分钟搞定Visual Studio Code配置C/C++开发环境 - 二的次方 - 博客园 \(cnblogs.com\)](#)

安装了c++环境，虽然感觉好像不完全正常，但是能正常hello world

成功使用VS2019打开lvgl simulation软件

使用TFT_eSPI库，屏幕测试可用，暂未测试lvgl库，lvgl的仿真需要在VS（不是vscode）环境使用



焊接完成mpu,tf卡,CNA总线 暂未测试

2022/7/15

2022年7月15日 20:07

测试mpu6050, IIC通讯失败, 死在第一个IIC通讯程序中

```
4 //  
5 void MPU6050::initialize() {  
6     Serial.println("aaaa");  
7     setClockSource(MPU6050_CLOCK_PLL_XGYRO);  
8     Serial.println("bbbb");  
9     setFullScaleGyroRange(MPU6050_GYRO_FS_250);  
0     setFullScaleAccelRange(MPU6050_ACCEL_FS_2);  
1     setSleepEnabled(false); // thanks to Jack Elst  
2 }
```

2022/7/17

2022年7月17日 17:32

Mpu6050新焊了一块，仍然同样问题

测试SD卡，初始化失败，测试硬件连接正常

目前怀疑两个问题均出在软件，这两个部分在芯片复位后都会在串口报错

```
ets Jun  8 2016 00:22:57
rst:0x1 (POWERON RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
configsip: 188777542, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1184
load:0x40078000,len:12784
load:0x40080400,len:3032
entry 0x400805e4
[  2][E][esp32-hal-gpio.c:95] _??: Invalid pin selected
E (181) gpio: gpio_set_level(226): GPIO output gpio_num error
[ 1031][E][sd_diskio.cpp:802] sdcard_mount(): f_mount failed: (3) The physical drive
cannot work
[ 1532][E][sd_diskio.cpp:126] sdSelectCard(): Select Failed
Card Mount Failed
```

2022/7/19

2022年7月19日 23:43

不断排查IIC到MPU6050问题，目前已知如下情报：

1. 可通过IIC读取到MPU，即ESP到MPU是可以通讯的!!!

```
Serial.println("Initializing I2C devices...");
for (address = 1; address < 127; address++)
{
    Wire.beginTransmission(address);
    error = Wire.endTransmission();
    if (error == 0)
    {
        Serial.print("I2C device found at address 0x");
        Serial.println(address, HEX);
        nDevices++;
    }
}
```

```
Initializing I2C devices...
I2C device found at address 0x68
```

2. 同时，例程的初始化程序仍然会卡死，目前排查卡死在Wire.cpp的
xSemaphoreTake(lock, portMAX_DELAY) != pdTRUE 中
3. 发现2问题中的函数运行在FREERTOS中的宏定义中，并且Wire.h中并没有关于该宏定义的
声明，在Wire.h中自行添加声明后不再卡死

```
#if !CONFIG_DISABLE_HAL_LOCKS
//acquire lock
if(lock == NULL || xSemaphoreTake(lock, portMAX_DELAY) != pdTRUE){
    log_e("could not acquire lock");
    return 0;
}
#endif
```

4. 某种意义上来说已经可以用了，而且焊接很简单，数据也稳定，就是不知道正确取消
FREERTOS的办法是什么

```
a/g: -304 196 16204 -86 506 -4
a/g: -400 160 15976 -67 204 -21
a/g: -288 132 16188 -50 200 -26
a/g: -404 232 16024 -60 475 -61
a/g: -348 172 16164 -55 49 -8
a/g: -160 216 16204 -91 573 -67
a/g: -440 216 16052 -79 174 -4
a/g: -312 188 16296 -85 220 -22
a/g: -368 184 16100 -69 544 -30
a/g: -396 164 16140 -65 -10 -27
a/g: -264 216 16276 -50 502 -45
a/g: -352 164 16000 -64 314 -39
a/g: -436 252 16420 -48 149 0
```

