

# 自作目覚まし時計の制作

石田 翔真・岡崎 由恆  
佃 一旗・中嶋 遥登

## 1. 研究概要

自分たちは一人で朝起きるのが苦手で朝起きるために目覚まし時計を作成することで配線や時間を利用したプログラムなどを学ぶために時計表示機能やアラーム機能などを備えた目覚まし時計を作成した。

## 2. 研究の具体的内容

現在時刻の取得、アラームが作動する時間の設定、現在時刻と設定した時間の判定のプログラム、機能を搭載するための木箱の作成を行った。

### <全体のプログラムの仕様>

最初にテンキーパッドでアラームの作動時間を入力し、送信する。  
次に現在時刻を RTC モジュールで取得し、現在時刻と設定した時間を LCD に出力する。  
最後に現在時刻と設定した時間が一致したらアラームが作動し、最初に戻る。

### <機能ごとの説明>

(1) 開発環境として Arduino IDE を使った。  
使用したボードはピンソケットを LCD とテンキーパッドで多く使うためピンソケットが多い Arduino Mega 2560 を使用した。



図 1 Arduino の基板

(2) 現在時刻を取得するプログラムにはアクセスポイントの SSID を必要とせず簡単に時間を取得できる RTC モジュールを使用した。



図 2 RTC モジュール

時間取得のプログラム(抜粋)

```
#include<Wire.h>
#include<DS3231.h>
DS3231 clock;
RTCDateTime dt;
clock.begin();
clock.setDateTime(__DATE__, __TIME__);
dt = clock.getDateTime(); /*変数 dt に現在時刻を入れる関数。値は整数型変数 dt.hour に時間、dt.minute に分が格納される*/
```

(3) アラームの作動する時間の入力、送信、取り消しの機能をテンキーパッドのモジュールを使って実装した。



図 3 テンキーパッド

アラームの設定時刻の入力のプログラム

```
#include<Keypad.h>
char keys[ROWS][COLS] = {
  { '1', '2', '3', 'A' },
  { '4', '5', '6', 'B' },
  { '7', '8', '9', 'C' },
  { '*', '0', '#', 'D' }
}; /*テンキーパッドを押した際に入力される文字を決める。グローバル変数として宣言*/
```

```
while(key_flag == 0) {
  readKeypad();
} /*アラームの作動する時間を取得する関数 readKeypad を作動する時間が送信されたら抜ける*/
```

```
void readKeypad()
{
  char key = keypad.getKey();
  if (key != NO_KEY) {
    switch(key) {
      case '*':
        z=0;
        for (int z=0; z<4; z++) {
          attempt[z]=0;
        }
        lcd.clear();
        lcd.print("Cleared!");
        lcd.setCursor(0,1);
        lcd.print("INPUT_settime");
        time_flag = 0;
        break;
      case '#':
        delay(100);
        if(time_flag == 0) {
          key_flag = 1;
        }
        break;
      default:
```

```
if(time_flag == 0) {
  attempt[z] = key;
  z++;
}
break;
}
}
```

/\*関数 readKeypad で設定時刻を取得。数字が押されたら押された数字を一次元配列 attempt に入力、\*を押したら配列の中身を初期化、#を押したら関数 readKeypad を抜け、現在時刻との判定を始める。A、B、C、D は押されても反応しない。\*/

(4) 現在時刻とアラームの時間と文字を表示するために上の行と下の行の二行出力することができる LCD モジュールを使った。



図 4 LCD モジュール

LCD に出力するプログラム (抜粋)

```
#include<LiquidCrystal.h>
LiquidCrystal lcd(7, 8, 9, 10, 11, 12) /*使用したピンの番号*/

lcd.begin(16, 2);
lcd.clear(); /*lcd.clear 関数で lcd の出力を初期化*/

lcd.setCursor(0, 0); /*lcd.setCursor 関数で出力する場所を移動*/

lcd.print("Now_Time.");
lcd.setCursor(9, 0);
lcd.print(dt.hour); /*lcd.print 関数で出力*/

lcd.setCursor(11, 0);
```

```

lcd.print(":");
lcd.setCursor(12, 0);
lcd.print(dt.minute);
lcd.setCursor(0, 1);
lcd.print("Set_Time.");
lcd.setCursor(9, 1);
lcd.print(hour);
lcd.setCursor(11, 1);
lcd.print(":");
lcd.setCursor(12, 1);
lcd.print(min); /*上の行に現在の時刻、
下の行にアラームの作動する時刻を出力する
*/

