

Dokumentacja

Bezkontaktowy miernik temperatury cieczy z funkcją alarmu - „Kubkuino”

Marek Ledworowski

ZS36 im. Marcina Kasprzaka w Warszawie

Klasa 3A

Opiekun: Piotr Wróblewski



Spis treści:

- **Cel powstania urządzenia**
- **Specyfikacja**
- **Schemat i wykonanie urządzenia**
- **Komponenty**
- **Budowa**
- **Kod źródłowy mikrokontrolera**
- **Komendy do komunikacji przez bluetooth**
- **Przykładowy przebieg komunikacji**
- **Plany na dalszy rozwój**
- **Informacje dodatkowe**

Cel powstania urządzenia

To urządzenie zostało stworzone z myślą o osobach, które nie chcą czekać aż ich gorący napój będzie dobry do picia.

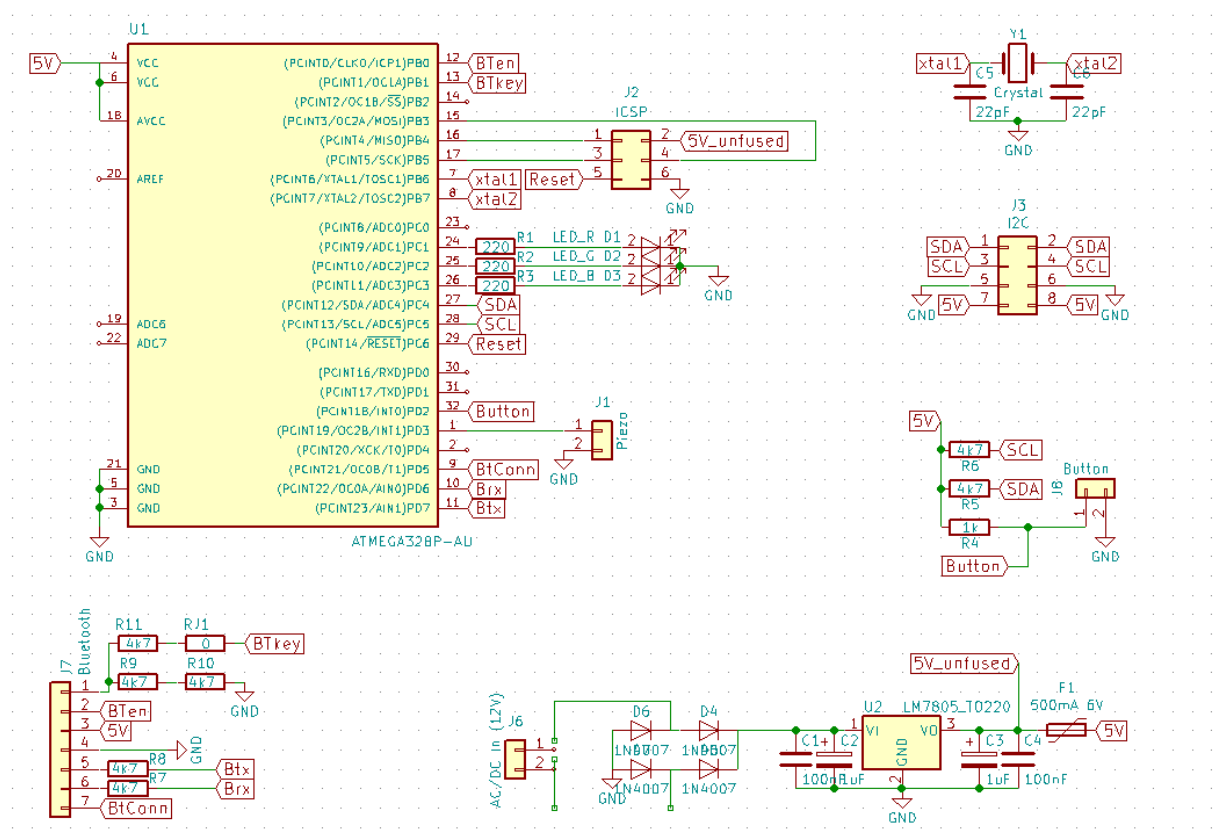
Wystarczy włączyć urządzenie, postawić napój i... poczekać!

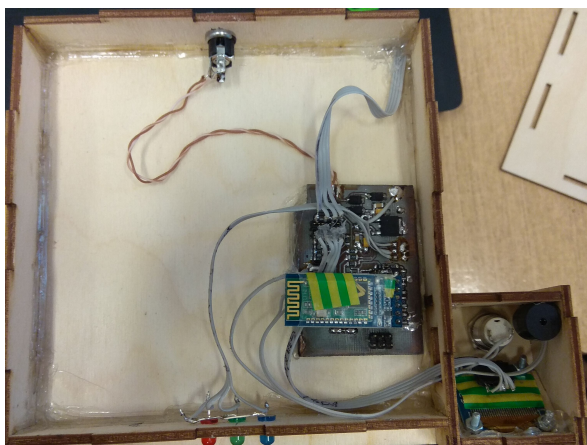
Może też mierzyć temperaturę pojemników które nie mogą zostać otwarte, np. gorąca zlewka w pracowni chemicznej czy próbki do badań w laboratoriach.