5. 重新更正，只有size_t TwoWire::requestFrom(uint16_t address, size_t size, bool
sendStop)函数中的if(lock == NULL || xSemaphoreTake(lock, portMAX_DELAY) !=
pdTRUE)会卡死，并且这句话在别的函数中也会调用，已经有点颠覆认知了，不知道具
体是哪的bug
6. 重大突破，虽然不知道为啥，发现是I2Cdev.cpp稚晖君的库中这个函数只有begin，新增
一个end就正常运行了，目前采用的做法是直接删去这组begin和end
!!!!!!!!!!!!!!!!!!!! (≡_≡)💎


```
PIO Home  MPU6050.cpp  I2Cdev.cpp  Wire.cpp  portmacro.h  Wire.h  semphr.h  MPU6050.h  I2C
I2Cdev.cpp > readBytes(uint8_t, uint8_t, uint8_t, uint8_t, uint16_t)
// smaller chunks instead of all at once
for (uint8_t k = 0; k < length; k += min((int)length, BUFFER_LENGTH)) {
    Wire.beginTransmission(devAddr);
    Wire.write(regAddr);
    Wire.endTransmission();
    Wire.beginTransmission(devAddr);
    Wire.endTransmission(); //这是加上去了，就可以用了！！
    Wire.requestFrom(devAddr, (uint8_t)min(length - k, BUFFER_LENGTH));
    for (; Wire.available() && (timeout == 0 || millis() - t1 < timeout); count++) {
        data[count] = Wire.read();
        #ifdef I2CDEV_SERIAL_DEBUG
            Serial.print(data[count], HEX);
            if (count + 1 < length) Serial.print(" ");
        #endif
    }
}
```

7. 结论，IICdev通讯中begin和end需要成对调用，缺失end会导致后续调用卡死在if(lock == NULL || xSemaphoreTake(lock, portMAX_DELAY) != pdTRUE)中

2022/7/20

2022年7月20日 11:14

随机二次元图片api:iw233.cn/api/Random.php

主站: <https://iw233.cn/main.html>

sd_card问题排查成功

HoloCubic使用的HSPI引脚为13/14/15/26

其中默认MISO引脚为12, 需要使用GPIO矩阵转换IO

在SD.cpp中将spi.begin(14,26,13,15);手动输入使用IO可使用

同时查询资料显示, 使用矩阵转换的IO速率会下降, 目前的得知的说法是正常80MHz, 转换后40MHz

但SD.cpp为系统库文件, 还需研究怎么通过私有文件修改IO使用

```
SD Card Type: SDHC  
SD Card Size: 3781MB
```

已修改sd_card.cpp, 可通过

```
tf.init(HSPI,14,26,13,15);
```

实现SPI通道设置和IO配置

2022/7/21

2022年7月21日 15:29

通过解决的IMU和SD卡，修改cubic程序，实现了这两个部分的运行，但仍存在以下问题：

1. 芯片复位后仍然会报gpio_num的错误，具体原因位置，暂时不影响使用

```
helloets Jun  8 2016 00:22:57

rst:0x1 (POWERON_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)
configsip: 188777542, SPIWP:0xee
clk_drv:0x00, q_drv:0x00, d_drv:0x00, cs0_drv:0x00, hd_drv:0x00, wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1184
load:0x40078000,len:12784
load:0x40080400,len:3032
entry 0x400805e4
[ 2][E][esp32-hal-gpio.c:95] _pinMode(): Invalid pin selected
E (240) gpio: gpio_set_level(226): GPIO output gpio_num error
[ 37][E][esp32-hal-gpio.c:95] _pinMode(): Invalid pin selected
E (33) gpio: gpio_set_level(226): GPIO output gpio_num error
SD Card Type: SDHC
SD Card Size: 3781MBReading file: /wifi.txt line: 1Reading file: /wifi.txt line: 2595 -2125 -289
```

