Thomas More campus De Nayer Jan De Nayerlaan 5 - 2860 Sint-Katelijne-Waver

**Project 2EE**

Schooljaar: 2018 – 2019

Studierichting Elektronica ICT

**Recordio**

Aäron Willemsens & Ward Lefevre

# Voorwoord

Dit verslag over Recordio is gemaakt voor het vak Practice Enterprise 2.

Ik zou als eerste de personen willen bedanken die me geholpen hebben bij dit project. Namelijk Mr. Dams voor zijn hulp bij de initiële start van het project en voor ons in de goede richting te sturen. Ook willen we Mr. Van Bauwel bedanken voor het helpen bij enkele problemen die we onderweg aantroffen.

Auteurs: Aäron Willemsens

R0714818

[Aaron.willemsens@gmail.com](mailto:Aaron.willemsens@gmail.com)

Ward Lefevre

R0626875

[wardlefevre97@gmail.com](mailto:wardlefevre97@gmail.com)

# Inhoudstafel

[Voorwoord](#_heading=h.gjdgxs) **1**

[Inhoudstafel](#_heading=h.30j0zll) **2**

[Projectvoorstel](#_heading=h.1fob9te) **4**

[Teamleden](#_heading=h.3znysh7) 4

[Titel project](#_heading=h.2et92p0) 4

[Korte omschrijving van het project](#_heading=h.tyjcwt) **4**

[Blokschema](#_heading=h.1t3h5sf) 4

[Welk probleem lost deze innovatie op?](#_heading=h.2s8eyo1) 5

[Wat is de behoefte die erachter schuilgaat?](#_heading=h.17dp8vu) 5

[Wie zijn de gebruikers?](#_heading=h.3rdcrjn) 5

[Wat zijn de trends & ontwikkelingen binnen dit vakgebied?](#_heading=h.26in1rg) 5

