

How to Make

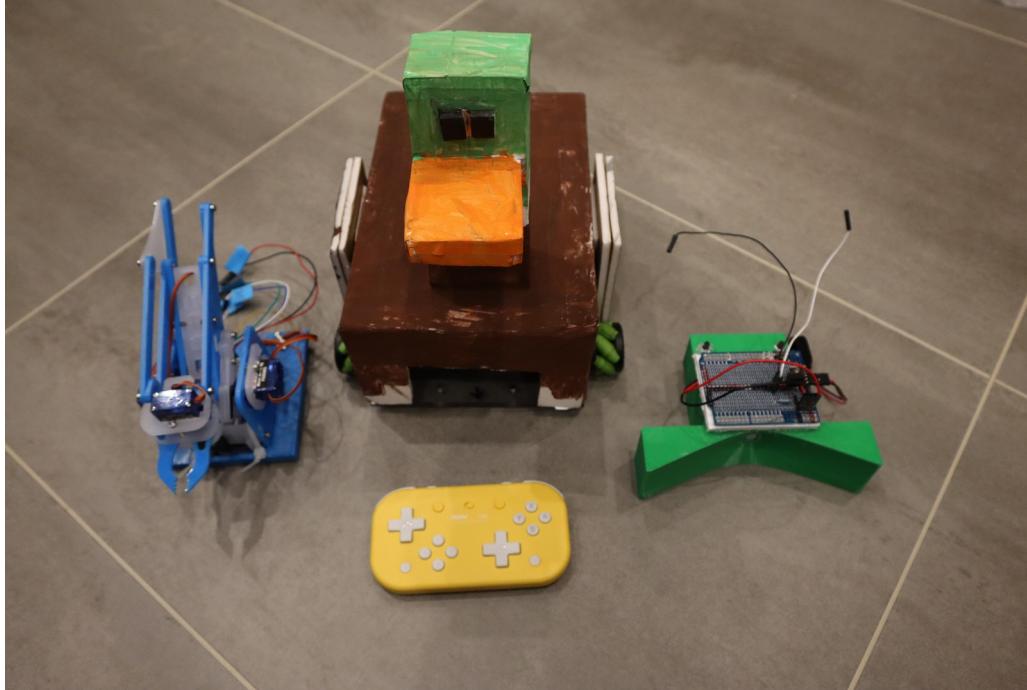
Katie Dumont

These slides can be found at:
https://github.com/hifromkatie/HowToMake_Slides

Who am I?

- BEng Computers, Electronics and Communications at the University of Bath
- STEM Ambassador
 - Code Club at Stithians school for the last 6 years
 - AstroPi Mission Zero and Space Lab (Top 10 Winners 2019, Through to report round 2020)
 - Pi Wars 2021, 8th in the Young Teams – Novice Category

PiWars 2021



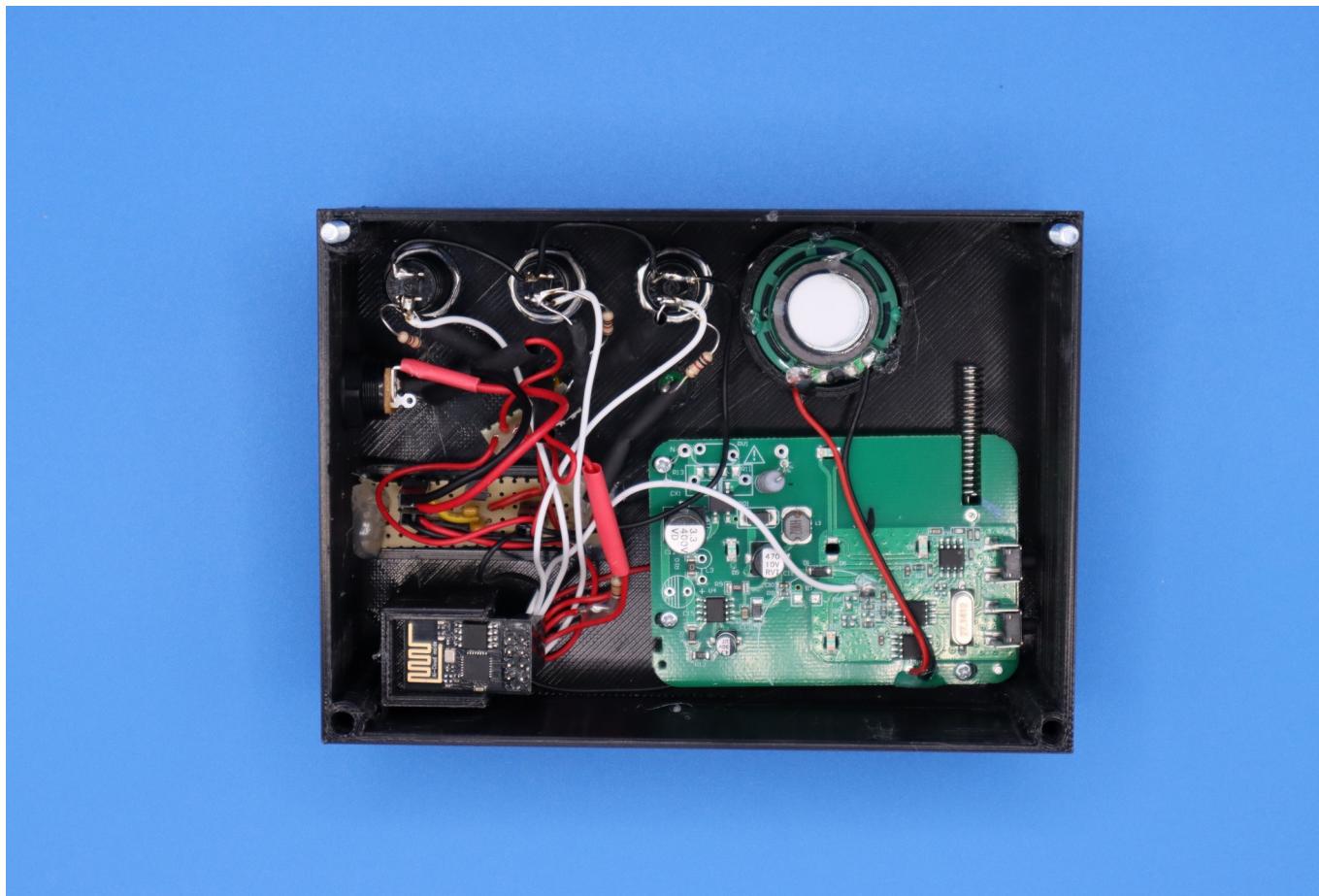
What I do?