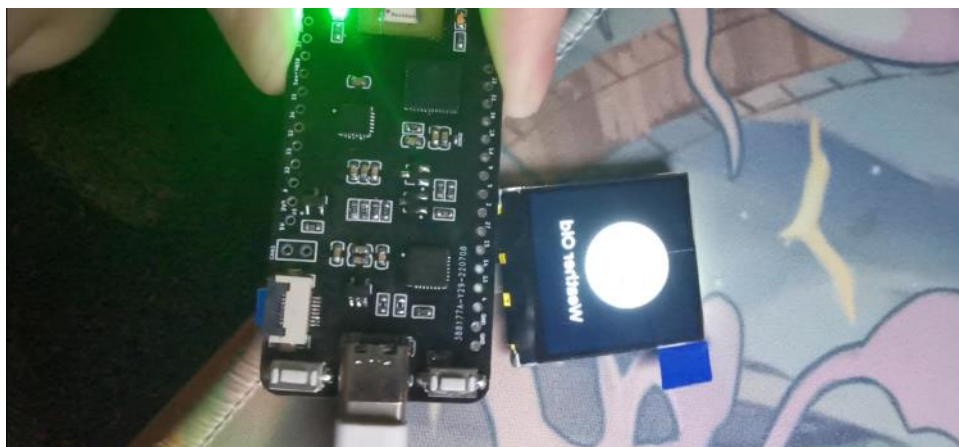
2. Cubic的IMU初始化中新增了

```
void IMU::init()
{
    Wire.begin(IMU_I2C_SDA, IMU_I2C_SCL);
    Wire.setClock(400000);
    // while (!imu.testConnection())
    // {
    //     Serial.println(imu.getDeviceID());
    // };
    imu.initialize();
}
```

发现在其中死循环，读取ID后发现返回31 32 0D 0A，目前采取的办法是直接注释该代码，可以正常读取IMU报文

3. 确保屏幕可以正常驱动的情况下，lvgl并没有正常运行，屏幕无任何显示

!!!!!!实在不知道Cubic原版lvgl的问题到底在哪，重新使用遛马小哥的AIO固件，除WIFI未连接测试外其余功能均正常运行。后续将在这个AIO_v2.0.6继续维护，外形和pcb计划使用metal版本，pcb厚度为1.2mm



2022/7/25

2022年7月26日 1:12

22~25进度内容:

利用AIO开源设计, 进行封装重置与电路重新布局。元件0402封装全部重置为0603, 电路进行重新布局, 4mil线路修改为10mil。主板布局删除光照传感器。

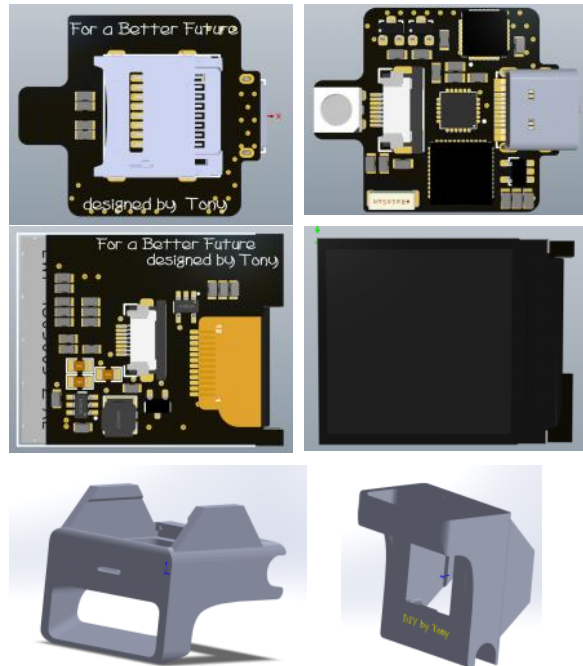
电路板单板嘉立创20元。

sw进行特征提取(现无法提取曲面特征, 需要安装新的插件), 进行logo打标。现在版本的名: 零件2.sldprt

3D打印嘉立创米白色光敏树脂:11块一个

.....目前作为第一版本, 拼装后在看时机效果进行修改

未来工场透明树脂打印23块, 铝合金打印78块
(但是没有表面处理, 最好能阳极氧化, 至少得喷漆), 都是可选的比较好的方案。



2022/7/28

2022年7月28日 19:42

两块电路板16:00到站

晚上焊接主板，除屏幕外全功能无问题，焊接情况良好。测试陀螺仪时发现固件中采用的是校准陀螺仪是否平方，才平方时获得姿态角结算参数。因此次陀螺仪发现需要将Type-C压到地上使其有几度的仰角，陀螺仪才认为平放，此问题可在程序中调整平方检测区间或手动添加偏移量解决。



2022/7/29

2022年7月30日 0:30

两块电路板焊接完成，DCDC模块测试成功。

仍存在以下问题

- 屏幕排线10cm过长，至少6cm以内。
- AIO程序发现wifi连接不太稳定，容易连不上。
- 天气API仅注册，暂未测试调用和收费情况。陀螺仪需在程序内进行角度修正。
- typec线接口过长，所有厂家都没有弯头尺寸数据，现已采购一款尝试
- 3d打印件安装合适，电路板固定用的胶还需选型，屏幕和分光棱镜所用的胶还未测试。



