2023 - 2024

|  |
| --- |
| Exploratie van IoT-toepassingen door middel van een Poortopener Project: Ontwerp, Implementatie en Reflectie |
| Ontwerp en Implementatie van een Poortopener Applicatie gebruikmaken van BLYNK |
| Door Nathan Van der Sande |

Inhoud

Inhoud

[Beschrijving van project: 3](#_Toc163988642)

[1. Beschrijving van Gebruikte Hardware 3](#_Toc163988643)

[1.1. Hardware Configuratie 3](#_Toc163988644)

[1.2. Berekening van de Weerstandswaarde voor de BC547-transistor 4](#_Toc163988645)

[1.3. Berekening van de Weerstandswaarde voor de IRL530 MOSFET 4](#_Toc163988646)

[2. Elektrisch Schema: Ontwerp en Verklaring 5](#_Toc163988647)

[3. Beschrijving en Gebruik van de Blynk Applicatie 5](#_Toc163988648)

[4. Probleemoplossing, Lessen en Toekomstige Verbeteringen 5](#_Toc163988649)

Beschrijving van project:  
In het kader van dit project heb ik ervoor gekozen om een applicatie te ontwikkelen die fungeert als poortopener, met de bijkomende functionaliteit om de status van de poorten te monitoren en door te geven. Dit werd gerealiseerd door het gebruik van field-effect transistors (FETs) en bipolaire transistoren die binnen handbereik waren. Het concept behelst het voorzien van drie poorten, elk uitgerust met twee fysieke knoppen en een virtuele knop. Daarnaast wordt de gesloten status van de poorten gedetecteerd met behulp van contactsensoren. Deze functionaliteit wordt uitgevoerd door middel van een ESP32 ontwikkelmodule. Bovendien wordt de ESP ook ingezet voor het meten van de temperatuur en luchtvochtigheid door gebruik te maken van een DHT11 sensor.

# Beschrijving van Gebruikte Hardware

* ESP32: Een programmeerbare microcontroller met ingebouwde Wi-Fi en Bluetooth-functionaliteit, gekozen vanwege zijn veelzijdigheid en connectiviteitsopties.
* IRL530: Een N-kanaal MOSFET-transistor die wordt gebruikt voor het aansturen van externe belastingen zoals motoren of lampen. De IRL530 is gekozen vanwege zijn hoge vermogenscapaciteit en efficiënte schakeling.
* BC547: Een NPN-bipolaire transistor die fungeert als een signaalversterker of schakelaar. De BC547 wordt gebruikt voor signaalversterking in het schakelcircuit van de poorten.
* MC-38 Magnetic Contact Switch Sensor: Een magnetische contactsensor die wordt gebruikt voor het detecteren van de open of gesloten status van de poorten. De sensor bestaat uit een magnetisch reed-contact dat wordt geactiveerd door de nabijheid van een magneet.
* DHT11: Een digitale temperatuur- en luchtvochtigheidssensor die wordt gebruikt voor het meten van omgevingscondities. De DHT11 communiceert via een enkele datalijn en is eenvoudig te integreren in het ESP32-platform.
* Jumperwires: Flexibele elektrische draden met connectoren aan beide uiteinden, gebruikt voor het tot stand brengen van elektrische verbindingen tussen componenten.
* Lusterklemmen: Elektrische klemmen die worden gebruikt om draden veilig en stevig met elkaar te verbinden. Lusterklemmen worden gebruikt voor het aansluiten van draden op de ESP32 en andere componenten, en bieden een betrouwbare verbinding.

Methodologie en Procesbeschrijving

## Hardware Configuratie

De initiële fase van het hardware-assemblageproces omvatte het ophangen van alle draden en sensoren. De bestaande fysieke schakelaars werden voorzien van jumperwires die rechtstreeks werden verbonden met de poortopener. Bovendien werden parallel aan deze draden twee extra jumperwires aangesloten om extra functionaliteit te bieden (zie bijlage 1).

De plaatsing van de contactsensoren werd strategisch bepaald om een signaal te activeren wanneer de poort gesloten is. Het belang van deze positie lag in het feit dat het de functionaliteit van de poort niet mocht belemmeren. Voor elke poort werd een unieke positie bepaald (zie bijlage 2.1|2.2|2.3). Om de juiste positionering te bereiken, werden de contactsensoren verlengd met behulp van jumperwires.

Alle bekabeling werd geleid naar een breadboard waarop de ESP32 microcontroller was gemonteerd. Dit centralisatiepunt vergemakkelijkte het beheer en de aansluiting van alle componenten (zie   
bijlage 3)

## Berekening van de Weerstandswaarde voor de BC547-transistor

Voor de weerstandswaarde tussen de ESP32 en de basis van de BC547-transistor, kun je deze berekenen met behulp van de volgende formule:

* R=​(VGPIO​−VBE)/IB​​

Waar:

* VGPIO​ de uitgangsspanning is van de ESP32 GPIO-pin (in dit geval 3,3V).
* VBE​ de basis-emitterspanning van de transistor is (ongeveer 0,7V voor een siliciumtransistor zoals de BC547).
* IB​ de gewenste basisstroom is (0,30mA in dit geval).

Als we aannemen dat VGPIO​=3,3V, VBE​=0,7V, en IB​=0,30mA, hebben we:

* R=(3,3V−0,7V)​/0,30mA
* R≈0,30mA​/2,6V
* ≈8666,67Ω

Je kunt een standaardweerstandswaarde gebruiken die dicht bij deze berekende waarde ligt, zoals 8,2kΩ of 10kΩ.

