

高気温時警告システムの仕様書

25G1051 近藤巧望

2025 年 7 月 20 日

1 本システムの概要

近年、気温の上昇に伴い、熱中症のリスクが高まっている。特に、夏季の高温環境下では、熱中症の発症が増加しており、その予防と対策が重要な課題となっている。熱中症は、重篤な場合には生命に関わることもあり、一命を取り留めたとしても高次脳機能障害などの後遺症が残ることがあるため非常に危険である。本人は意外にも自身が熱中症にかかっていることに気づかない事が多く、熱中症への意識を高めることが重要である。そこで今回、私は高気温時警告システムを提案する。このシステムは、気温のリアルタイム表示に加え気温が 30 度を超えた場合にアラームで警告し、熱中症への対策を促すことを目的としている。これにより、現在の気温が高いことを意識させ、熱中症の予防につなげることができると考えられる。主に気温検知用の温度センサー、気温のリアルタイム表示用の 7 セグメントディスプレイ、アラームを鳴らすための圧電ブザー、およびシステム全体を制御する Arduino UNO R3 で構成される。

2 本システムの構成

2.1 入力・処理・出力の構成

表 1 本システムの構成要素

項目	構成	役割
1	温度センサー [TMP36]	気温を検知する
2	7 セグメントディスプレイ (クロック表示)	気温を表示する
3	圧電ブザー	アラームで警告する
4	Arduino UNO R3	システム全体を制御する

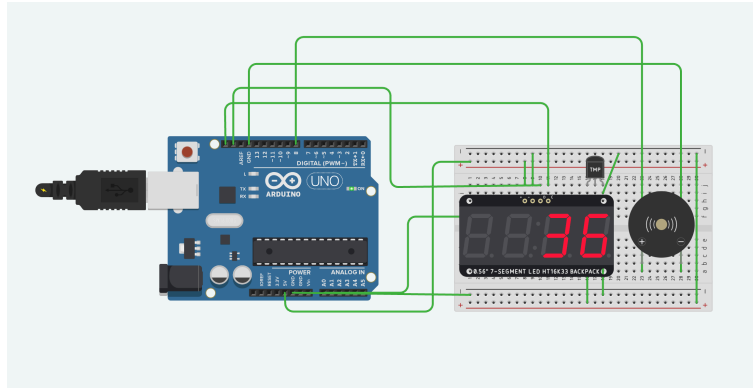


図1 本システムの動作の様子。電圧は 5.0V であり、アナログ値の取得のために A0 ピンに接続されている。圧電ブザーには 8 ピンを使用し、7 セグメントディスプレイには SDA ピンと SCL ピンを使用している。なお、7 セグメントディスプレイは 0x70 のアドレスで接続されている。

本システムの構成要素を表 1 に、動作の様子を図 1 に示す。まず入力として、温度センサー [TMP36] を用いて、A0 ピンから温度センサーのアナログ電圧を取得する。次に、温度センサーからのアナログ電圧を Arduino UNO R3 でデジタル値に変換し、気温を算出する。アナログ電圧から気温を算出する式は (1)(2) 式の通りである。

$$V_{in} = \frac{\text{analogRead}(A0)}{1023} \times V_{ref} \quad (1)$$

$$T [^{\circ}C] = (V_{in} - 0.5) \times 100 \quad (2)$$

ここで、 V_{ref} は Arduino UNO R3 の基準電圧 5.0V であり、 T は気温である。また、(1)(2) 式から求められた気温をもとに、7 セグメントディスプレイの表示制御と圧電ブザーのアラーム制御を行う。これらの一連の処理は Arduino UNO R3 のプログラムにより実行される。最後に、7 セグメントディスプレイと圧電ブザーで出力を行う。