2022/7/30

2022年7月30日 20:36

今天在AIO的基础上修正了陀螺仪偏移补偿，增加了新的状态检测，还没写功能接口，有点小开心 ٩(๑>ٓ<๑)۶

自动校准补偿：在MPU6050.cpp中通过多数据采样，并根据其均值误差判断为修改原点补偿，采样次数为100*10次，每次误差小于500会修改当前角度为初始化：

```
Imu_data Imu_origin;
void MPU6050_Base::PID(uint8_t ReadAddress, float kP,float kI, uint8_t Loops){
    uint8_t SaveAddress = (ReadAddress == 0x3B)?((getDeviceID() < 0x38 )? 0x06:0x77):0x13;
    int16_t Data;
    float Reading;
    int16_t BitZero[3];
    uint8_t shift =(SaveAddress == 0x77)?3:2;
    float Error, PTerm, ITerm[3];
    int16_t sample_temp[100][3],origin_temp[3];
    int32_t setting_data[3];
    uint8_t setting_flag=1,origin_flag=0;
    int16_t eSample;
    uint32_t eSum;
    uint16_t gravity = 8192; // prevent uninitialized compiler warning
    if (ReadAddress == 0x3B) gravity = 16384 >> getFullScaleAccelRange();
    Serial.write('>');
    for (int i = 0; i < 3; i++) {
        I2Cdev::readWords(devAddr, SaveAddress + (i * shift), 1, (uint16_t *)&Data, I2Cdev::readTimeout, wireObj); // reads 1 or more 16 bit
integers (Word)
        Reading = Data;
        if(SaveAddress != 0x13){
            BitZero[i] = Data & 1; // Capture Bit Zero to properly handle Accelerometer calibration
            ITerm[i] = ((float)Reading) * 8;
        } else {
            ITerm[i] = Reading * 4;
        }
    }
    for (int L = 0; L < Loops; L++) {
        eSample = 0;
        memset(setting_data,0,sizeof(setting_data)); // 修正累加值清零
        origin_flag = 0; // 原点数据设置标志位清零
        for (int c = 0; c < 100; c++) { // 100 PI Calculations
            eSum = 0;
            for (int i = 0; i < 3; i++) {
                I2Cdev::readWords(devAddr, ReadAddress + (i * 2), 1, (uint16_t *)&Data, I2Cdev::readTimeout, wireObj); // reads 1 or more 16 bit
integers (Word)
                Reading = Data;
                if ((ReadAddress == 0x3B)&&(i == 2)) Reading -= gravity; //remove Gravity
                // Serial.print(Reading);
                // Serial.print("\t");
                // Serial.print(" || ");
                //修正mpu偏差后采样
                if(ReadAddress == 0x3B){
                    if(i == 0) Reading -= Imu_origin.ax;
                    else if(i == 1) Reading -= Imu_origin.ay;
                    else if(i == 2) Reading -= Imu_origin.az;
                }
                else if(ReadAddress == 0x43){
                    if(i == 0) Reading -= Imu_origin.gx;
                    else if(i == 1) Reading -= Imu_origin.gy;
                    else if(i == 2) Reading -= Imu_origin.gz;
                }
            }
            Error = -Reading;
            sample_temp[c][i] = Reading;
            // Serial.print(Reading);
            // Serial.print("\t");
            // if(i==2) Serial.println("");

            //原点设置修正部分
            if(c > 1){
                if(setting_flag == 1){
                    if(abs(sample_temp[c][i] - sample_temp[c-1][i]) > 500){ //出现跳变，偏差设置标志位置0
