

装置名	その回路には裏がある
装置番号	000-003
作問者	Takana Norimasa
作問日	2021 9/29
制限時間	15m00s

1 作問者より一言

うまい話や不可解な謎には大抵「裏」があるもんです。裏にあるものごとをよく探ることが解決の糸口になることがしばしば。

ヒント：データシートを見るのも良いですが、よく使われてる部品ならインターネットにおいてあるコード例で概略を掴めるかもしれません。（最も間違ったコードの可能性もありますが。）

もう一つアドバイスを。作問するときも装置を解くときも大事なのは「慎重であること」です。
あなたの目の前にあるものは擬似的ではありますが爆弾です。お気をつけて。

2 回路

回路の全体の写真を図 1 に示す。

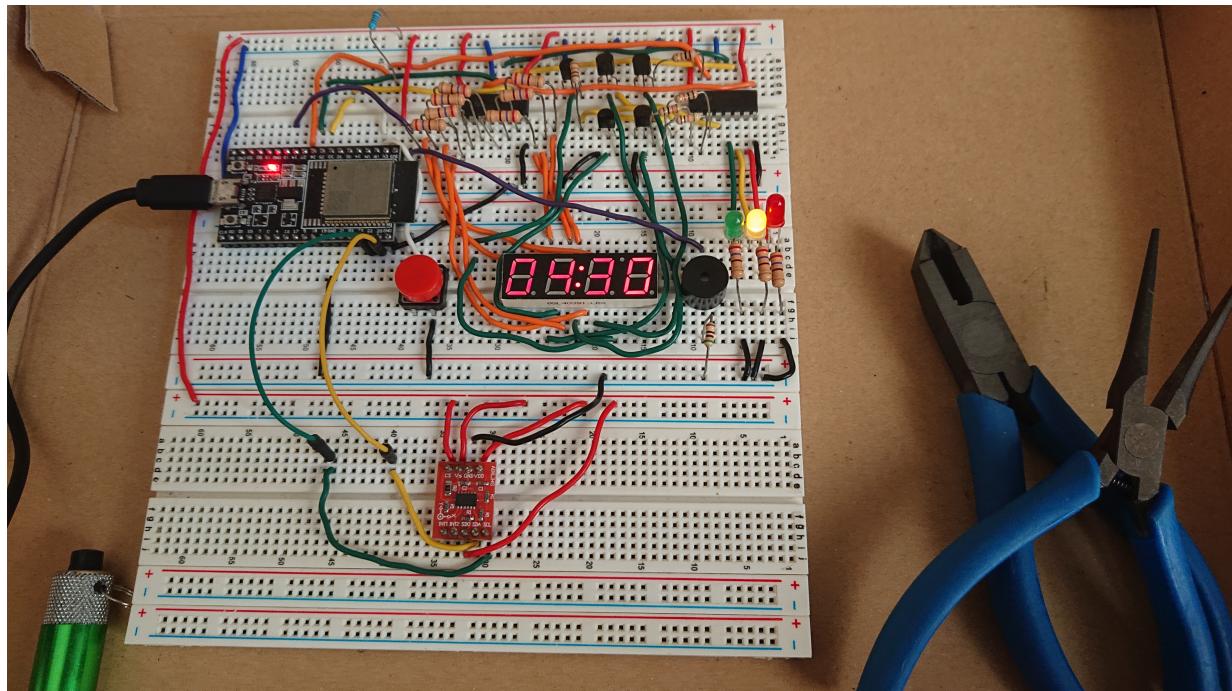


図 1: 回路全体の写真

また、回路に使用する部品の一覧を表 1 に示す。

表 1: 回路に使用する部品の一覧

部品名	個数	データシート
ADXL345	1	https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf

3 ソースコード

リスト 1 に競技に使用するソースコードを示す。

Listing 1: timer.ino

```
1 int time_limit = 900;
2 uint8_t acce_data[6];
3 const uint8_t DEVICE_ADDRESS = 0x1D;
4
5 // giver pin assgin
6 const uint8_t mySCL = 22;
7 const uint8_t mySDA = 23;
8
9 void setup_pin(void) {
10     pinMode(mySCL, INPUT_PULLUP);
11     pinMode(mySDA, INPUT_PULLUP);
12     Serial.begin(115200);
13
14     Wire.begin(mySDA, mySCL);
15     Wire.beginTransmission(DEVICE_ADDRESS);
16     Wire.write(0x31);
17     Wire.write(0x0B);
18     Wire.endTransmission();
19     Wire.beginTransmission(DEVICE_ADDRESS);
20     Wire.write(0x2d);
21     Wire.write(0x08);
22     Wire.endTransmission();
23 }
24
25 void getAccelerationData(void) {
26     Wire.beginTransmission(DEVICE_ADDRESS);
27     Wire.write(0x32);
28     Wire.endTransmission();
29
30     Wire.requestFrom(DEVICE_ADDRESS, 6);
31
32     for(int i = 0; i < 6; i++) {
33         delay(1);
34         acce_data[i] = Wire.read();
35     }
36 }
37
38 void gaming(void *pvParameters) {
39     bool flag1 = false;
40     bool flag2 = false;
41     bool flag3 = false;
42     bool flag4 = false;
43     double ax = 0;
44     double ay = 0;
45     double az = 0;
46
47     while(1) {
48         getAccelerationData();
```

```
49     ax = (int16_t)((acce_data[1] << 8) | acce_data[0]) * 0.0392266;
50     ay = (int16_t)((acce_data[3] << 8) | acce_data[2]) * 0.0392266;
51     az = (int16_t)((acce_data[5] << 8) | acce_data[4]) * 0.0392266;
52
53     Serial.printf("ax: %g, ay: %g, az: %g\n", ax, ay, az);
54
55     flag1 = ax > 2.0;
56     flag2 = ay < -2.0;
57     flag3 = az > 9.0;
58     flag4 = az > 12.0;
59
60     // succeeded
61     if(flag1 && flag2 && flag3) {
62         succeeded();
63     }
64
65     // failed
66     if(flag4) {
67         failed();
68     }
69 }
70 }
```

Listing 1: 競技に使用するソースコード