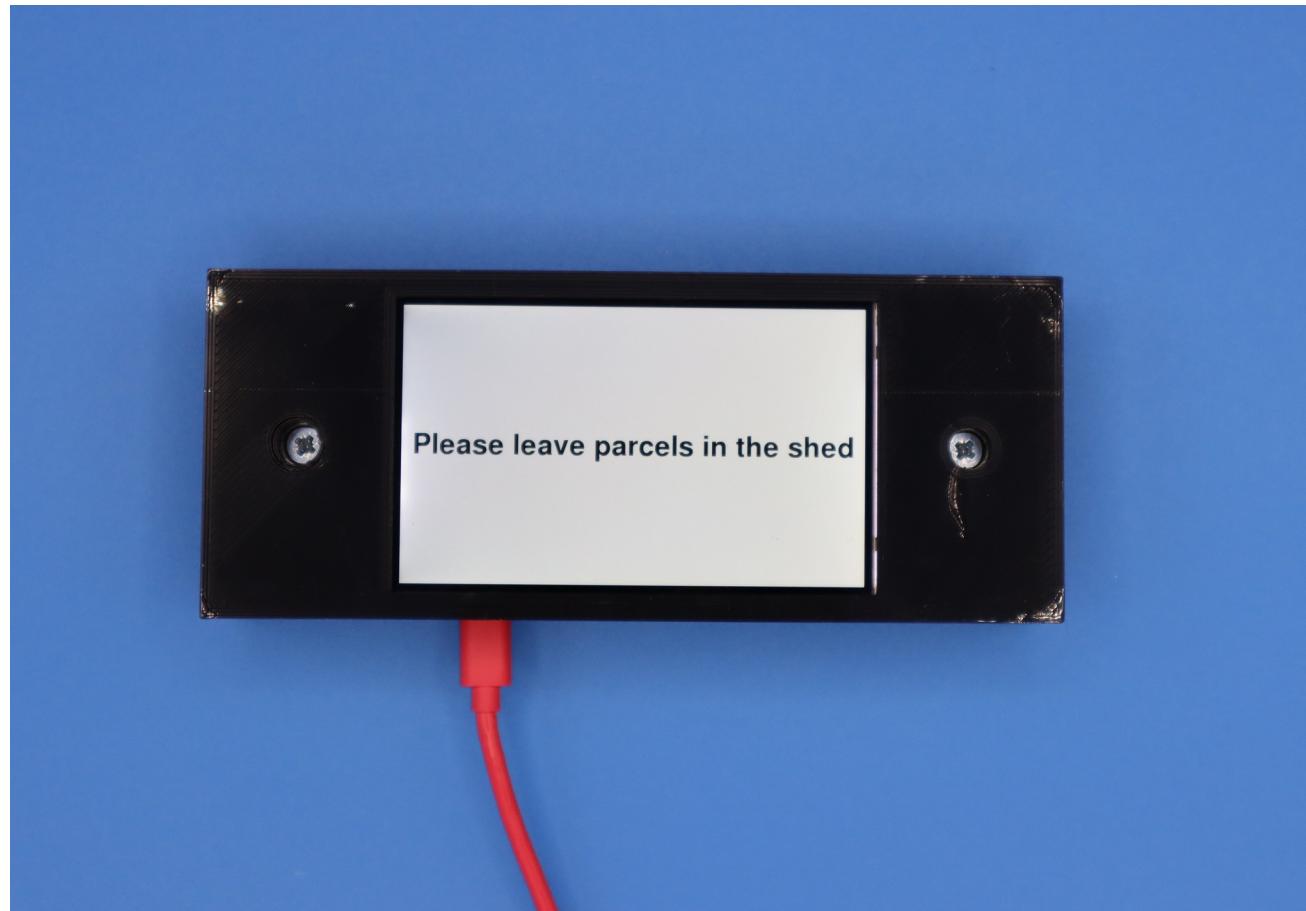
- VCP – Video Content Producer for Element14
 - Plan and present idea for project
 - Parts list
 - Create project
 - Video creating project
 - Create a “tutorial” video
 - Help & support others recreating project

Ideas

- Look at things around you
 - Needs improving



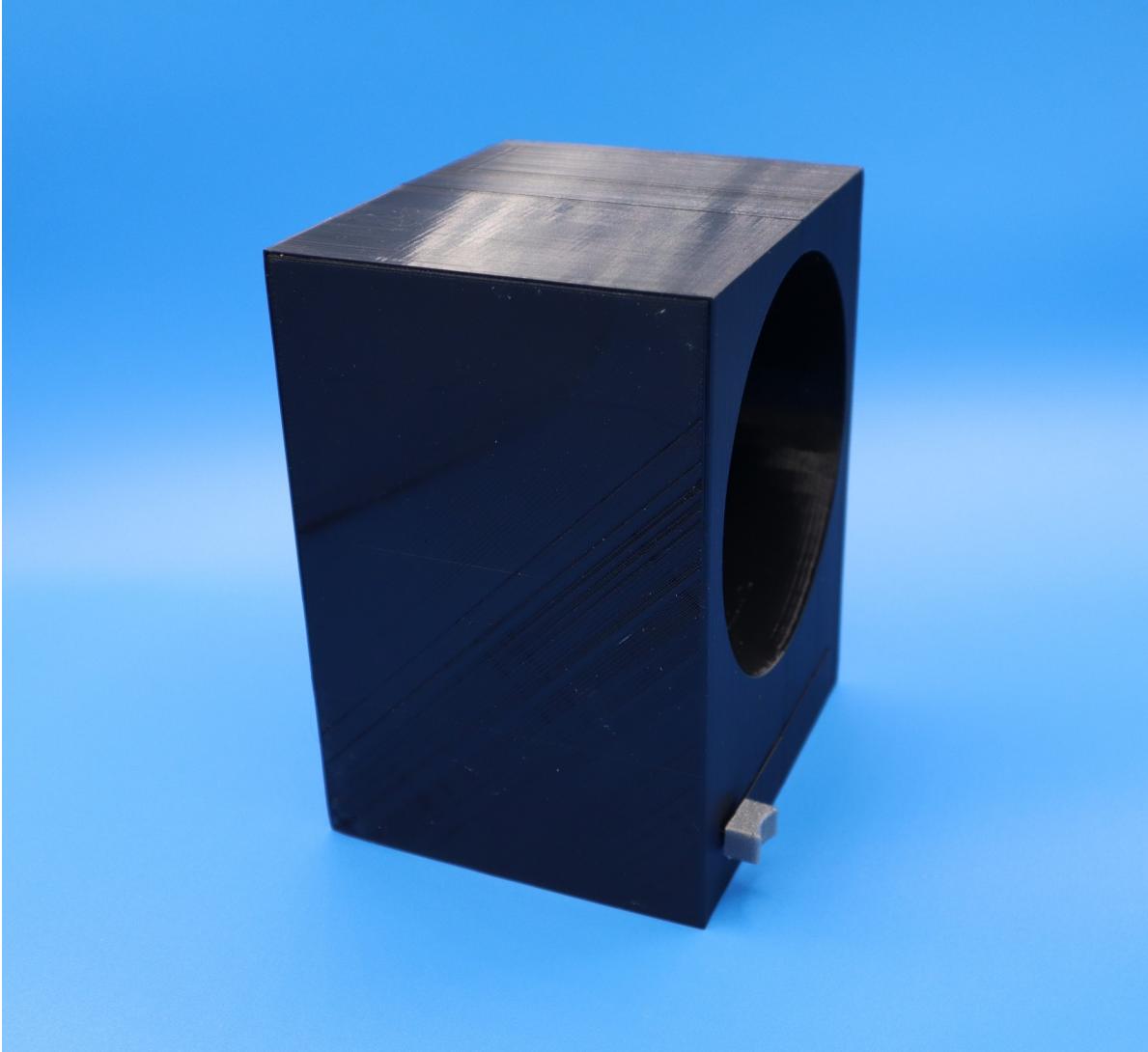




Ideas

- Look at things around you
 - Needs improving
 - Requires a solution





Ideas

- Look at things around you
 - Needs improving
 - Requires a solution
- Ask people if they need something



Ideas

- Look at things around you
 - Needs improving
 - Requires a solution
- Ask people if they need something
- Just for fun!