2.2 プログラムの概要

Listing 1 Arduino UNO R3 のソースコード

```
1 #include <Adafruit_LEDBackpack.h>
2 #include <Wire.h>
3
4 const float vRef = 5.0;
5
6 Adafruit_7segment ht = Adafruit_7segment();
7
8 void setup() {
9   Serial.begin(9600);
10  pinMode(A0, INPUT);
11  pinMode(8, OUTPUT);
12
13  ht.begin(0x70);
14 }
15
16 int readtemperatureC(int pin){
17
18   int ad = analogRead(pin);
19
20   float voltage = ad * vRef / 1023.0;
21   int temperatureC = (voltage - 0.5) * 100.0;
22
23   return temperatureC;
24 }// ピンから取得したアナログ値を基に気温を算出する関数
25
26 void loop() {
27   int temperatureC=readtemperatureC(A0);
28
29   ht.print((int)temperatureC); // 気温を整数で表示
30
31   ht.writeDisplay(); ディスプレイ上に表示//
32
33   if (temperatureC > 30) {
34
35     tone(8, 523);
36     delay(600);
37     noTone(8);
38     delay(600);
39   }
40 }
```

2.2.1 プログラム全体の説明

Arduino UNO R3 のプログラムは上記のとおりである。また、本システムで使用したライブラリは Adafruit_LEDBackpack と Wire であり、プログラム全体の概要は以下の通りである。

- ・ 初期化処理として、シリアル通信の開始、A0 ピンの入力モード設定、8 ピンの出力モード設定、7 セグメントディスプレイの初期化を行う。
- ・ ループ処理では、まず A0 ピンからアナログ値を読み取り、それを基に電圧を計算する。
- ・ 次に、計算した電圧から気温を算出する。
- ・ 7 セグメントディスプレイに気温を表示する。
- ・ 気温が 30 度を超えた場合、圧電ブザーを鳴らして警告する。
- ・ 圧電ブザーは 523Hz の音を 600 ミリ秒間鳴らし、2 回繰り返す。

2.2.2 関数の説明

表 2 使用した関数の一覧

関数名	引数	戻り値	機能
Serial.begin	baudrate	void	シリアル通信を開始する
analogRead	pin	int	指定したピンのアナログ値を読み取る
pinMode	pin, mode	void	指定したピンのモードを設定する
ht.begin	address	void	7 セグメントディスプレイを初期化する
ht.print	value	void	7 セグメントディスプレイに値を表示する
ht.writeDisplay	なし	void	7 セグメントディスプレイに表示を反映する
tone	pin, frequency	void	指定したピンで指定した周波数の音を鳴らす
noTone	pin	void	指定したピンの音を止める
delay	milliseconds	void	指定した時間だけ処理を停止する
readtemperatureC	pin	int	指定したピンからアナログ値を読み取り、気温を算出する

本システムで使用した関数の説明を表 2 に示す。まず、Listing1 の 8 行目から 14 行目の `setup` 関数内では、`Serial.begin` によるシリアル通信の開始、`pinMode` による A0 ピンの入力モード設定と 8 ピンの出力モード設定、`ht.begin` による 7 セグメントディスプレイの初期化を行う。`Serial.begin` では引数としてボーレートを指定し、今回は 9600bps を指定している。`pinMode` では引数としてピン番号とモードを指定し、A0 ピンは入力モード、8 ピンは出力モードに設定している。`ht.begin` では引数として 7 セグメントディスプレイのアドレスを指定し、今回は 0x70 を指定している。次に、16 行目から 24 行目では、電圧のアナログ値を読み取り、気温を算出する `readtemperatureC` 関数を定義している。この関数では、`analogRead` 関数を使用して A0 ピンからアナログ値を取得し、それを基に電圧を計算している。また、引数はピン番号であり、その後は計算した電圧から気温を算出し、整数値として返す処理を行っている。26 行目から 46 行目の `loop` 関数では、気温を 7 セグメントディスプレイに表示させたり圧電ブザーを鳴らすための制御を行っているが、27 行目では前述した `readtemperatureC` 関数を呼び出して A0 ピンから気温を取得し、`temperatureC` 変数に格納している。その後、`ht.print` 関数を使用して 7 セグメントディスプレイに気温を表示し、`ht.writeDisplay` 関数で表示を反映

