Jochem van Kranenburg, 1741991, 25-5-2019

Jochem.vankranenburg@student.hu.nl

plan van aanpak

Voor IPASS

# Hardware

In de basis ga ik gebruik maken van een Arduino Due, TEA5767 (<https://bit.ly/2vt4BRC>), 3W 4 Ohm speakers en PAM8403 (<https://bit.ly/2PfxWKx>) om een FM-radio te maken. Dit deel omvat dan ook de Library. Daarnaast ga ik, indien de tijd het toelaat, de radio uitbreiden met een 0,96 inch SSD1306 (<https://bit.ly/2gYB8r0>) (of later 1,77inch ST7735 Display (<https://bit.ly/2LbwXum>)), KY040 Rotary Encoder, 24LC256I2C EEPROM IC (<https://bit.ly/2YKlYtH>), ST1089 IR receiver (<https://bit.ly/2DCbBA4>), Bluetooth Audio Module (gesloopt uit USB dongle), 18650 met TP4056 (<https://bit.ly/2fv19Od>) in combinatie met zonnepaneel (5V, 250mA) en step up converter en daarnaast nog een 3.5mm audio jack output of input voor externe speaker.

Wat betreft demo projecten is [dit](https://bit.ly/2vtk1Fo) de meest uitgebreide.

# Library

De library is bedoeld voor de TEA5767. Dit is een FM-ontvangstmodule die wordt aangestuurd via de I2C bus. De module bestaat uit vijf 8-bits registers die de volgende zaken besturen:

* Wel of geen output
* Wel of geen zoekmode
* Mono of stereo output
* Links wel of geen output
* Rechts wel of geen output
* Standby mode
* Noise cancelling
* Een aantal synthesizer opties.

Wat betreft methoden en inhoud van klassen vermoed ik één klasse te hebben met een aantal private attributen betreffende stereo/mono, frequentieband beperkingen en huidige frequentie. Verder verwacht ik voor alle hierboven genoemde instellingen een functie te hebben waarmee deze instelling kan worden uitgezet of toegepast. Daarnaast zullen er functies beschikbaar moeten zijn die de statussen van deze instellingen op kunnen vragen.

Hier verwacht ik dan ook dat de complexiteit van de library zit; tamelijk groot en niet altijd even triviaal. De datasheet zal een zeer groot aantal vragen moeten beantwoorden. Daarnaast zou ik het project graag kunnen overhevelen naar een ESP32 (of ander bord met WiFi capabiliteit) voor eventuele extra wifi-functionaliteit (en het totaalplaatje) met webserver op de tweede core. Daarom zou ik het, indien de tijd het toelaat, leuk vinden de library naast dat het compatibel is met hwlib/bmptk ook compatibel te maken met Arduino.

Indien het zo uit komt dat ik ook gebruik maak van de KY-040 Rotary Encoder en 24LC256, zal ik voor deze componenten zelf ook een library-deel proberen te schrijven.

# Applicatie

Als minimaal eindresultaat zie ik een draagbare radio voor ogen die in eerste instantie enkel FM-radio af kan spelen. Deze radio bevat ook een OLED en rotary encoder (waarvoor de interfacing zelf geschreven is). Deze radio heeft indien de tijd het toelaat ook een bluetoothmodule voor bluetoothaudio, potentiometer om het volume aan te passen, een scherm (eerst een SSD1306 OLED, later eventueel ST7735) om instellingen op weer te geven en een zonnepaneel met micro-usb aansluiting om de accu op te laden. Daarnaast zou er een EEPROM IC voor opslaan van zenders en instellingen, website voor besturing en afstandsbediening kunnen zijn indien er nog weer tijd over is.

Hierbij wordt gebruik gemaakt van hwlib en bmptk. In het geval van lezen van infraroodsignalen en hosten wordt er gebruik gemaakt van externe libraries.

# Risicobeheersing

Het project zal in de volgende stappen worden doorlopen:

1. Na afloop van het lezen van de datasheet het met behulp van de datasheet schrijven van de library op de manier waarvan wordt vermoed dat deze werkt.
2. Library fine-tunen zodat de minimale functionaliteit (wisselen tussen en zoeken van zenders) betrouwbaar en goed werkt.
3. Schakeling maken voor de speakers met versterker zodat de library voor de TEA5767 eenvoudig en goed kan worden getest (2x3W 4Ohm speakers en PAM8403).
4. Overige functionaliteiten in de library implementeren (Stereo/Mono, Links/Rechts, Noise Cancelling).
5. Library debuggen, testen, edge cases afvangen en van documentatie voorzien.
6. SSD1306 OLED met rotary encoder toevoegen zodat de radio kan worden gedemonstreerd en instellingen aan kunnen worden gepast.
7. Totale werking testen.