```

- (5) microSD に保存した MP3 ファイルを再生することができ、スピーカーを繋げるだけで音が鳴るため使いやすい DFPlayer mini を使用した。microSD は別に買う必要がある。



図 5 DFPlayer mini

- (6) 設定した時間になったら音を出すスピーカーを実装した。



図 6 DFPlayer mini と接続したスピーカー

```

アラームを鳴らすプログラム(抜粋)
#include "Arduino.h"
#include "SoftwareSerial.h"
#include "DFRobotDFPlayerMini.h"
SoftwareSerial mySoftwareSerial(2, 3);
/*使用したピンの番号*/

DFRobotDFPlayerMini myDFPlayer;
myDFPlayer.volume(1); /*ボリュームをセッ
ト。(ボリュームは 0 から 30 の値で指定可
能)*/

myDFPlayer.play(1); /*先頭の mp3 ファイル
を再生*/

```

- (7) 作成した機能を木で作った箱に搭載した。箱は横幅 25cm、縦幅 16cm、高さ 13cm と少し大きい時計が完成した



図 7 完成品

```

完成したプログラム (抜粋)
dt = clock.getDateTime(); /*この行で現在
時刻を取得*/
String str = String(attempt); /*文字列型
変数 str に作動時刻を格納する一次元配列
attempt を文字型に変換して入れる*/

String strHour = str.substring(0, 2); /*
文字列型変数 strHour に変数 str の先頭の

```

二文字(時間)を入れる\*/

```
String strMinutes = str.substring(2, 4);  
/*文字列型変数 strMinutes に変数 str の  
後ろの二文字(分)を入れる*/
```

```
int hour = strHour.toInt(); /*文字列型変  
数 strHour を整数型にして整数型変数 hour  
に入れる*/
```

```
int min = strMinutes.toInt(); /*文字列型  
変数 strMinutes を整数型にして整数型変  
数 min に入れる*/
```

```
/*この後に LCD で現在時刻と設定した時刻を  
表示する*/
```

```
if(dt.hour == hour && dt.minute == min) {  
  /*dt.hour と hour、dt.minute と min の値  
  が同じときアラームを鳴らす*/  
}
```

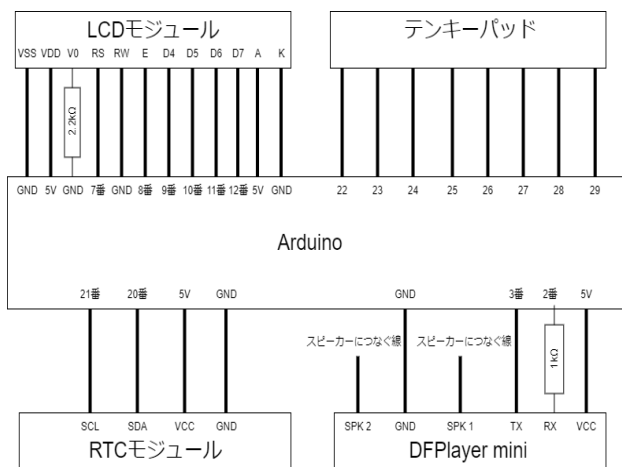


図 8 配線をした際の回路図

### 3. 研究のまとめ

今までの授業や実習、部活動で学んだ知識が活用できた。しかし、モジュールの使い方や時間取得の方法などわからないものが多かったが課題研究を通して学ぶことができた。計画を立てていたが予定通りには進まず焦りがあったが作業を分担して終わらせることができた。大変だったことは各機能のプログラ

ムを一つにまとめるのが大変だった。

参考文献

LCD ディスプレイ

<https://zenn.dev/suzuky/articles/73552d9deccc79>

RTC モジュール

<https://laboratory.kazuuu.net/using-the-clock-module-ds3231-rtc-chip-ds1307/>

テンキーパッド

<https://atatat.hatenablog.com/entry/2020/04/14/205913>

DFPlayer mini

<https://zenn.dev/suzuky/articles/deca20bb6e8499>