させている。さらに 33 行目からは、if 文を使用して気温が 30 度を超えた場合には圧電ブザーを鳴らす処理を行っている。圧電ブザーは tone 関数を使用して 8 ピンで 523Hz の音を 600 ミリ秒間鳴らし、noTone 関数で音を止めた後、さらに 600 ミリ秒間の待機を行っており、これらの処理によって圧電ブザーのアラームを実現している。

3 本システムの実動作

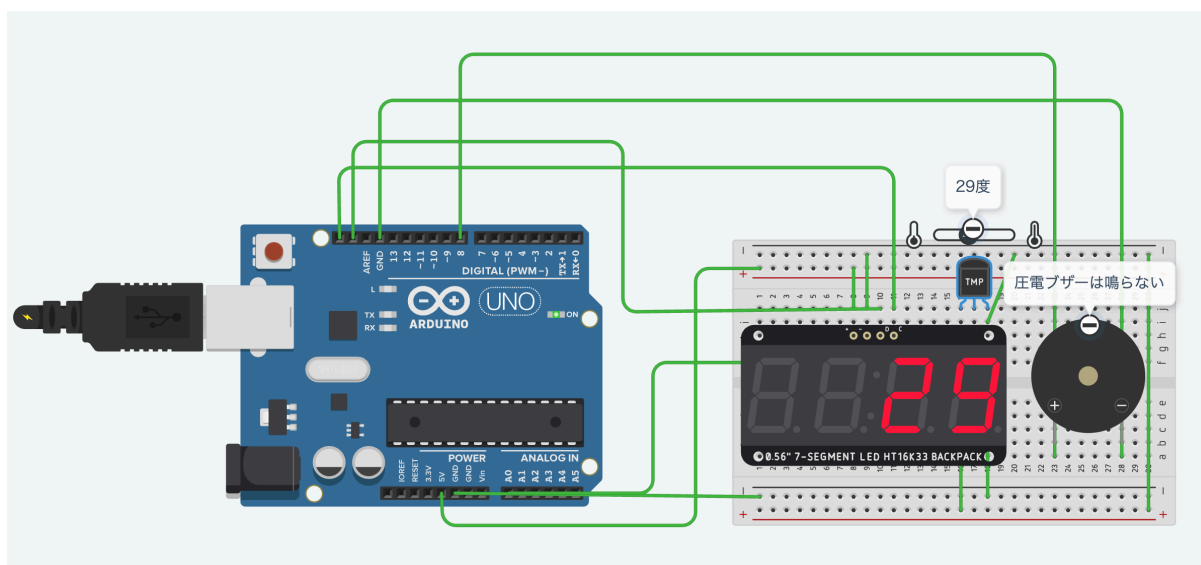


図 2 30 度以下の気温でのシステムの動作.

