

BÁO CÁO ĐỒ ÁN CUỐI KÌ MÔN IOT

I. Thông tin nhóm

- + 1753062 – Lê Minh Khang
- + 1753002 – Huỳnh Minh Chí
- + 1753032 – Nguyễn Tất Bình

II. Thông tin đề tài

- Tên sản phẩm: **Smart DoorBell**

- Mô tả tổng quan đề tài:

+ Smart Doorbell là loại chuông cửa thông minh để thông báo cho điện thoại hoặc thiết bị điện tử khác của chủ nhà khi có khách bấm chuông cửa.

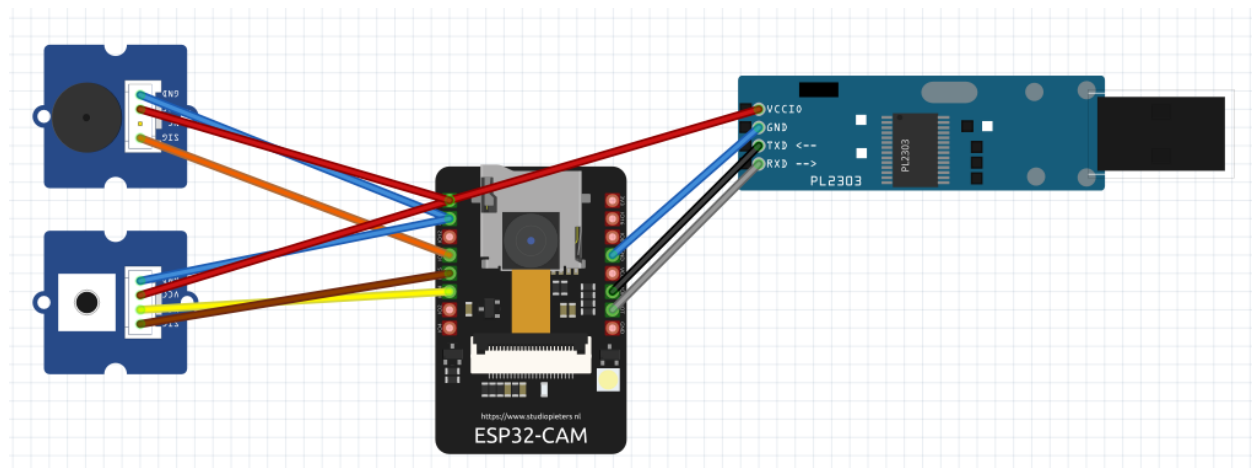
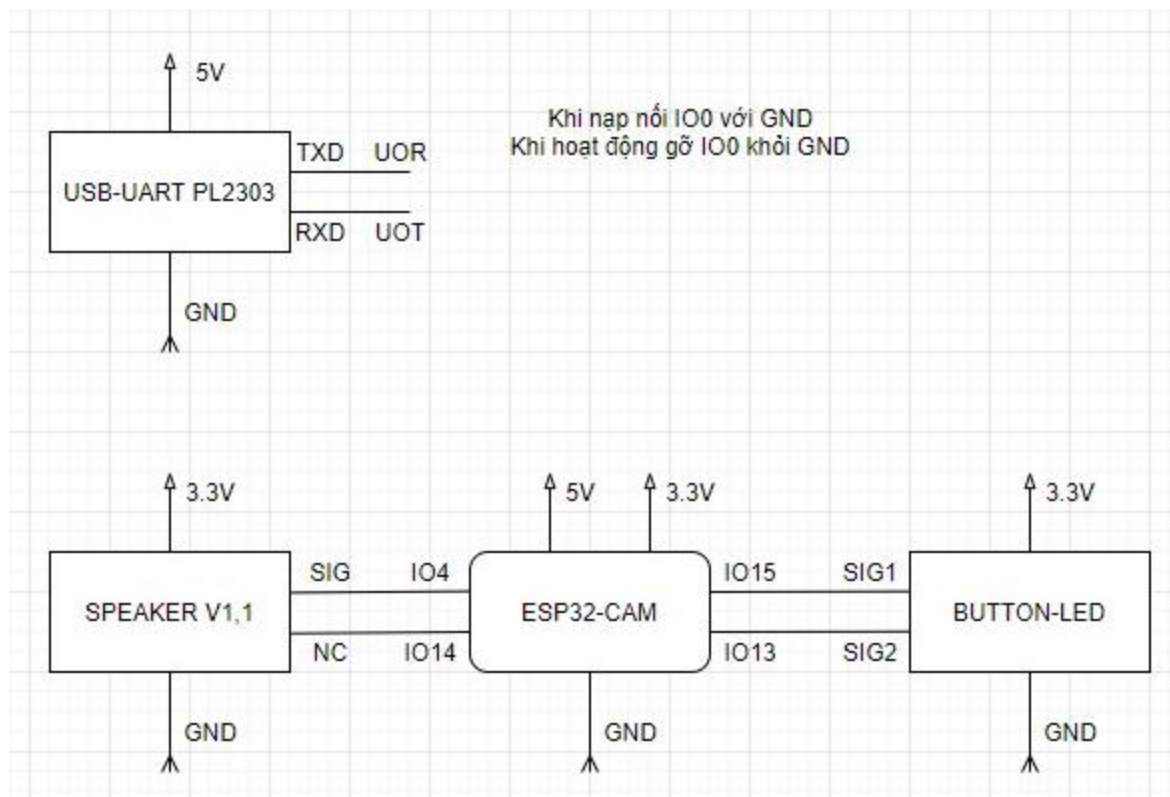
+ Smart Doorbell tích hợp với các ứng dụng được thiết kế cho các thiết bị Android và Apple như điện thoại thông minh.