Break project down

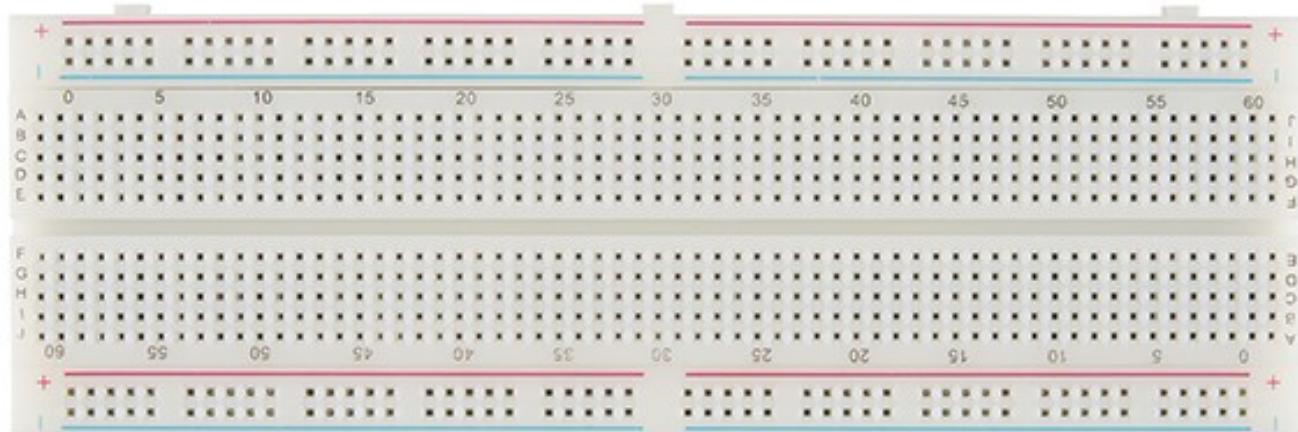
- Break project down into stages/steps/targets
 - Physical stages
 - Electronics stages
 - Hardware/Software Goals
 - Goals/targets

Parts

- Working out what parts you need
 - Development/evaluation boards, maker boards
 - Individual components
 - If SMD components, PCB design
 - All electronics or a Hardware/software mix?
 - Microcontroller
 - Raspberry Pi