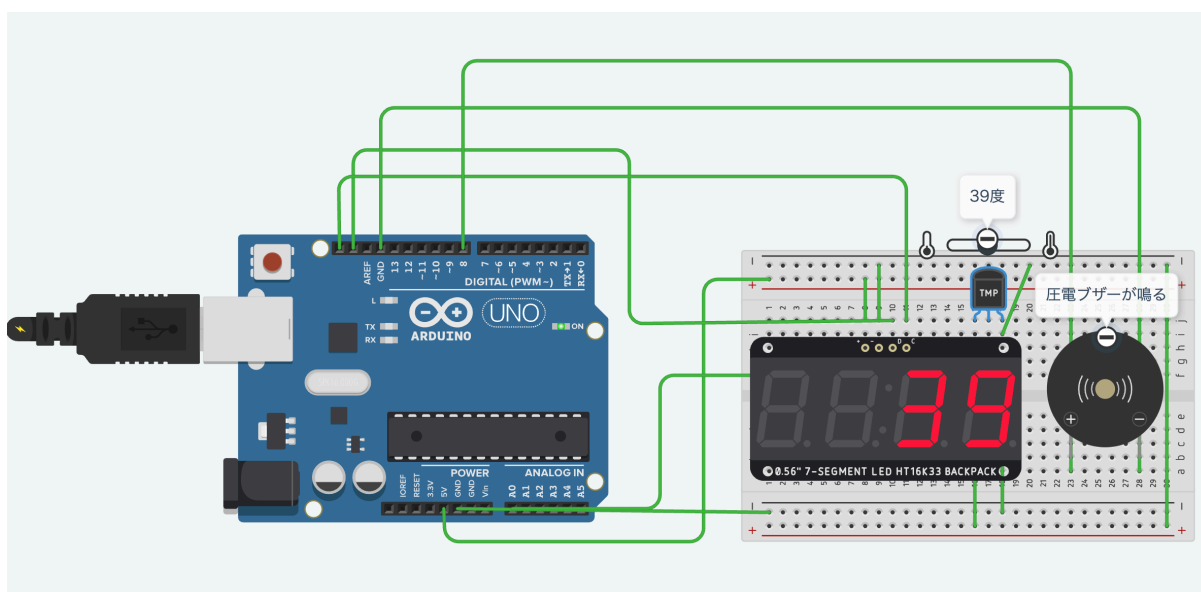


図 3 30 度を超える気温でのシステムの動作.

本システムの実動作を図2と図3に示す。図2は30度以下の気温でのシステムの動作を示しており、7セグメントディスプレイには30度以下の気温が表示されている。この場合、圧電ブザーは鳴らず、システムは正常に動作していることが確認できる。一方、図3は30度を超える気温でのシステムの動作を示しており、7セグメントディスプレイには30度を超える気温が表示されている。この場合、圧電ブザーが鳴り、アラームで警告が行われていることが確認できる。

4 本システムのアピールポイント

本システムは、講義内で用いた温度センサーと7セグメントディスプレイ・圧電ブザーを組み合わせて、気温のリアルタイム表示とアラームによる警告を実現している点が特徴である。特に、7セグメントディスプレイは講義内では固定の値を表示するために使用されていたが、本システムでは温度センサーからの入力値を変換し、求められた気温を変数として扱うことで、リアルタイムに表示することが可能となっている。また、これらの要素の組み合わせは、熱中症対策という当初の目的に対して効果的なものとなっており、システムとしての実現可能性と妥当性もあると考えられる。

5 注意事項

本システムの温度検知において、Arduino UNO R3 の処理によって求められた気温は、小数点以下を切り捨てた整数値であるため、表示される温度は実際の温度よりも低くなる場合がある。そのため、表示される温度が30度を超えた場合でも、実際の温度が30度以下である可能性があることに注意が必要である。また、圧電ブザーの音量はArduino UNO R3 の出力電圧に依存するため、環境によっては音が小さく感じられることがある。そのため、圧電ブザーの音量を調整するためには、ブザーの電源電圧を変更するか、ブザー自体の仕様を確認する必要がある。さらに、7セグメントディスプレイの表示は、Arduino UNO R3 の処理速度に依存するため、表示が遅れることがあり、表示が遅れる場合はプログラムの処理速度を調整するか、7セグメントディスプレイの仕様を確認する必要がある。

6 おわりに

本稿では、高気温時警告システムの概要や構成、プログラムの説明、実動作の様子を示した。本システムは、気温のリアルタイム表示とアラームによる警告を実現しており、熱中症対策に有効であると考えられる。実際の環境での使用にあたっては、注意事項を考慮し、適切な設定や調整を行うことが重要である。また、今回はアクチュエータとして圧電ブザーを使用した。より大きな音量が必要な場合はスピーカーなどの他のアクチュエータを使用することも検討できる。単純なシステムであるが単純故に拡張性が高く、今後の改良や機能追加が容易であり、さらなる発展が期待できる。

7 Tinkercad のリンク

https://www.tinkercad.com/things/6XujYAWZ3oc-/editel?returnTo=%2Fdashboard%2Fdesigns%2Fcircuits&sharecode=rltL3EJg_tPyliN_Yu4awmcOKybTstgs-JpTvKHY65Y