[Plan van aanpak](#_heading=h.lnxbz9) **6**

[Doel](#_heading=h.35nkun2) 6

[Ontwerpspecificaties](#_heading=h.1ksv4uv) 6

[Gebruik](#_heading=h.44sinio) 6

[Hoe gaan we te werk? (GANTTCHART)](#_heading=h.2jxsxqh) 6

[Blokschema’s](#_heading=h.z337ya) **7**

[Hardware](#_heading=h.3j2qqm3) **8**

[STM32F746g-disco board](#_heading=h.59xhl9v8i65l) 8

[ADP1712](#_heading=h.dhg43q98fuk2) 8

[TPA6021A4](#_heading=h.k5hz3acgejkb) 8

[RK097](#_heading=h.4x9cz6f6j2c9) 8

[Keystone Semi T3B DAB/DAB+/FM/RDS module](#_heading=h.83lj7w181gao) 8

[Schema’s](#_heading=h.wd1syug4l9hg) 10

[Software](#_heading=h.49x2ik5) **12**

[Interface](#_heading=h.vsp4l0afr4jp) 12

[Embedded Wizard](#_heading=h.h6y5qvv3uain) 12

[CubeMX](#_heading=h.mkahyzn74rdb) 12

[TouchGFX](#_heading=h.bs45nqom5gwk) 13

[STM32CubeProgrammer](#_heading=h.2ufu0zcimllx) 14

[Klok](#_heading=h.gnjhx0v0c2k5) 14

[Applicatie Architectuur](#_heading=h.lvjurcf8f50o) 14

[Klok](#_heading=h.to66iavulno0) 14

[Module aanspreken](#_heading=h.x0w38cprzofe) 18

[Command format](#_heading=h.hkciujiu1fdf) 18

[Flowcharts](#_heading=h.7vy7kpeymmxx) 19

[Problemen en hun oplossingen](#_heading=h.ihv636) **21**

[PCB pin headers te groot](#_heading=h.32hioqz) 21

[Voedingsspanningen](#_heading=h.1hmsyys) 21

[Te veel stroom](#_heading=h.41mghml) 21

[Pinheaders verkeerd op PCB](#_heading=h.4obwyou7qkvu) 21

[Grounds niet verbonden](#_heading=h.2grqrue) 21

[Datasheet verkeerd geïnterpreteerd](#_heading=h.vx1227) 21

[Geen reset](#_heading=h.l6gyhhm0aukq) 22

[Geen opslag](#_heading=h.xrj3flkkekhb) 22

[Geen communicatie met DAB module](#_heading=h.qpuucm43ammm) 22

[Logboek](#_heading=h.3fwokq0) **23**

[Aäron](#_heading=h.lq5slhvnl7wg) 23

[Ward](#_heading=h.6pfm3h6cro8m) 25

[Besluit & Reflectie](#_heading=h.1v1yuxt) **26**

[**Bijlagen**](#_heading=h.gamvb13vywhe) **27**

[T3B module datasheets](#_heading=h.28h4qwu) 27

[DAB+ shield componenten datasheets](#_heading=h.67iq4ge8w2ns) 27

[DAB+ shield PCB](#_heading=h.2xa3olkvum37) 27

[STM32F746g board](#_heading=h.jwg710kj6gcv) 27

# Projectvoorstel

## Teamleden

Ward Lefevre

Aäron Willemsens

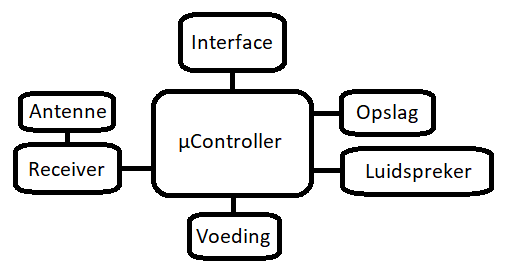
## Titel project

Recordio

# Korte omschrijving van het project

Recordio is een wekkerradio die zijn signaal opvangt van het DAB+ netwerk. De wekker bestaat uit het STM32F7 board met lcd dat gebruikt wordt als interface. In combinatie met een DAB+ module, de T3B module van KeyStone Semi, die verbonden is met de STM32F7 via een shield dat we zelf maakten. De wekkerradio zal muziek kunnen spelen, een klok tonen en een alarmfunctie hebben.

## Blokschema

Budgetraming (schatting)

* Microcontroller STM32F7: 0 euro (al in bezit)
* T3B module: 44.84 euro
* Antenne: 17.75 euro
* Shield: 40 euro
* Behuizing: 15 euro
* Voeding: 0 euro (al in bezit)

+\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

117.59 euro

## Welk probleem lost deze innovatie op?

We lossen geen probleem op, maar we vernieuwen een huidig concept met nieuwe technologie. We proberen het de gebruiker makkelijker te maken en er voor te zorgen dat ze kunnen kiezen wanneer ze naar een nieuwsfragment of een verkeersmelding beluisteren.

## Wat is de behoefte die erachter schuilgaat?

Men kan kiezen wat je opneemt en later afspeelt om bijvoorbeeld je wekker te zetten voor kwart na 7 maar dan opstaan met het nieuws van 7 uur om toch op de hoogte te blijven van het nieuws en eventueel een verkeersituatie.

## Wie zijn de gebruikers?

Elke persoon dat een wekker gebruikt kan dit product gebruiken. Het zal makkelijker voor hen zijn omdat ze meer keuze hebben wat ze willen beluisteren als ze opstaan. We raden het dus aan van personen boven de 12 jaar.

## Wat zijn de trends & ontwikkelingen binnen dit vakgebied?

Wekkers bestaan al reeds en daar zijn verschillende versie van. Zo kan je er eentje kopen met gewone radiofunctie of eentje met bluetooth. Je kan tegenwoordig je wekker ook al zetten met je telefoon. In auto’s wordt DAB+ al reeds gebruikt en in sommige radio’s komt dit ook al voor. Meerdere radiostations in België schakelen ook over op DAB+ of geven een extra optie voor DAB+. Wij voegen deze twee technologieën samen in een DAB+ radiowekker.

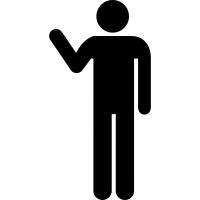
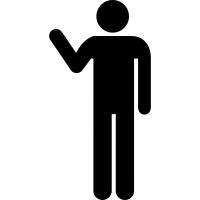
# Plan van aanpak

## Doel

Het uiteindelijke doel is om een werkend product te hebben. Hieronder verstaan we dat je de wekker kan instellen, bijvoorbeeld het uur en een alarm. Je kan ook van radiozender veranderen en het geluid aanpassen en het geluid afspelen via de speakers maar ook via een aux aansluiting. Je kan een opname maken van een nieuws- of verkeersuitzending.