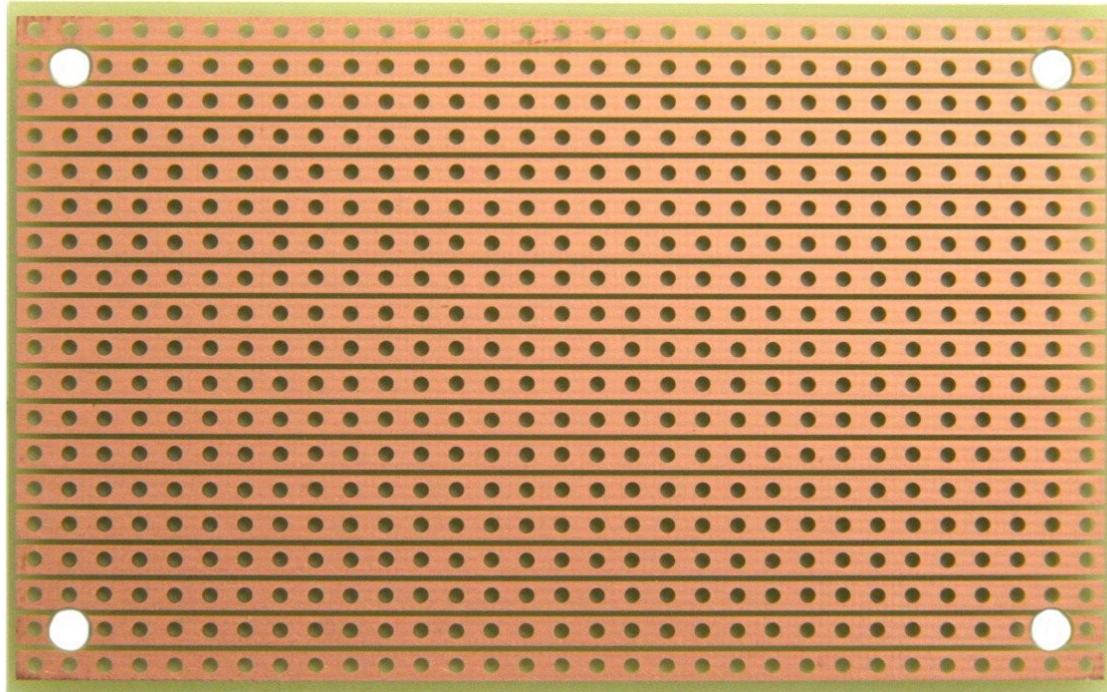
Circuit design

- Breadboard



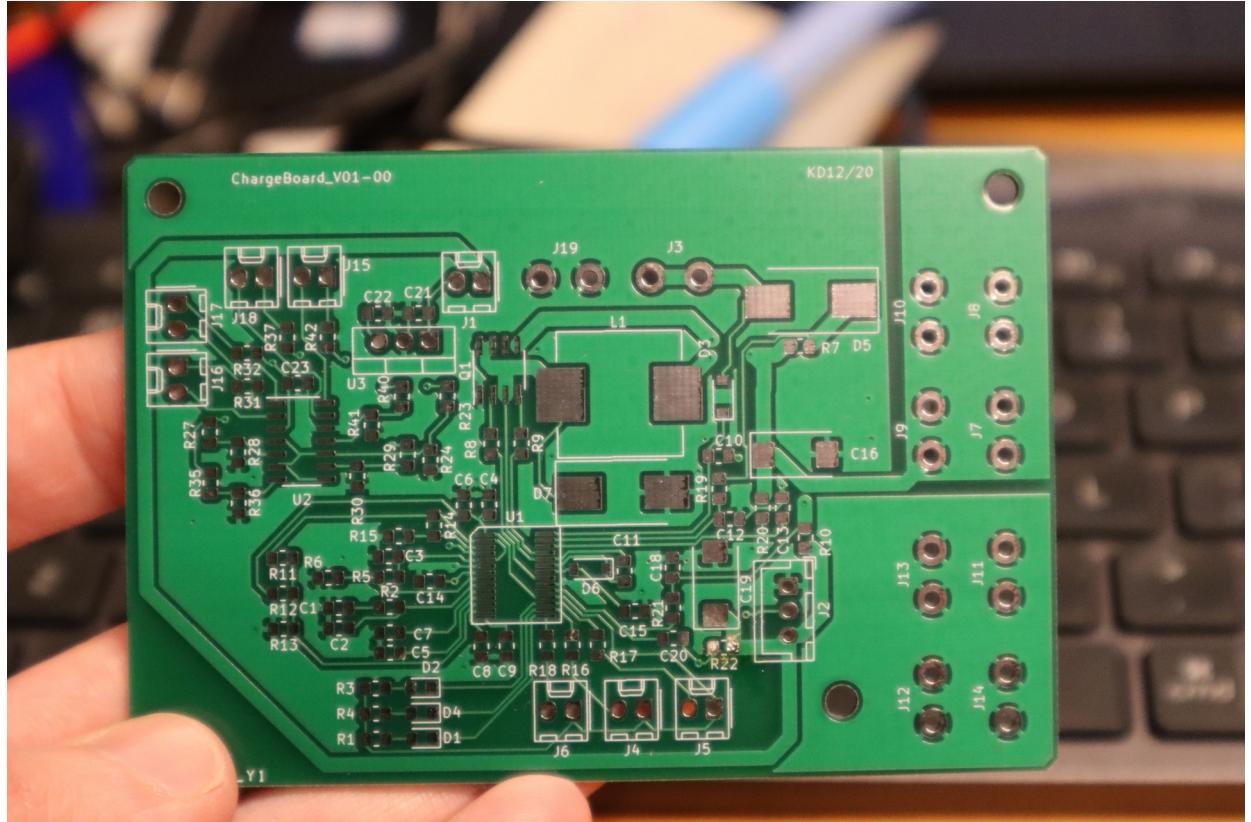
Circuit design

- Breadboard
- Strip Board



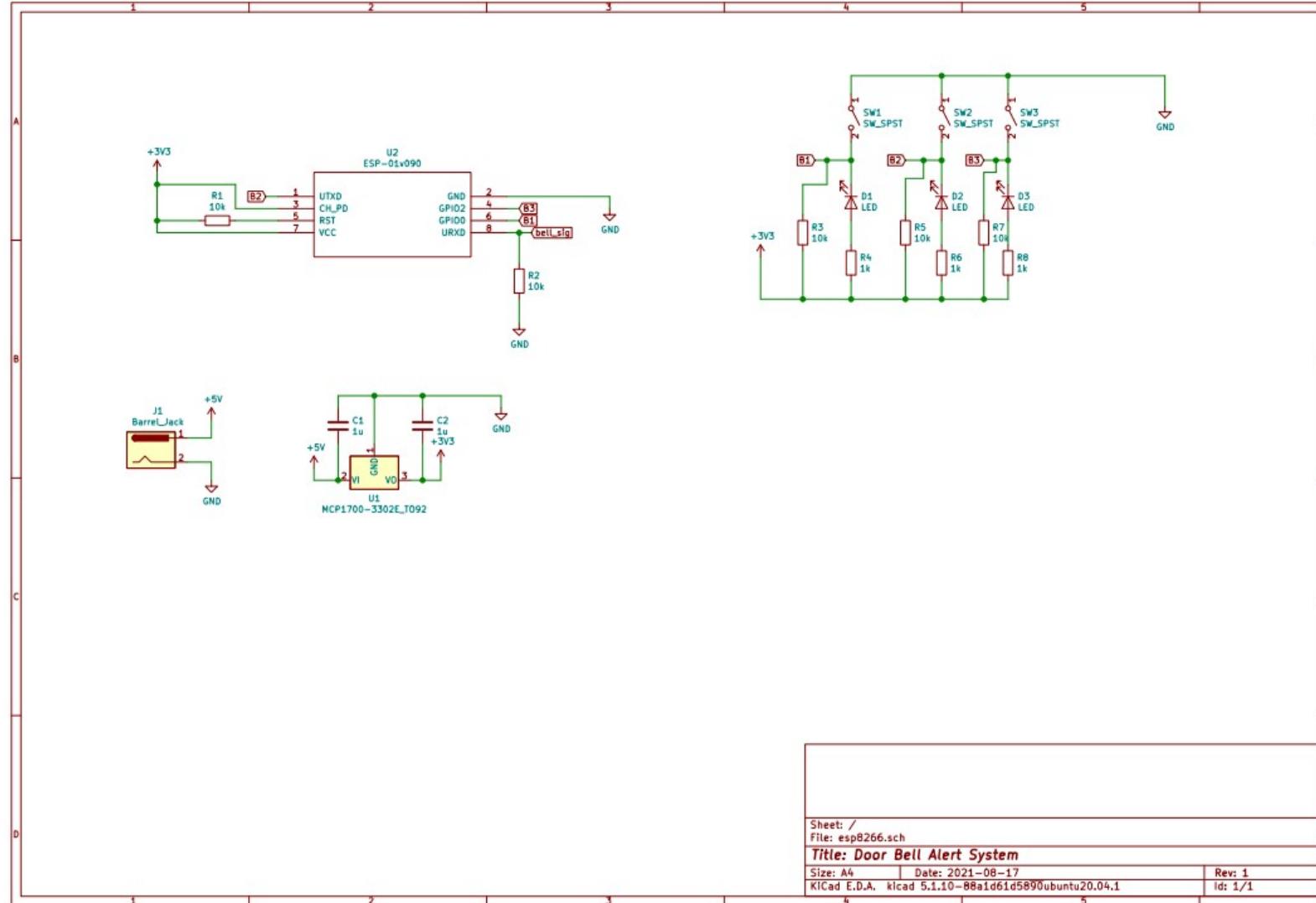
Circuit design

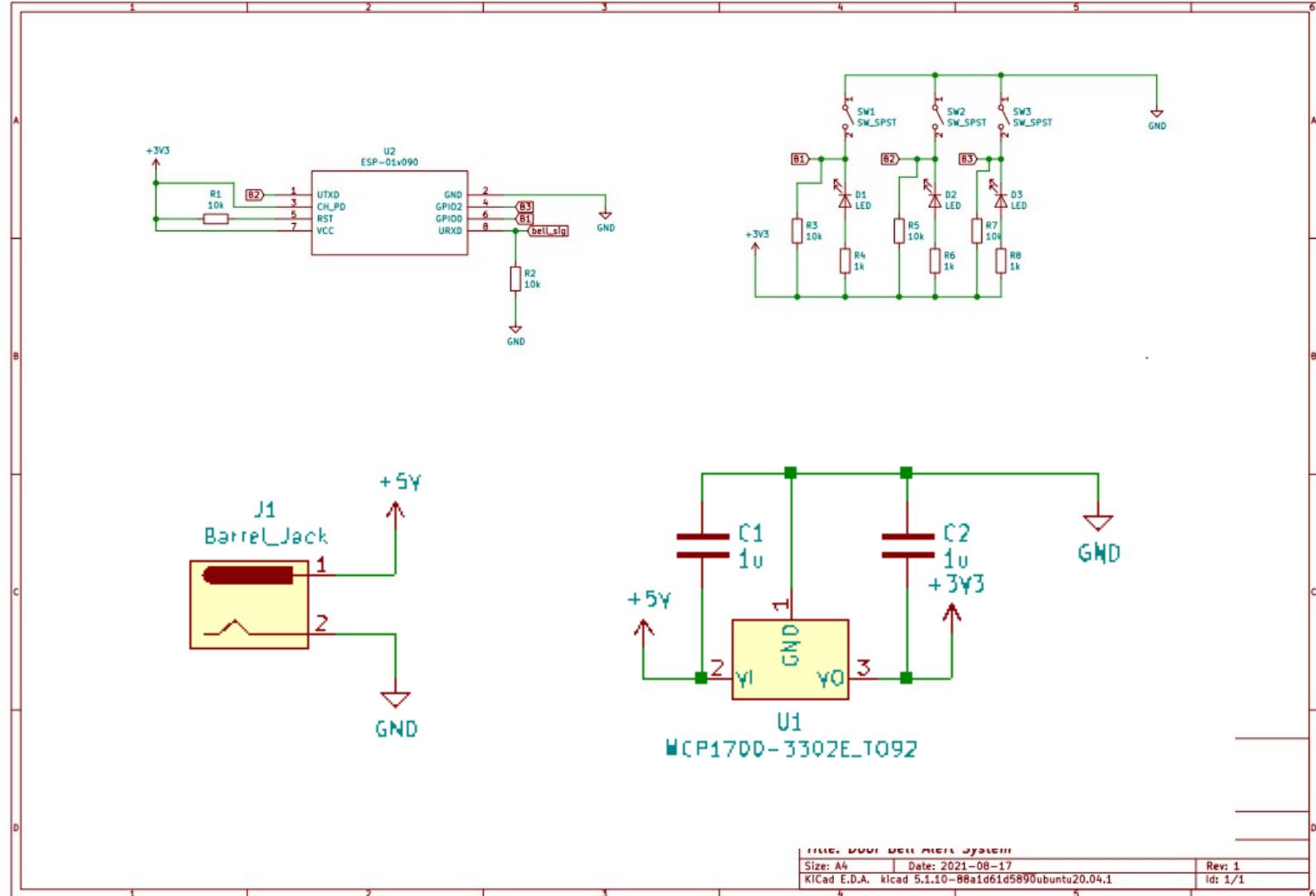