Specyfikacja techniczna

Napięcie zasilania	7 – 30V DC / 10-15V AC
Prąd pobierany z zasilacza (12V DC)	Max 100mA / typ. 35-50mA
Temperatura pracy	0 - 30 st.C.
Odporność na zalania	Tak (częściowo)

Schemat urządzenia i wykonanie





Komponenty użyte do budowy urządzenia

- Moduł Bluetooth HC-05 (złącze J7) – zmodyfikowany aby użyć pełni funkcji jakie oferuje moduł
- Podczerwony czujnik temperatury GY-906 (J3) o FOV 90 stopni
- Wyświetlacz OLED 128x64 oparty na sterowniku SSD1306 (J3)

Urządzenie posiada gniazdo na zasilacz z wtyczką DC2.1/5.5 dzięki czemu może zostać zasilone praktycznie każdym zasilaczem.

Polaryzacja zasilania nie jest określona.

Budowa

Urządzenie składa się z dwóch części: komory na naczynie i części sterującej z wyświetlaczem.

Obudowa została wykonana z drewnianych modułów co pozwoliło na:

1. Obniżenie kosztów prototypu
2. Zmniejszenie wagi urządzenia
3. Łatwe nanoszenie poprawek konstrukcyjnych

Naczynie powinno być wysokie na tyle, aby poziom cieczy był w odległości 4-5cm od czujnika zamontowanego na górze komory.

Kod źródłowy mikrokontrolera

Program został napisany za pomocą środowiska Arduino.

Użyta została biblioteka ClickButton (na licencji GPLv3, autor: raronzen@gmail.com) do której wprowadziłem pewne modyfikacje.

Obsługa komend specyficznych dla modułu Bluetooth (komend AT) również została dodana aby w razie potrzeby była możliwość zmiany jego ustawień (np. nazwa, pin parowania etc. Więcej informacji znajduje się w nocie producenta).

Aby uprościć konfigurację urządzenia stworzony został program Kubkuino. Więcej informacji o programie znajduje się w jego instrukcji.

Całość kodu źródłowego można znaleźć pod linkiem:

<http://github.com/marecl/Kubkuino>

Krótki spis funkcji programu Kubkuino

- Ustawianie temperatur granicznych
- Włączanie i wyłączanie alarmu
- Włączanie i wyłączanie elementów składowych urządzenia (diody, wyświetlacz etc)
- Zapis ustawień na urządzeniu
- Reset fabryczny, zmiana nazwy i kodu parowania BT
- Konsola do własnej komunikacji z urządzeniem

Lista komend do komunikacji z urządzeniem

X - Pierwszy argument

Y - Drugi argument

e - Nawiązanie połączenia (zwraca "HELLO\r\n")

qquit - Rozłączenie

CXX.X - Ustawienie temperatury zimna

HXX.X - Ustawienie temperatury ciepła

C?/H? - sprawdzenie temperatur

r - Temperatura z czujnika

S[komenda AT] - Przesyła komendę AT do modułu Bluetooth.

Zwraca odpowiedź modułu.

DXY - Moduł zarządzania komponentami:

0Y* - LCD
1Y* - Podświetlenie
2Y* - Bluetooth
3Y* - LED
4Y* - Wyciszenie
5 - Status LCD, Podświetlenie, LED, Alarm
6 - Przejście w tryb uśpienia
B - Przywrócenie modułu Bluetooth do ustawień fabrycznych
D - Domyślne ustawienia urządzenia
L - wczytanie ustawień z pamięci urządzenia
S - zapis ustawień do pamięci urządzenia

*Jak jako argument poda się znak zapytania (?) zostanie zwrócony stan pojedynczego modułu, np. H? / D0? / D3? / M?

Uwaga! Każda komenda jest zakończona średnikiem ';'!
W przypadku łączenia się innym programem należy wyłączyć wysyłanie znaków końca linii.

Przykładowy przebieg komunikacji

```
e;           // Nawiązanie połączenia
D5;          // Odczyt co jest włączone
H?;          // Odczyt górnej temperatury granicznej
C?;          // Odczyt dolnej temperatury granicznej
r;           // Odczyt bieżącej temperatury
H70;         // Ustawienie górnej temperatury granicznej na 70 stopni
C50;         // Ustawienie dolnej temperatury granicznej na 50 stopni
D00;         // Wyłączenie wyświetlacza
D30;         // Wyłączenie diód
D41;         // Wyciszenie alarmu
qquit;       // Rozłączenie
```

Plany na dalsze usprawnienia

- Użycie innego rodzaju mikrokontrolera – pozwoli na pełną dowolność tworzenia programu
- Zmiana czujnika temperatury – obecnie zainstalowany czujnik GY-906 jest w wersji z 90-stopniowym FOVem czego skutkiem jest zmniejszona dokładność pomiarów. Planowana jest zmiana na wersję z 15 lub 30-stopniowym.
- Dodanie wiatraka wydmuchującego parę wodną z pojemnika

Informacje dodatkowe

Istnieje możliwość pobrania kodu źródłowego oraz schematów.

Strona domowa projektu:

<https://github.com/marecl/Kubkuino>