                        setting_flag = 0;
                    }
                } else{
                    setting_data[i] += sample_temp[c][i];
                }
            }
        }
    }
}
```

```

    }
}

eSum += abs(Reading);
PTerm = kP * Error;
ITerm[i] += (Error * 0.001) * kI;          // Integral term 1000 Calculations a second = 0.001
if(SaveAddress != 0x13) {
    Data = round((PTerm + ITerm[i]) / 8);    //Compute PID Output
    Data = ((Data)&0xFFFE) | BitZero[i];    // Insert Bit0 Saved at beginning
} else Data = round((PTerm + ITerm[i]) / 4); //Compute PID Output
I2Cdev::writeWords(devAddr, SaveAddress + (i * shift), 1, (uint16_t *)&Data, wireObj);
}
if((c == 99) && eSum > 1000) {              // Error is still to great to continue
    c = 0;
    Serial.write('*');
    if(setting_flag == 1) { // 偏差设置标志位为1, 进行偏差计算
        if(ReadAddress == 0x3B) { //加速度计原点修正
            for(int i = 0; i < 3; i++) {
                if(abs(origin_temp[i] - setting_data[i]/100)<400) { //两次100组数据比较, 均值差位400以内有效
                    origin_flag++;
                }
                else {
                    origin_flag = 0; //均值差异过大, 所有数据清零
                    break;
                }
            }
            origin_temp[0] = setting_data[0]/100;
            origin_temp[1] = setting_data[1]/100;
            origin_temp[2] = setting_data[2]/100;
            Serial.print(origin_flag);
            if(origin_flag >= 30) {
                Imu_origin.ax = setting_data[0]/100;
                Imu_origin.ay = setting_data[1]/100;
                Imu_origin.az = setting_data[2]/100;
                origin_flag = 0;
                Serial.print("Imu_origin Accel Set: ");
                Serial.print(Imu_origin.ax);
                Serial.print("\t");
                Serial.print(Imu_origin.ay);
                Serial.print("\t");
                Serial.println(Imu_origin.az);
            }
        }
        else if(ReadAddress == 0x43) { //陀螺仪原点修正
            Imu_origin.gx = setting_data[0]/100;
            Imu_origin.gy = setting_data[1]/100;
            Imu_origin.gz = setting_data[2]/100;
        }
    }
    else {
        origin_flag = 0; //偏差设置标志位为0, 表示出现跳变, 所有数据清零
    }
    memset(setting_data, 0, sizeof(setting_data)); // 修正累加值清零
    setting_flag = 1;
}
if((eSum * ((ReadAddress == 0x3B)?0.05: 1)) < 5) eSample++; // Successfully found offsets prepare to advance
if((eSum < 100) && (c > 10) && (eSample >= 10)) break; // Advance to next Loop
delay(1);
}
Serial.write('.');
kP *= .75;
kI *= .75;
for (int i = 0; i < 3; i++) {
    if(SaveAddress != 0x13) {
        Data = round((ITerm[i]) / 8);    //Compute PID Output
        Data = ((Data)&0xFFFE) | BitZero[i]; // Insert Bit0 Saved at beginning
    } else Data = round((ITerm[i]) / 4);
    I2Cdev::writeWords(devAddr, SaveAddress + (i * shift), 1, (uint16_t *)&Data, wireObj);
}
}
resetFIFO();
resetDMP();
}