- Breadboard
- Strip Board
- PCB



An example

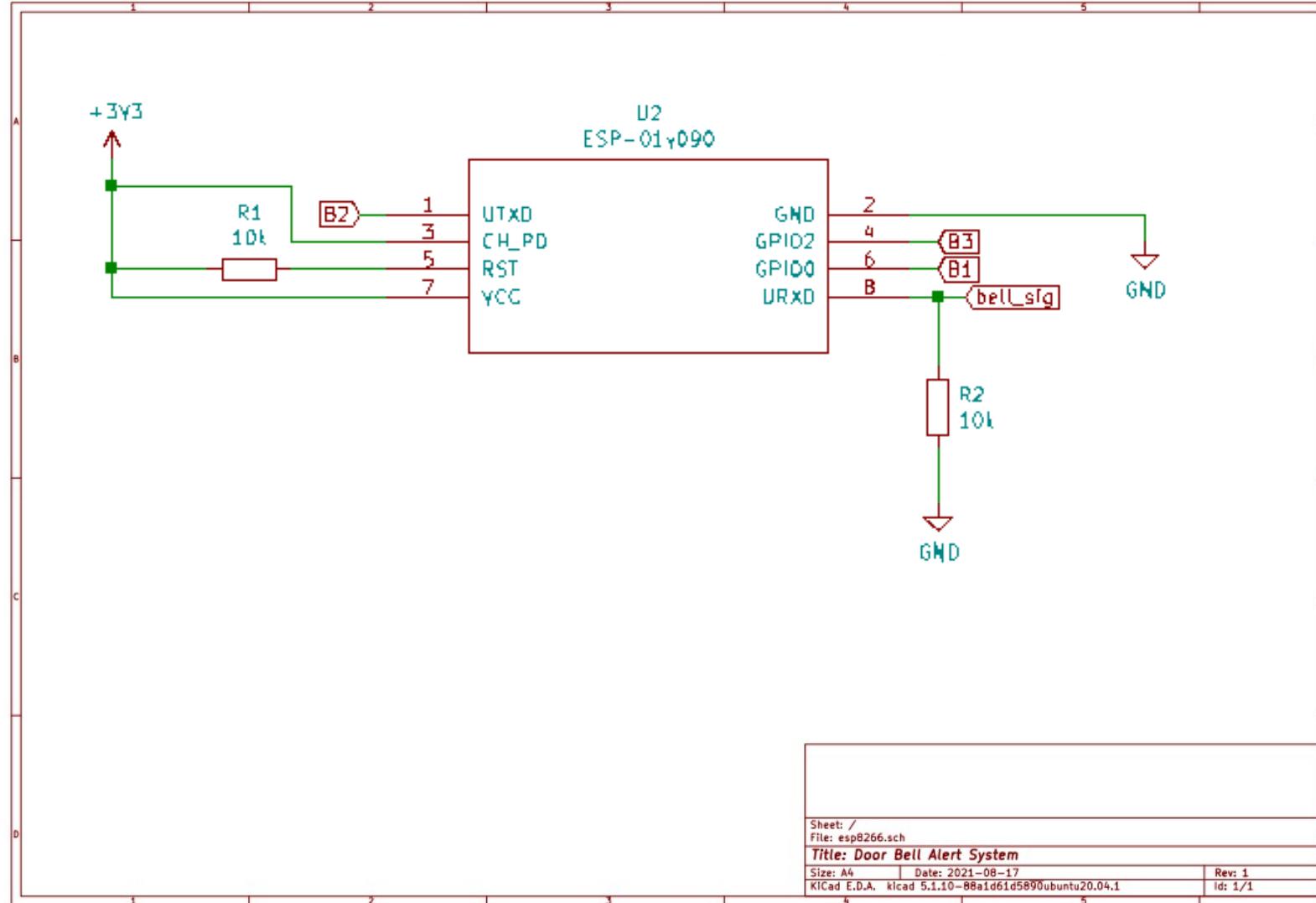
IOT Door Bell and door information screen





DOOR BELL ALERT SYSTEM

Size: A4	Date: 2021-08-17	Rev: 1
KICad E.D.A.	Kicad 5.1.10-88a1d61d5890ubuntu20.04.1	Id: 1/1