Het minimale eindresultaat (Arduino Due met PAM8403, Speakers, TEA5767, Rotary Encoder en OLED) zou hier behaald moeten zijn. Voordat ik verder ga, zal ik eerst alles dubbel en dwars langslopen om te kijken of alles aan de eisen voldoet, kwalitatief goed is en er geen betere oplossing voor opgedane problemen zijn. Indien dat het geval is, ga ik verder met het volgende:

1. 24LC256 toevoegen zodat zenders en instellingen tussen stroomonderbrekingen door aanwezig kunnen blijven en er niet telkens opnieuw gezocht hoeft te worden naar zenders.
2. Bluetoothmodule (enkel hardware-matig) toevoegen. Er kan tussen audio-inputs worden gewisseld door middel van een transistor, mosfet of andere schakelaar.
3. De schakeling afmaken met TP4056, 18650 en zonnepaneel. Indien alles werkt wordt alles gesoldeerd op één of meerdere protoboards.

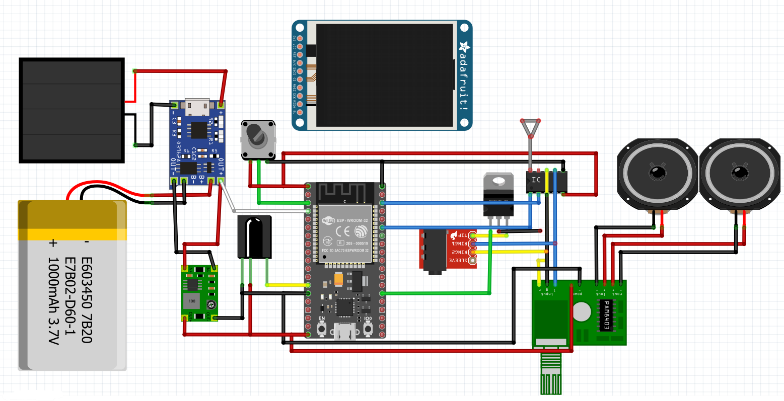
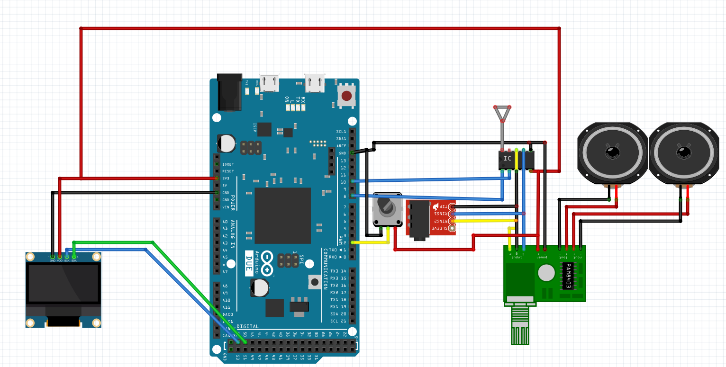
Het maximale eindresultaat voor de Arduino Due (Arduino Due met PAM8403, Speakers, TEA5767, Rotary Encoder, OLED, 24LC256, Bleutoothmodule, TP4056, 18650 en Zonnepaneel) zal hier behaald zijn. Afhankelijk van hoeveel tijd er nog over is, zal er gekozen worden tussen dit:

1. Rotary encoder en 24LC256 implementaties (tot nu toe waarschijnlijk geïmplementeerd als klassen) omschrijven naar libraries.
2. Library(ies) compatibel maken met Arduino.
3. Code omschrijven naar Arduino-Code zodat de ST7735, Webserver en IR-ontvanger gebruikt kunnen worden. Op het protoboard is met de female header-pins rekening gehouden met de ESP32. De Arduino Due en ESP32 kunnen zo eenvoudig worden afgewisseld tijdens de demonstratie.

Of dit (vrijwel zeker niet haalbaar):

1. Met behulp van datasheet schrijven van library voor BMP180 (<https://bit.ly/2sj53AK>).
2. Library testen, uitbreiden, testen en documenteren.

Aangezien er heel stapsgewijs wordt gewerkt en alles voortbouwt op het eerder gedane werk, zal er hoe dan ook een eindresultaat zijn wat bestaat uit een radio, FM-ontvanger en speaker. Aangezien vorige projecten ook met ESP’s, displays, rotary encoders en accu’s is gewerkt, verwacht ik daar geen problemen mee. Ik kan me wel verkeken hebben op de library. Indien de library lastiger is dan ik vermoed, zal de library in de basis betrouwbaar maar niet uitgebreid zijn. Ik denk dan ook dat daar de prioriteit ligt. Al met al vermoed ik dat er met het hierboven beschreven stappenplan niet veel fout kan gaan. Indien er tijd over is, en de kwaliteit van het geleverde werk goed is, kan ik op tal van manieren verder uitbreiden. Indien het maximale eindresultaat wordt bereikt, hoop ik alle onderdelen op een protoboard te kunnen solderen. De ESP32 zou dan op female header-pins geklikt kunnen worden zodat met zowel de ESP32 als de Arduino Due dezelfde hardware kan worden gebruikt tijdens de demonstratie.



Maximaal eindresultaat

Minimaal resultaat