```


并在imu.cpp中增加对偏差计算后的新角度：

//函数：使用原点修正陀螺仪采样

```
void IMU::fixOriginDeflection6(ImuAction *action_info, Imu_data *origin_data)
{
    action_info->v_ax -= origin_data->ax;
    action_info->v_ay -= origin_data->ay;
    action_info->v_az -= origin_data->az;
    action_info->v_gx -= origin_data->gx;
    action_info->v_gy -= origin_data->gy;
    action_info->v_gz -= origin_data->gz;
}
```

并在imu.cpp与imu.h中新增陀螺仪状态，并添加超长按状态检测，此处新增为HIBERNATE状态：

enum ACTIVE_TYPE

```
{
    TURN_RIGHT = 0,
    RETURN,
    TURN_LEFT,
    UP,
    DOWN,
    GO_FORWARD,
    HIBERNATE,
    SHAKE,
    UNKNOWN
};
```

```
const char *active_type_info[] = {"TURN_RIGHT", "RETURN",
                                   "TURN_LEFT", "UP",
                                   "DOWN", "GO_FORWARD",
                                   "HIBERNATE", "SHAKE", "UNKNOWN"};
```

//识别为超长按，多用于检测人已经不在情况

```
for(int i=0;i<40;i++){
    if(act_info_history[i] == act_info_history[index]
        && act_info_history[index] != ACTIVE_TYPE::SHAKE
        && act_info_history[index] != ACTIVE_TYPE::UNKNOWN)
        isHibernate = true;
    else {
        isHibernate = false;
        break;
    }
}
if(isHibernate){
    action_info.isValid = 1;
    action_info.active = ACTIVE_TYPE::HIBERNATE;
}
```

2022/7/31

2022年7月31日 23:58

发现AIO 2.0.8重大bug，wifi连接有ap和STA两种模式，共用同一个标志位，运行Webserver只会开启AP节点，并给wifi标志位置1，使STA正常的Wifi不进入连接状态。

```
boolean m_wifi_status;           // 表示是wifi状态 true开启 false关闭
boolean sta_wifi_status;         // 发现bug, ap与sta标志位重合, 新增sta标志位, 表示是wifi状态 true开启 false关闭

case APP_MESSAGE_WIFI_CONN:
{
    // 更新请求
    // CONN_ERROR == g_network.end_conn_wifi() {}
    Serial.println("mission is start try to connect WiFi");
    Serial.println(sta_wifi_status);
    if (false == sta_wifi_status)
    {
        Serial.println("start try to connect WiFi");
        g_network.start_conn_wifi(sys_cfg.ssid_0.c_str(), sys_cfg.password_0.c_str());
        sta_wifi_status = true;
    }
    m_preWifiReqMillis = millis();
    if ((WiFi.getMode() & WIFI_MODE_STA) == WIFI_MODE_STA && CONN_SUCC != g_network.end_conn_wifi())
    {
        // 在STA模式下 并且还没连接上wifi
        return false;
    }
}
```

修改Webserver中的Wifi连接逻辑，现在STA本地Wifi和AP都会开启

```
if (0 == run_data->web_start && 0 == run_data->req_sent)
{
    // 预显示
    display_setting(
        "WebServer Start",
        "Domain: holocubic",
        "Wait...", "Wait...",
        // "", "",
        LV_SCR_LOAD_ANIM_NONE);
    // 如果web服务没有开启 且 ap开启的请求没有发送 message这边没有作用 (填0)
    sys->send_to(SERVER_APP_NAME, CTRL_NAME,
        APP_MESSAGE_WIFI_AP, NULL, NULL);
    //同时开启STA_Wifi连接
    sys->send_to(SERVER_APP_NAME, CTRL_NAME,
        APP_MESSAGE_WIFI_CONN, NULL, NULL);
    run_data->req_sent = 1; // 标志为 ap开启请求已发送
}
```

使用新版天气API，采用V6版本API，目前没有发现官网上明确写了这个版本，暂时没看到收费的情况，可能这就是隐藏福利吧 ~(^_^)~

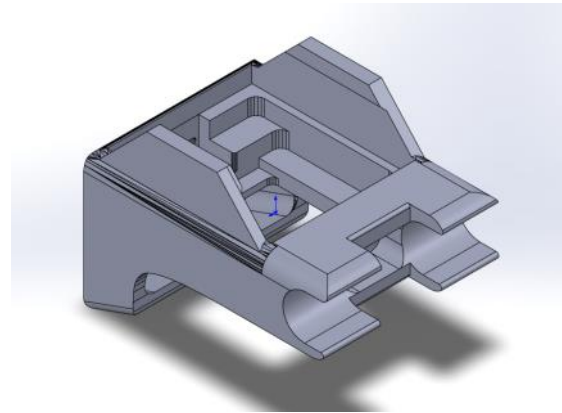


2022/8/1

2022年8月2日 16:18

完成第二版3d文件设计，现命名为 my_cubic3D_v2.0，根据购买的typec弯头测量尺寸绘制，将后部接头处完全包裹。并分别打印了透明，黑色和铝合金版本。