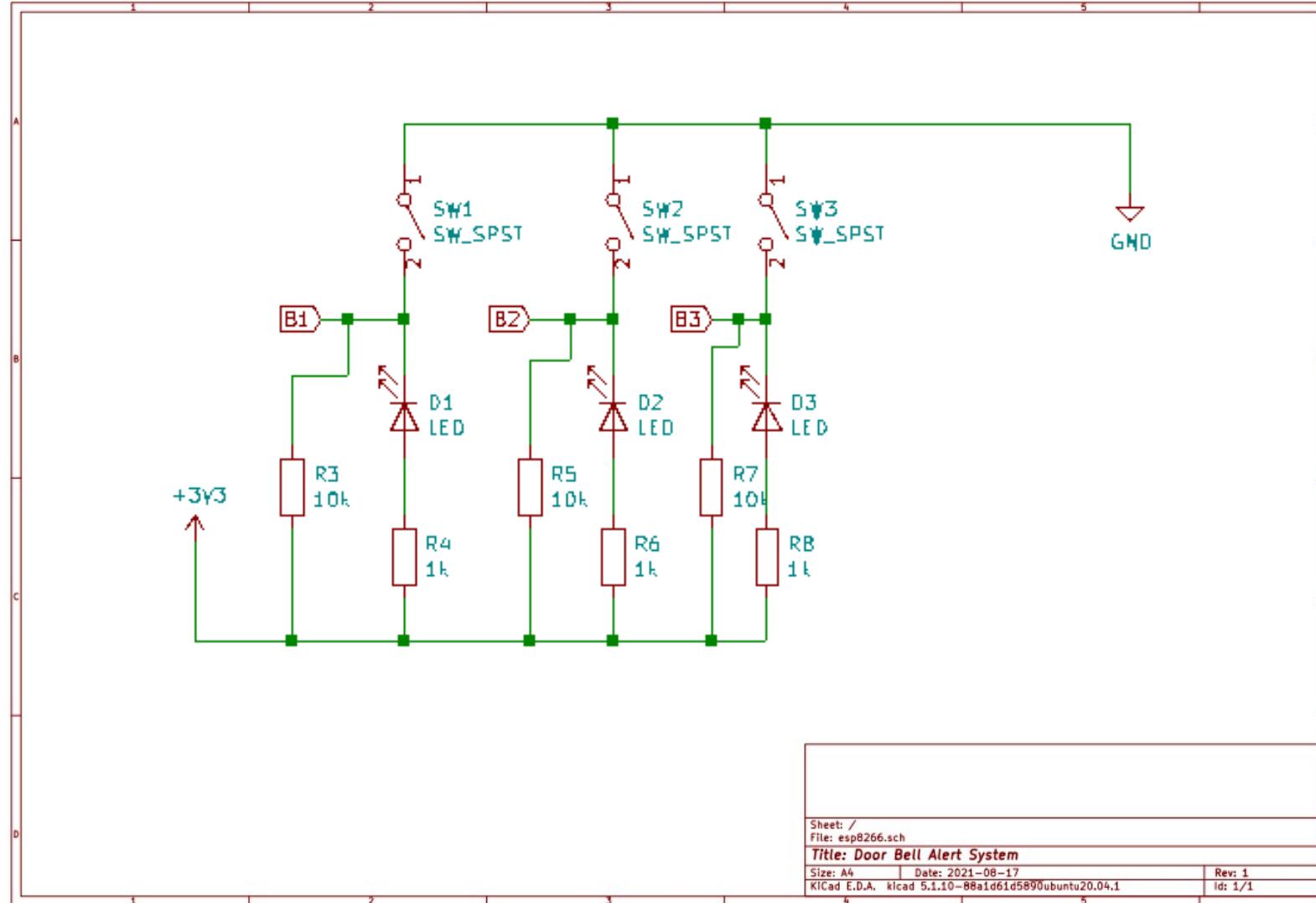


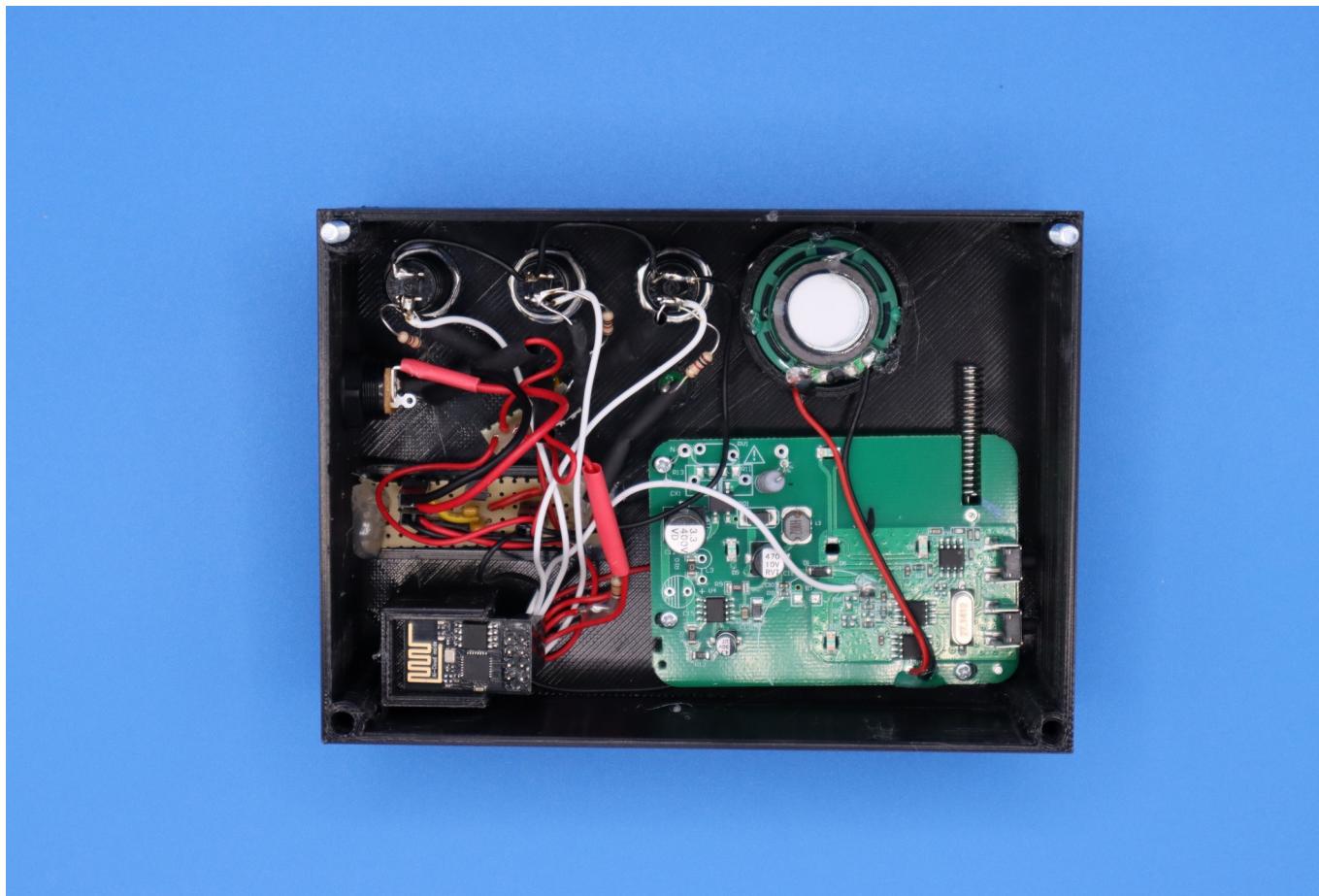
Sheet: /
File: esp8266.sch

Title: Door Bell Alert System

Size: A4 | Date: 2021-08-17
KICad E.D.A. kicad 5.1.10-88aid61d5890ubuntu20.04.1

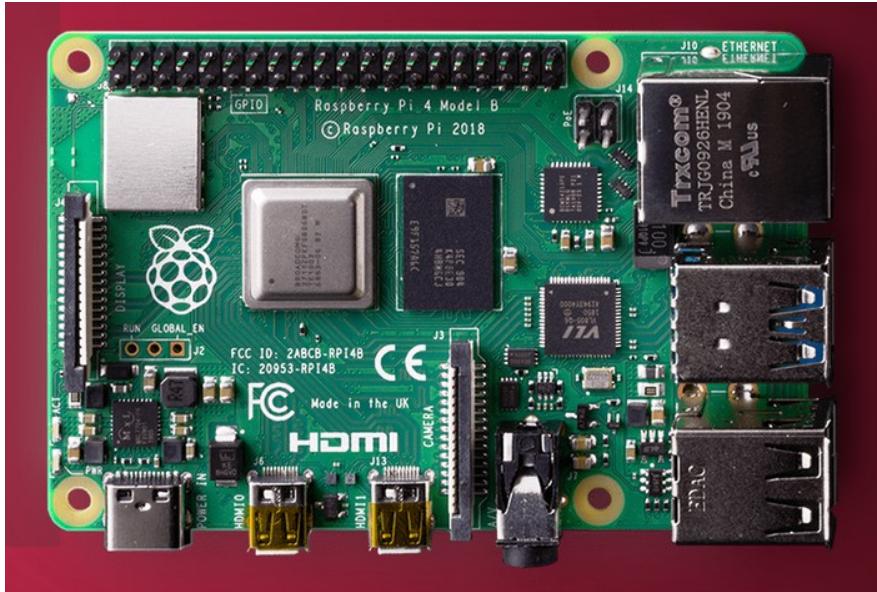
Rev: 1
Id: 1/1





Door side

- Raspberry Pi



Door side

- Raspberry Pi
- HyperPixel Screen





Please leave parcels in the shed

Where software meets hardware

- Microcontroller:
 - C /C++
 - MicroPython
 - CircuitPython (Adafruit)
 - ...
 - Assembly