## Ontwerpspecificaties

Een interface waarmee de klant interactie kan hebben met het product en een uur en alarm instellen. Hij kan de radiozenders veranderen via een aantal knoppen. Het STM32F7 board kan data halen uit de T3B module en kan deze afspelen op de speakers. Het geluid kan aangepast worden via een potentiometer maar ook via een knop op de interface.



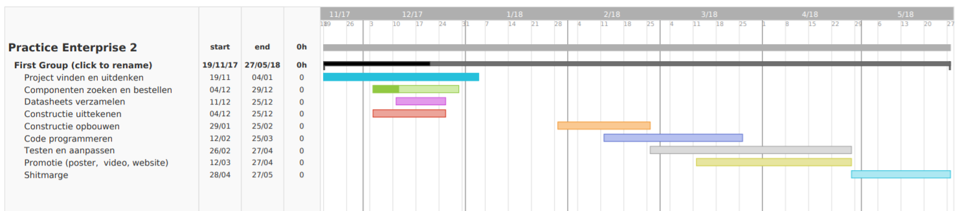




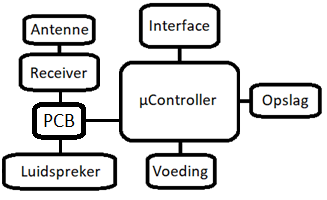
## Gebruik

Door middel van een touchscreen interface kan men alles instellen en veranderen. Via een potentiometer kan men het geluid regelen en via een aux aansluiting kan men de audio ook op een andere manier afspelen.

## Hoe gaan we te werk? (GANTTCHART)



# Blokschema’s





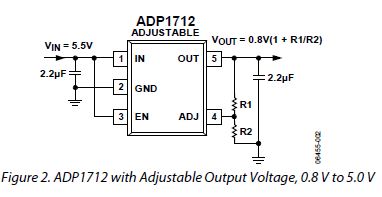
De microcontroller, STM32F7 bestuurd het gehele systeem. Het maakt gebruik van een 12V voeding die wordt omgezet naar 5V. De microcontroller bevindt zich om het STM32F746 board, op dit bord sluiten we dan onze shield aan. Op de shield wordt dan de DAB+ module geplaatst die op zijn beurt verbonden is met een antenne waarvan deze dan zijn signaal kan halen. OP de shield staat ook een amplifier die verbonden is met 4 speakers om de audio af te spelen.

Via het LCD scherm dat op het STM32F7 board staat kan de gebruiker het uur instellen, van radiozender veranderen, het volume veranderen en kiezen wanneer en wat hij wil opnemen.

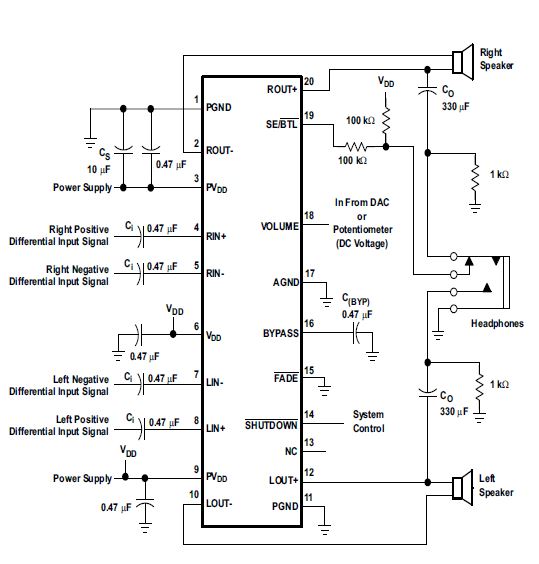
# Hardware

## 

## STM32F746g-disco board

Het STM32F7 board hebben we gekozen voor dit project omdat we dit reeds aangekocht hadden via het school voor het vak Embedded. We hadden het dus al in bezit en het paste perfect in onze toepassing. Het STM32F7 board heeft al een LCD ingebouwd dat we kunnen gebruiken voor onze interface en weergave van onze klok. Het board heeft ook een groot aantal pinnen die we kunnen gebruiken als input en output. We gebruiken de I/O pinnen om 3.3V en 5V te halen voor onze shield. Via deze pinnen kunnen we ook de communicatie voor de T3B module voorzien, dit doen we met UART met een baudrate van 57600.