完成了第二块电路板的焊接，这里强烈谴责自己居然把32的方向焊反了，幸好没烧。但是测试屏幕和电路板粘贴的时候粘废了一块屏幕。粘性很强，一定要仔细粘。



2022/8/3

2022年8月3日 20:33

成功通过lvgl在开机的时候显示自定义文字，现将内容总结如下：

screen.init() 函数中包含 lv_init(); 等于lvgl的相关初始化函数，在该函数后调用lvgl库即可

lvgl调用时根据显示的内容为文字，图片，组件等都不相同，此处仅测试使用文字。

```
lv_obj_t *my_main_scr = NULL;
lv_obj_t *my_title_label;
static lv_style_t default_style;
static lv_style_t label_style;
LV_FONT_DECLARE(lv_font_montserrat_24);

void my_gui_init(void)
{
    my_main_scr = NULL;
    my_title_label = NULL;
    lv_style_init(&default_style);
    lv_style_set_bg_color(&default_style, LV_STATE_DEFAULT,
LV_COLOR_BLACK);
    lv_style_set_bg_color(&default_style, LV_STATE_PRESSED,
LV_COLOR_GRAY);
    lv_style_set_bg_color(&default_style, LV_STATE_FOCUSED,
LV_COLOR_BLACK);
    lv_style_set_bg_color(&default_style, LV_STATE_FOCUSED |
LV_STATE_PRESSED, lv_color_hex(0xf88));
    // 本地的ip地址
    my_main_scr = lv_obj_create(NULL, NULL);
    lv_obj_add_style(my_main_scr, LV_BTN_PART_MAIN,
&default_style);
    lv_style_init(&label_style);
    lv_style_set_text_opa(&label_style, LV_STATE_DEFAULT,
LV_OPA_COVER);
    lv_style_set_text_color(&label_style, LV_STATE_DEFAULT,
LV_COLOR_WHITE);
```

```

        lv_style_set_text_font(&label_style, LV_STATE_DEFAULT,
&lv_font_montserrat_24);
    }

```

```

void my_display_setting_init(void)
{
    lv_obj_t *act_obj = lv_scr_act(); // 获取当前活动页
    if (act_obj == my_main_scr)
        return;
    lv_obj_clean(act_obj); // 清空此前页面
    my_title_label = lv_label_create(my_main_scr, NULL);
}

```

```

void display_setting(void)
{
    my_display_setting_init();
    lv_obj_add_style(my_title_label, LV_LABEL_PART_MAIN,
&label_style);
    lv_label_set_text(my_title_label, "Tony");
    lv_obj_align(my_title_label, NULL,
LV_ALIGN_OUT_BOTTOM_LEFT, 5, -200);
    // lv_scr_load_anim(my_main_scr, anim_type, 300, 300,
false);
    lv_scr_load(my_main_scr);
    screen.routine(); //若之前有已经显示了相关内容需要刷新
一遍
}

```

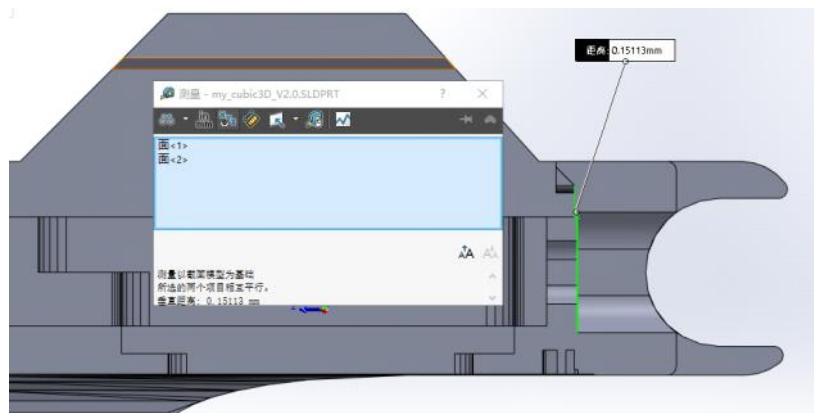
在start中依次调用 `my_gui_init` 与 `display_setting` 可以完成 "Tony"的
显示

修改相关style设置可以更改显示的内容格式。

2022/8/6

2022年8月7日 1:53

打印金属件已到，发现这个位置设计有个缺缺。



同时，金属打印公差较大，TYPEC口，屏幕，棱镜接触面均需要打磨，且一定要小心弄坏棱镜，已经弄坏了一个缺缺了。

同时发现，最新修改的程序，tmd图片和视频打不开了，现在好不清楚具体是哪出了问题，9成是程序问题，需要回溯进行排查。发现好像程序没问题，变成卡和芯片有问题了。。。。

问题确定，第二张sd读不到，两个板子都能读到第一个sd卡，但有问题的sd卡使能通过电脑读取到的，可能需要重新购买。