## Berekening van de Weerstandswaarde voor de IRL530 MOSFET

Om de juiste weerstandswaarde te bepalen voor de IRL530 MOSFET, maken we gebruik van de volgende formule:

* R= VGPIO/​​ IGATE​

Waar:

* VGPIO​ de uitgangsspanning is van de ESP32 GPIO-pin (3,3V).
* IGATE​ de stroom is die nodig is om de gate van de MOSFET te activeren (een kleine waarde typisch in het bereik van microampères tot milliampères).

De gatestroom (IGATE​) voor MOSFETs zoals de IRL530 is meestal zeer klein en kan vaak worden genegeerd voor digitale schakeltoepassingen. Echter, voor een veilige marge, laten we IGATE​ gelijk zijn aan ongeveer 10 keer de lekstroom, die typisch in het nanoamperebereik ligt.

Laten we IGATE​=10×100nA=1μA=0,000001A aannemen.

Door de waarden in te vullen, krijgen we:

* R=3,3V/0,000001A​
* R=3,3×10^6Ω
* R=3,3MΩ

Echter, zo'n hoge weerstandswaarde kan het schakelen vertragen. Het is praktischer om een weerstand met een lagere waarde te gebruiken, typisch in het bereik van kilo-ohm.

Bijvoorbeeld, een 10kΩ weerstand zou goed werken:

* R=10kΩ

Dit zal voldoende stroombeperking bieden terwijl snelle schakeltijden worden gewaarborgd.

# Elektrisch Schema: Ontwerp en Verklaring

(zie bijlage 4)

Hier volgt een beschrijving van het electrisch schema voor het systeem:

Drie transistoren, respectievelijk Transistor1, Transistor2 en Transistor3, zijn geconfigureerd als schakelaars om de bediening van verschillende poorten, genaamd Poort1, Poort2 en Poort3, te regelen. Transistor1 is verbonden met GPIO pin 2 van de microcontroller, samen met de ground- en poort1-aansluitingen. Transistor2 is aangesloten op GPIO pin 4, samen met de ground- en poort2-aansluitingen. Transistor3 is verbonden met GPIO pin 5, samen met de ground- en poort3-aansluitingen.

Daarnaast zijn er drie contactsensoren, respectievelijk Switch1, Switch2 en Switch3, die dienen als externe bedieningselementen voor het systeem. Switch1 is verbonden met GPIO pin 18 en ground, Switch2 met GPIO pin 19 en ground, en Switch3 met GPIO pin 21 en ground.

Tenslotte is er een DHT-sensor geïntegreerd in het systeem om omgevingsgegevens te meten. De sensor is aangesloten op de 3,3V-voedingsbron van de microcontroller, GPIO pin 22 voor gegevensuitwisseling en ground voor de referentieaarding.

# Beschrijving en Gebruik van de Blynk Applicatie

Voor het aansturen van de poort wordt gebruik gemaakt van de BLYNK app. Deze app maakt gebruik van een legacy versie die gehost wordt op een zelf gehouden server. Het gebruik van deze optie kan echter beperkt zijn voor sommige gebruikers. Ik raad daarom aan om mijn home-assistant repository te bekijken (momenteel in ontwikkeling) voor een zelf gehoste alternatief.

In bijlage 5 is een screenshot van de app lay-out te vinden.

Een LED in de app licht op wanneer de poort open is, waardoor op afstand kan worden gecontroleerd of de poort open staat. Twee meters worden gebruikt om de temperatuur en luchtvochtigheid visueel weer te geven. Daarnaast zijn er drie knoppen beschikbaar als schakelaar voor de poorten. Het is echter belangrijk op te merken dat deze knoppen minstens 500 milliseconden ingedrukt moeten worden om ervoor te zorgen dat er een volledig contact tot stand komt.

# Probleemoplossing, Lessen en Toekomstige Verbeteringen

Een van de belangrijkste uitdagingen die zich voordeed, was het verlengen van de draden. Uiteindelijk werd besloten om lusterklemmen te gebruiken. Deze oplossing bleek echter niet ideaal vanwege de dunheid van de jumperwire. Met name in omgevingen waar water of wind aanwezig is, kan deze optie problematisch zijn.

Het gebruik van een breadboard bleek onoverzichtelijk. Voor toekomstige projecten zou de overstap naar een printplaat (PCB) worden overwogen. Het nadeel van een PCB is echter dat het mogelijk minder flexibiliteit biedt voor uitbreiding. Bovendien kan het vervangen van poorten de functionaliteit van de transistoren verstoren.

Bij het plaatsen van de contactsensoren is het van belang ervoor te zorgen dat ze de werking van de poort niet hinderen. Als de poort niet volledig sluit, moet de positie van de sensor worden aangepast. Daarnaast is het essentieel om een geschikte afstand tussen de sensoren te handhaven om valse negatieve waarden te voorkomen.

Verder bleek de afhankelijkheid van de Blynk-app van een derde partij een beperking. Daarom werd overgestapt naar een homeassistant-applicatie, waarbij gebruik wordt gemaakt van YAML-configuratiebestanden. Deze alternatieve aanpak maakt zelfhosting van het project mogelijk, waardoor de autonomie wordt vergroot en de afhankelijkheid van externe platforms wordt verminderd. Details van deze projectaanpassing worden beschreven in een apart GitHub-repository.

|  |
| --- |
| contact  Nathan Van der Sande | student  nathanvandersande@gmail.be  Volg ons  www.thomasmore.be  fb.com/ThomasMoreBE  #WeAreMore |
|  |