+ Khi có khách kích hoạt Smart Doorbell của bạn, nó sẽ gửi cảnh báo đến ứng dụng của bạn, cho phép bạn xem trực tiếp của hành động xảy ra ngay trước cửa nhà bạn.

III. Thiết kế sản phẩm



IV. Mạch điện



V. Danh sách thiết bị thực tế

Tên thiết bị	Số lượng
ESP32-Cam	1
Grove - Blue LED Button	1
Grove - Speaker	1
Mạch Chuyển USB UART PL2303	1

Bảng đấu chân mạch:



Grove - Button	ESP32-Cam
GND	GND
VCC	5v
SIG1	15
SIG2	14
Grove - Speaker	ESP32-Cam
GND	GND
VCC	5v
SIG1	13
Mạch Chuyển USB UART PL2303	ESP32-Cam
GND	GND
VCC	5v
TXD	U0R
RXD	U0T

Hướng dẫn sử dụng:

- Thông báo khi có người nhấn chuông:

+ Khi người có người nhấn vào button, chuông sẽ phát ra âm thanh, và gửi 1 mail tới địa chỉ mail của chủ nhà để thông báo, trong mail có 1 đường link server để chủ nhà thao tác xem camera và đóng/ mở cửa.

+ Khi người dùng sử dụng lần đầu tiên cần setup wifi, và địa chỉ của server

  192.168.4.1

Input value Conenct

SSID

ThanhVy

Mật khẩu

230220081

IP

192.168.1.88

PORT

8888

Connect

Dành cho người mới cần setup lại thiết bị:

+ Cấu hình wifi của esp32:

```
const char *ssid = "ESP32-SETUP";  
const char *password = "12345678";
```

+ Cấu hình mail gửi mà nhận thông báo:

```
#define emailSenderAccount    "minhkhait.dev@gmail.com"  
#define emailSenderPassword  "0123456789khang"  
#define emailRecipient       "minhkhait123@gmail.com"  
#define smtpServer            "smtp.gmail.com"  
#define smtpServerPort        465  
#define emailSubject          "Thông báo ring bell"
```

+ Cấu hình server gửi dữ liệu của camera:

```
<script>  
  const img = document.querySelector('img');  
  const WS_URL = 'ws:///192.168.1.88:8888';  
  const ws = new WebSocket(WS_URL);  
  let urlObject;  
  ws.onopen = () => console.log(`Connected to ${WS_URL}`);  
  ws.onmessage = message => {  
    const arrayBuffer = message.data;  
    if (urlObject) {  
      URL.revokeObjectURL(urlObject);  
    }  
    urlObject = URL.createObjectURL(new Blob([arrayBuffer]));  
    img.src = urlObject;  
  }  
</script>
```

➔ Sửa lại địa chỉ thành địa chỉ ip của server