

Zadání

Cílem je vytvořit senzor pro měření hladiny vody v nádrži za použití ESP32 a ultrazvukového senzoru. Senzor bude odesílat neměřená data na server (MQTT broker) , kde bude možné hodnoty vyčítat, ukládat.

Senzor je rozšířen o další funkce :

- Měření teploty a vlhkosti ovzduší
- Spínání 230V AC
- Solární dobíjení
- Konfigurace velikosti nádrže

Hardware

Z hlediska fyzické realizace senzoru bylo třeba propojit několik prvků. Senzor vzdálenosti je prostým zvukovým pingem. Pro širší použití je přidán senzor teploty a vlhkosti vzduchu. K senzoru je též možné připojit modul pro spínání dvou zařízení na 230V střídavého napětí. Spínací modul je připojitelný pomocí vodotěsného konektoru. Příkladem použití spínacího modulu může být čerpadlo na zavlažování případně dopouštění nádrže.

Určitá míra vodotěsnosti, která by byla ve skutečném nasazení zapotřebí je demonstrována vodotěsným konektorem Superseal, kterým se připojuje spínací modul a návrhem pouzdra . Vrchní kryt pouzdra se solárním panelem přesahuje vlastní tělo a je mírně zešíkmený, aby do pouzdra nezatékalo. Veškerá kabeláž je vyvedena ze spodní strany a vše je utěsněno gumovými průchodkami a tavným lepidlem. Ultrazvukový senzor je odolný maximálně proti postříkání vodou, ale v aktuálním stavu není připraven na venkovní podmínky. Všechny části pouzdra jsou tisknuty na 3D tiskárně materiálem PLA s rozlišením 0.35mm, bez podpor. Pouzdro pro spínací modul je zakoupeno jako univerzální prototypové pouzdro.

Senzor

- ESP WROOM32
- HC-SR04
- DHT11
- Obousměrný převodník log. napětíových úrovní 3V3/5V
- Nabíječka TP4056
- Step-Up měnič – MT3608
- Solární panel 6V 0.6W
- Li-Po 3,7V 900 mAh
- Zenerova dioda ZY6.8
- odpor 4.7k

Server

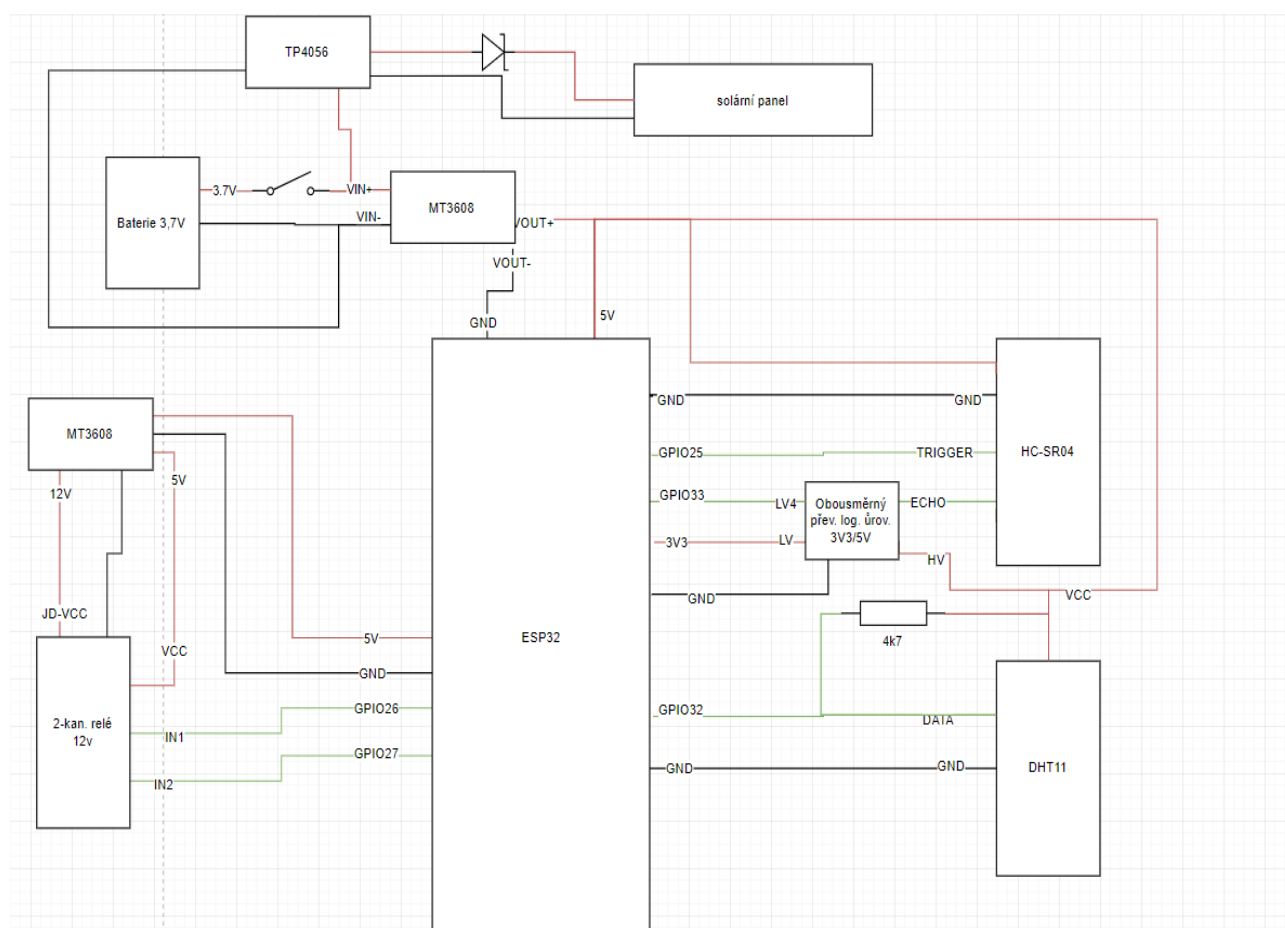
- Raspberry Pi
- Dotykový TFT displej 3.2 palců