File Edit Sketch Tools Help



ESP8266-code § wifi-secrets.h

wifi-secrets_template.h

```
1 #include <ESP8266WiFi.h>
2 #include <PubSubClient.h>
3 #include "wifi-secrets.h"
4
5 #define MQTT_SERVER "192.168.0.166"
6
7
8 WiFiClient espClient;
9 PubSubClient client(espClient);
10
11 int message = 0;
12
13 int lastMessage = 0;
14
15 void reconnect(){
16     // Loop until we're reconnected
17     while (client.connected() == false) {
18         if (client.connect("doorbell")) {
19             client.publish("doorBell","0");
20             client.publish("doorMessage","0");
21         } else {
22             //Serial.print("failed, rc=");
23             delay(5000);
24         }
25     }
26 }
27
28 void setup() {
29     WiFi.begin(SECRET_SSID, SECRET_PASS);
30     while (WiFi.status() != WL_CONNECTED){
31         delay(500);
32     }
33
34     client.setServer(MQTT_SERVER, 1883);
35
36     pinMode(0, INPUT);
37     pinMode(1, INPUT);
38     pinMode(2, INPUT);
39     pinMode(3, INPUT);
40 }
41
42 void loop() {
43     client.loop();
44     if (!client.connected()){
45         reconnect();
46     }

```

File Edit Sketch Tools Help



ESP8266-code \$

wifi-secrets.h

wifi-secrets_template.h

```
45     reconnect();
46 }
47
48 if (digitalRead(3) == HIGH){
49   client.publish("doorBell","door");
50   delay(10000);
51 }
52 int newMessage=0;
53 lastMessage = message;
54 if (digitalRead(0)== LOW){
55   if (lastMessage != 1){
56     message = 1;
57     newMessage=1;
58   }
59 } else if (digitalRead(1)== LOW){
60   if (lastMessage != 2){
61     message = 2;
62     newMessage = 1;
63   }
64 } else if (digitalRead(2)== LOW){
65   if (lastMessage !=3){
66     message = 3;
67     newMessage = 1;
68   }
69 } else {
70   if (lastMessage != 0){
71     message =0;
72     newMessage = 1;
73   }
74 }
75 if (lastMessage != message){
76   //publish no message
77   client.publish("doorMessage","0");
78 }
79 if (newMessage == 1){
80   switch (message){
81     case 0:
82       //No message
83       client.publish("doorMessage", "0");
84       break;
85     case 1:
86       //Set message 1
87       client.publish("doorMessage", "1");
88       break;
89     case 2:
90       //Set message 2
91       client.publish("doorMessage", "2");
92       break;
93     case 3:
94       //Set message 3
95       client.publish("doorMessage", "3");
96       break;
```



```
1 import paho.mqtt.client as mqtt
2 import pygame
3 import pygame.freetype
4
5 def on_message(client, userdata, msg):
6     print(msg.payload)
7     if (str(msg.payload) == "b'0'"):
8         print("Please ring door bell ----->")
9         screen.fill((0,0,0))
10        screen.blit(message0, (400 - message0.get_width() // 2, 240 - message0.get_height() // 2))
11        pygame.display.flip()
12    if (str(msg.payload) == "b'1'"):
13        print("Just Comming, please wait")
14        screen.fill((255,255,255))
15        screen.blit(message1, (400 - message0.get_width() // 2, 240 - message0.get_height() // 2))
16        pygame.display.flip()
17    if (str(msg.payload) == "b'2'"):
18        print("We're out, please leave parcels in the shed")
19        screen.fill((255,255,255))
20        screen.blit(message2, (400 - message0.get_width() // 2, 240 - message0.get_height() // 2))
21        pygame.display.flip()
22    if (str(msg.payload) == "b'3'"):
23        print("We're in the garden, please come around to the back gate")
24        screen.fill((255,255,255))
25        screen.blit(message3, (400 - message0.get_width() // 2, 240 - message0.get_height() // 2))
26        pygame.display.flip()
27
28 pygame.init()
29 screen = pygame.display.set_mode((0,0), pygame.FULLSCREEN)
30 #screen = pygame.display.set_mode((0,0))
31 pygame.mouse.set_visible(0)
32 font = pygame.font.SysFont("comicsansms", 72)
33
34 message0 = font.render("Please ring door bell ----->", True, (255,255, 255))
35 message1 = font.render("Just comming, please wait", True, (0,0, 0))
36 message2 = font.render("We're out, please leave parcels in the shed", True, (0,0, 0))
37 message3 = font.render("We're in the garden, please come around to the back garden", True, (0,0, 0))
38
39 screen.fill((0,0,0))
40 screen.blit(message0, (400 - message0.get_width() // 2, 240 - message0.get_height() // 2))
41 pygame.display.flip()
42
43 client = mqtt.Client()
44 client.on_message = on_message
45
46 client.connect("192.168.0.166", 1883, 60)
47 client.subscribe("doorMessage",0)
48
49 running = True
50 while running:
51     for event in pygame.event.get():
52         if event.type == pygame.QUIT:
53             running= False
54         elif event.type == pygame.KEYDOWN:
55             if event.key == pygame.K_ESCAPE:
56                 running = False
57     client.loop()
58
59 pygame.quit()
```

