

Vysoké Učení Technické v Brně
Fakulta Informačních Technologií

Mikroprocesorové a vestavěné systémy

Měření teploty

Nikita Smirnov, xsmirn02

15. prosince 2024

1 Úvod

Cílem tohoto projektu je implementovat program pro měření teploty pomocí analogového čidla.

Program zpracovává každou sekundu přijatý signál z teplotního čidla a převádí jej na skutečnou teplotu. Naměřená teplota je poté odeslána do brokeru mqtt a uložena do nvs. Hlavním cílem bylo vytvořit funkční, snadno rozšiřitelný a udržitelný kód. Toho bylo dosaženo rozdělením programu do modulů, které jsou řízeny ze souboru `main.c`.

Videoprezentace funkčnosti projektu: [video](#)

1.1 Použité nástroje

- **Programovací jazyk** - C.
- **Prostředí** - ESP-IDF.
- **Deska** - Wemos D1 R32(ESP32).
- **Teplotní čidlo** - LMT85LPG.
- **IDE** - PlatformIO IDE for VSCode.

2 Implementace

Program je rozdělen do několika modulů, které obsahují realizaci konkrétních částí projektu:

- *main.c* - Odpovídá za řízení hlavní smyčky programu. Rovněž inicializuje komponenty.
- *config.h* - Obsahuje nastavení pro provoz programu a definuje konfigurační strukturu pro jeho další načítání z paměti nebo konfigurační stránky.
- *memory_handler.h* - Inicializuje a odpovídá za práci s perzistentní pamětí, včetně logiky načítání konfigurace projektu.
- *mqtt_handler.h* - Inicializuje spojení s mqtt brokerem, je zodpovědný za zpracování informací přijatých ze serveru a odesílání dat na server. V současné době podporuje pouze připojení přes TCP.
- *temp_sensor.h* - Odpovídá za řízení provozu teplotního čidla. Obsahuje také funkce pro získávání napětí a jejich převod na stupně. Vzhledem k tomu, že informace jsou přijímány ne častěji než jednou za sekundu, byla zvolena metoda využívající *esp_adc/adc_oneshot.h*

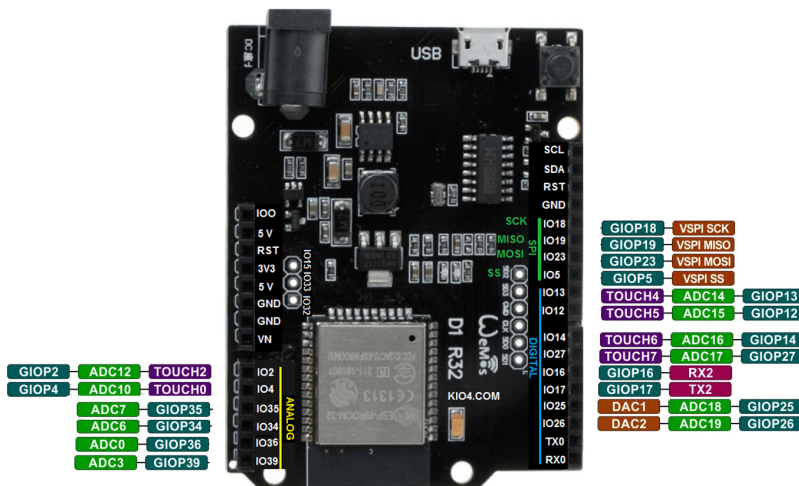
- *threshold.h* - Funkce byla vyhrazena do samostatného modulu, protože má potenciál pro rozšíření, nicméně v současné době pouze přepínáme diodu v závislosti na konfiguraci prahového tepla a hysterezi.
- *wifi_handler.h* - Odpovídá za inicializaci a zpracování událostí, které jsou generovány během připojení k wifi.

3 Spuštění

3.1 Zapojení

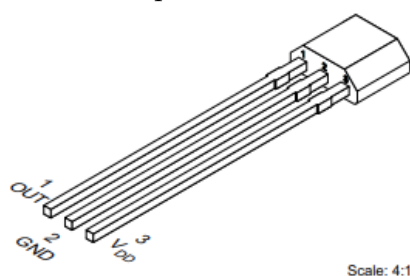
3.1.1 Deska

Používáme desku Wemos D1 R32. V našem případě používáme porty GPIOP2(ADC12) pro připojení diody a GPIOP39(ADC3) pro čtení teplotního čidla. Tyto porty lze samozřejmě v konfiguraci(*config.h*) změnit.

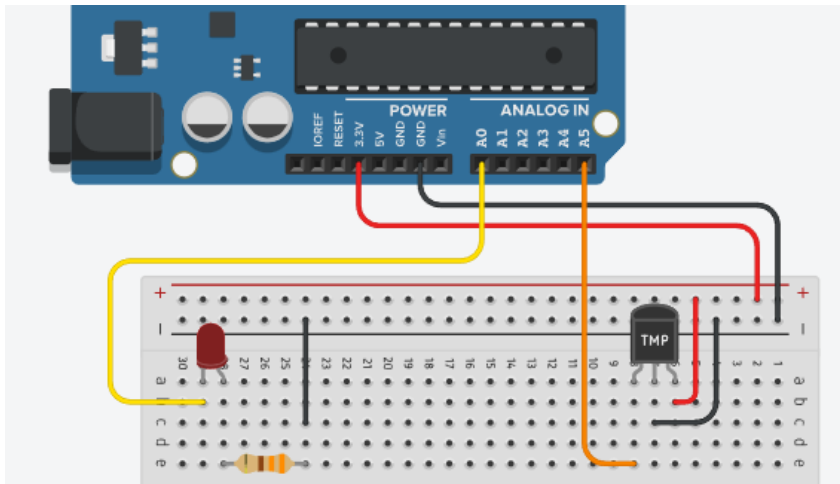


3.1.2 Teplotní čidlo

Používáme teplotní čidlo LMT85LPG(TO-92S) pro 3.3V



3.1.3 Celkový pohled na připojení



3.2 Nastavení

Konfigurace se provádí přes konfigurační stránku(menuconfig — > Temperature controller). Pokud není zadána žádná konfigurace, program se pokusí převzít hodnotu, která byla uložena v nvs. Pokud tato hodnota není přítomna, bude program pracovat s omezeními (např. nebude odesílat informace do mqtt brokeru).

```
(Top) → Temperature controller
(TP-Link_E854) Wifi SSID
(123456789) Wifi password
(20) Temperature threshold
(3) HYSTERESIS
(mqtt://broker.emqx.io:1883) MQTT Broker url
```

3.2.1 Nastavení debug

V souboru main.c můžete nastavit debug = 1 pro zobrazení ladicích informací.

3.2.2 Nastavení wifi

Zadejte SSID a heslo sítě, ke které se chcete připojit. Upozorňujeme však, že se nelze připojit k sítím 5G.

3.2.3 Nastavení temperature threshold a hysteresis

Můžete zadat teplotní limit pro provedení modulu `threshold.h` a také hysterezi. Program také umožňuje aktualizovat tyto hodnoty pomocí MQTT, a to tak, že se program přihlásí k odběru témat:

- *xsmirn02/temp_threshold*
- *xsmirn02/temp_hysteresis*

Tyto hodnoty lze aktualizovat jednoduše zasláním požadavku obsahujícího pouze novou hodnotu do spojeného tématu.

3.2.4 Nastavení MQTT broker url

Zadejte brokera pro použití MQTT, pokud není broker zadán, program bude pokračovat bez použití MQTT. Je třeba poznamenat, že program pracuje s brokery MQTT pouze prostřednictvím připojení TCP.