Modul spínače 230V

- modul 2-kanálového relé 12V
- Step-Up měnič – MT3608
- Vodotěsná 2-vidlicová zásuvka



Schéma zapojení



Software

Programová část projektu zahrnuje několik částí. Program pro senzor, příprava serveru a vizualizace zasílaných dat a jejich další interpretace.

ESP32

Senzor je programován za pomoci Arduino IDE a knihovny ESP32-Arduino. Celý koncept je navržen k maximalizaci úspory energie a ESP32 po provedení svých úkolů vstupuje do režimu

DEEP SLEEP. Pro rozesílání informací je použito protokolu MQTT a knihovny PubSubClient¹. V aktuálním stavu senzor změří teplotu a vlhkost ovzduší a knihovna DHTesp, která slouží pro komunikaci se senzorem poskytuje funkce na dopočítání pocitové teploty a rosného bodu. Výpočet těchto dodatečných parametrů z teploty a vlhkosti není přesný, protože je ve výpočtu napevno zadán barometrický tlak a senzor pro měření tlaku senzor neobsahuje. Po již zmíněném měření a vykonání ostatních činností se senzor uspí.

Příprava serveru

Na Raspberry Pi, které slouží jako základ serveru, kde poběží MQTT broker a Apache server s PHP a MYSQL databází. Tyto funkce nejsou součástí základního vybavení systému Raspbian, který je v tomto projektu použit. Je nutné je doinstalovat a některé části dodatečně nakonfigurovat. Popis instalace jednotlivých součástí je obsažen v souboru instalace_rpi.txt.

Důležitou součástí je přepis dat z MQTT topiců do MYSQL databáze, aby bylo možné vytvářet statistiky a provádět další výpočty. Problém je řešen pomocí skriptu v Pythonu a je spuštěn při startu systému. Skript v nekonečné smyčce naslouchá na zadaných topicích a data vkládá do databáze.

Vizualizace

Pro vizualizaci dat ze senzoru je použito webového serveru Apache a PHP s použitím knihovny Mosquitto² jako MQTT klienta a MYSQL databáze pro ukládání dat.



Pro vizualizaci je vyhrazeno několik demonstračních HTML dokumentů. Stylistika a formátování je triviální a má pouze demonstrační účel. K urychlení procesu tvorby grafické stránky je použito volně dostupných Bootstrap snippetů³ s drobnými úpravami, aby vyhovovaly tomuto projektu.

Kalibrace

Kalibrační proces je možné spustit z nastavení. Pro vizuální nebo zvukovou signalizaci kroků slouží LED dioda nebo piezoelektrický bzučák připojený na Superseal konektor mezi pin 4(data) a zem na pinu 1. Zaznamenaná-li senzor po probuzení požadavek na kalibraci je to signalizováno 2 pípnutími

1 <https://github.com/knolleary/pubsubclient>

2 <https://github.com/mgdm/Mosquitto-PHP>

3 <https://bootsnipp.com>

bzučáku. Poté má uživatel 10 vteřin na nastavení senzoru pro měření průměru nádrže. Ukončení měření poloměru je signalizováno jedním delším pípnutím. Poté má uživatel dalších 10 vteřin nastavit senzor do základní polohy pro změření výšky prázdné nádrže. Konec kalibrace je doprovázen dvěma pípnutími bzučáku. Změřená data jsou zapsána do FLASH paměti odkud se při každém probuzení vyčítají a používají se v k výpočtu požadovaných hodnot.

Závěr

Senzor se podařilo úspěšně sestavit a zprovoznit. Pouzdro a další součásti vyžadují další testování. Zadaní se podařilo splnit, avšak některé součásti jsou pouze demonstrativního charakteru.

Dalším vhodným rozšířením by byla dynamická konfigurace WiFi nastavení . Například pomocí Bluetooth a Android zařízení, pro která již existují aplikace pro tento způsob konfigurace.

Ultrazvukový senzor vyžaduje ještě několik vývojových iterací, aby byl použitelný i ve venkovním prostředí. Případně je možné ho zaměnit za JSN-SR04T , ale ten je citlivý až od 20 cm.

Ve reálném nasazení by se lépe osvědčil nějaký volně dostupný projekt IoT frameworku, který již obsahuje požadované funkce a nástroje pro správu a vizualizaci.

Zdroje

- [1] https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
- [2] <https://www.hivemq.com/blog/mqtt-essentials-part-4-mqtt-publish-subscribe-unsubscribe>
- [3] <https://www.eclipse.org/paho/clients/python/docs/>