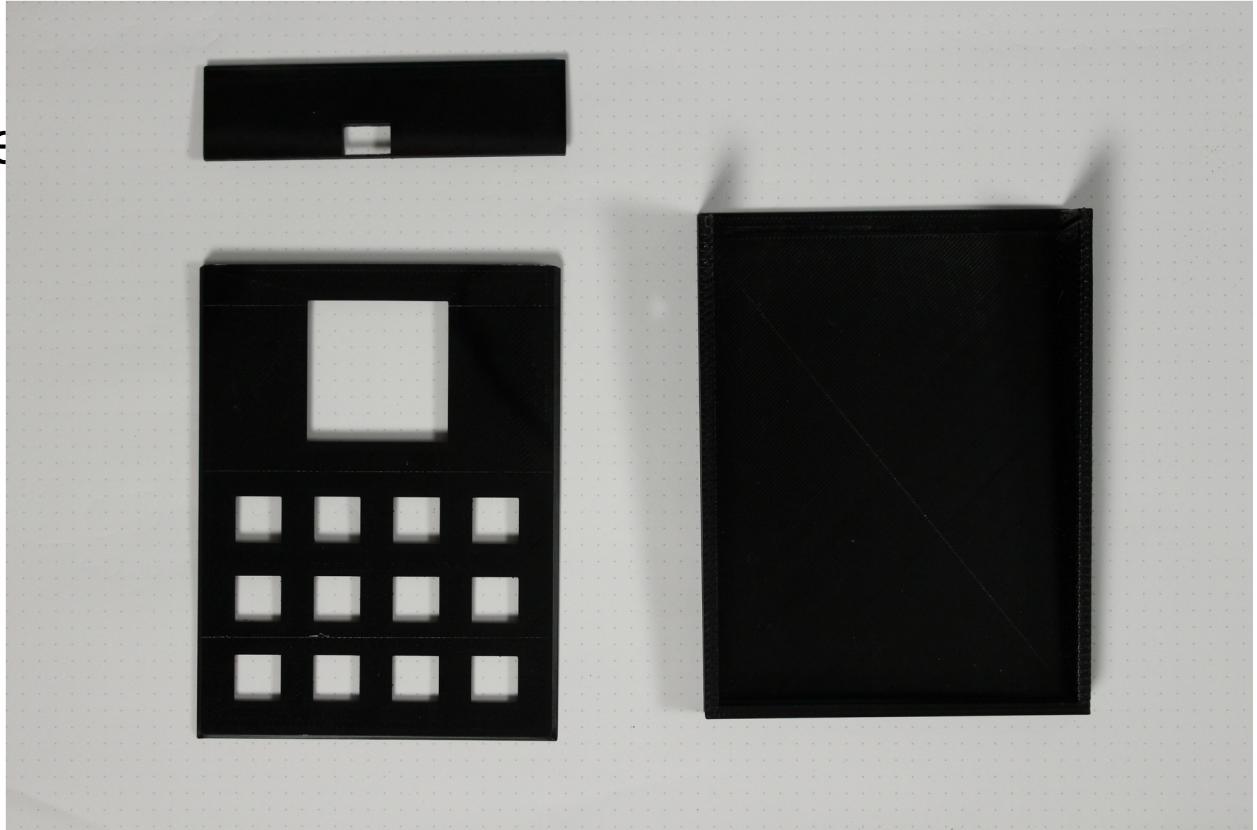
Case

- Bought case
 - Drill holes as required



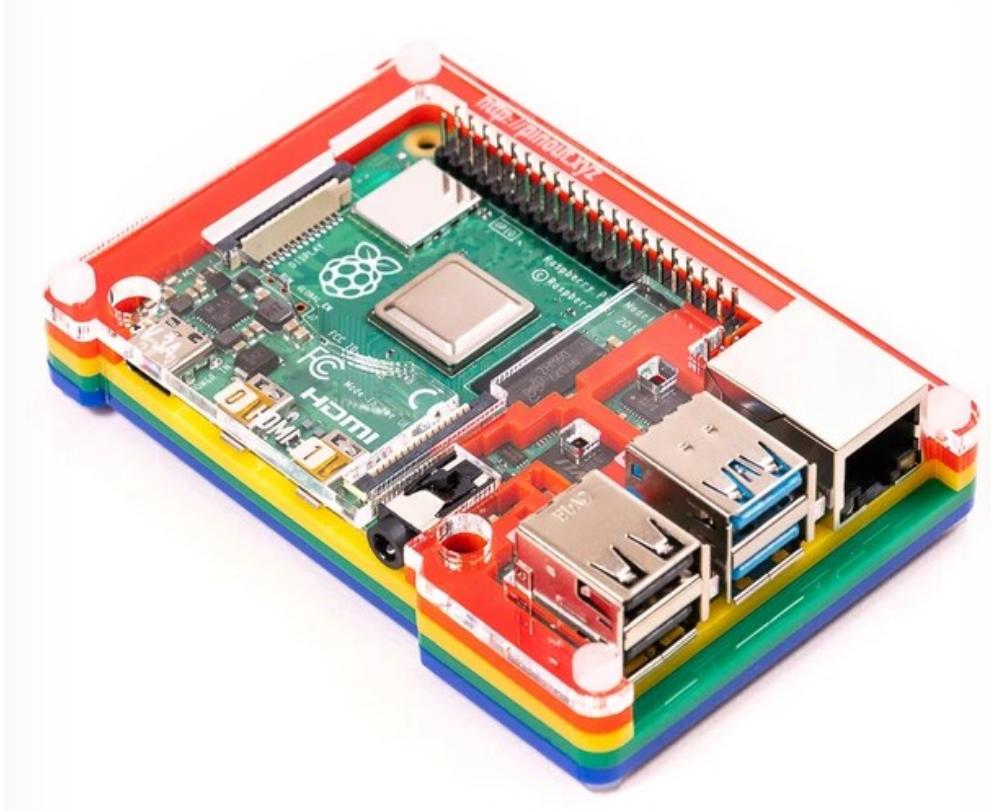
Case

- Bought case
 - Drill holes as required
- 3D printed



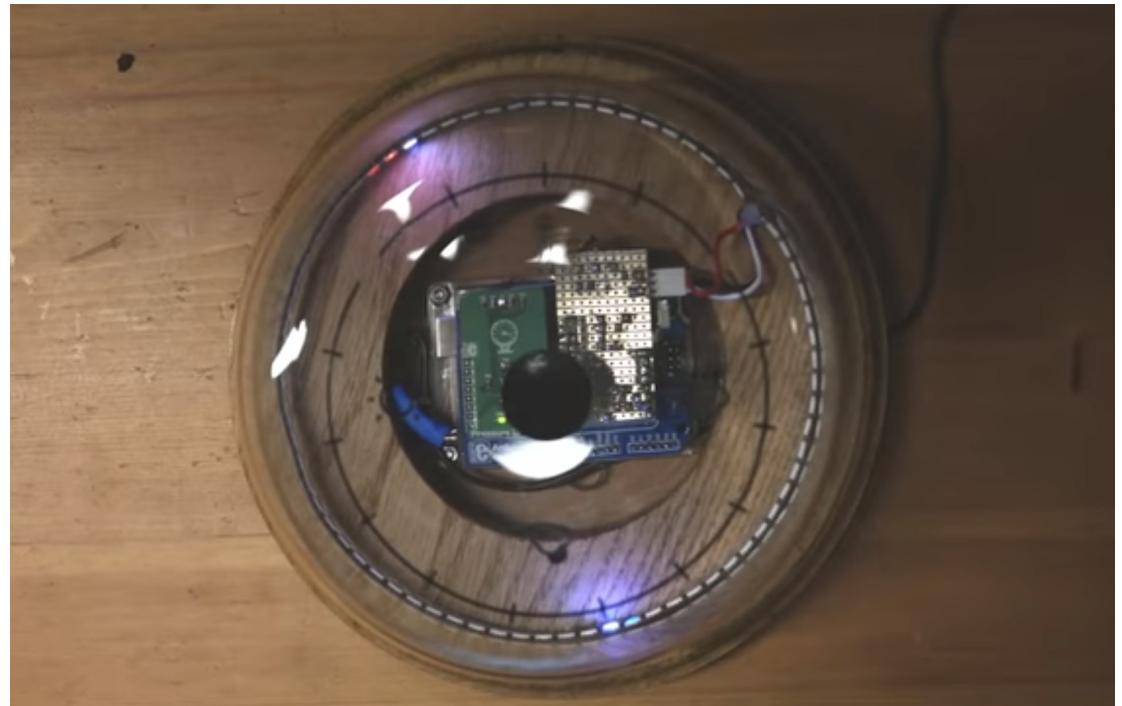
Case

- Bought case
 - Drill holes as required
- 3D printed
- Laser cut



Case

- Bought case
 - Drill holes as required
- 3D printed
- Laser cut
- Other Materials



Case

- Bought case
 - Drill holes as required
- 3D printed
- Laser cut
- Other Materials



Next...

- Test
- Fix
- Improve
- Think of the next project!

Any Questions?

- These slides can be found at:

https://github.com/hifromkatie/HowToMake_Slides