## ADP1712

ADP1712 heeft verschillende mogelijkheden om gebruikt te worden. Zo kan men de ADP1712 gebruiken als fixed-output met een soft-start capacitor, maar men kan hem ook gebruiken als output voltage tracking. In ons geval gebruiken we hem als adjustable output voltage, we maken zo van 5v 1.2v die we nodig hebben voor onze T3B module.

## TPA6021A4

TPA6021 is een audio amplifier die we gebruiken om onze geluidsignalen te versterken en het volume ervan aan te passen. Met deze versterker kunnen we ook 4 speakers van 8 ohm sturen. Met behulp van een potentiometer kunnen we het geluid instellen op de TPA6021A4.

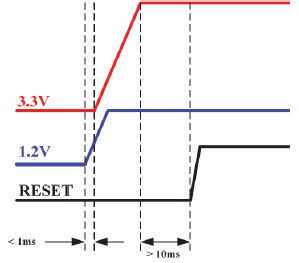
## 

## RK097

RK097 is een potentiometer die we gebruiken om het volume aan te passen via de TPA6021.

## 

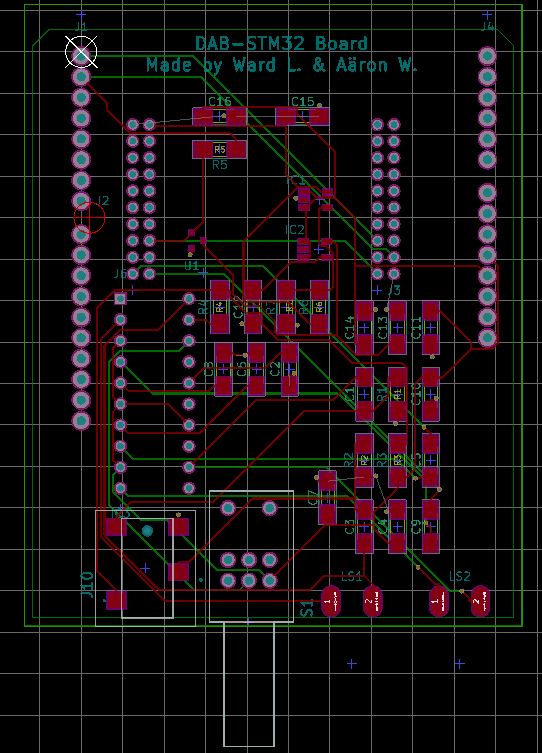
## Keystone Semi T3B DAB/DAB+/FM/RDS module

De DAB+ module die we gebruiken is de T3B module van Keystone Semi, deze module kan ook DAB, FM en AM signalen ontvangen. De module is redelijk compact en kan dus makkelijk in een systeem toegevoegd worden. De module heeft wel enkele speciale voorwaarden, zo moet je bijvoorbeeld 1.2v en 3.3v aanleggen en wordt er aangeraden dat de 1.2v minder dan een milliseconde eerder wordt aangelegd dan de 3.3v. Er moet ook een reset signaal worden aangelegd wanneer de module opstart, dit signaal moet ongeveer een tiental milliseconden later aangelegd worden dan de voedingssignalen. De module heeft ook een antenne nodig om zijn signaal te ontvangen, deze antenne moet de juiste lengte hebben anders kunnen we het signaal niet ontvangen. 

## 

## 

## Schema’s

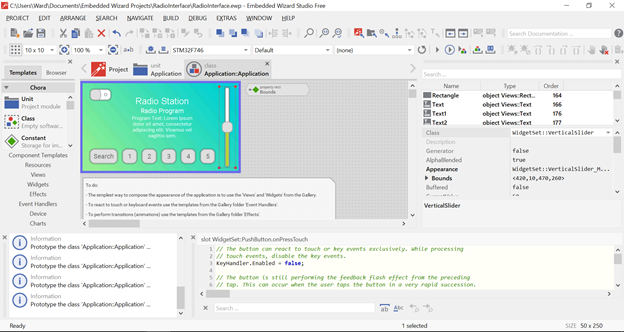


# Software

## Interface

#### Embedded Wizard

De GUI van de radio moet eenvoudig in gebruik zijn. Met slechts enkele tikken op het scherm moet de gebruiker de radio aanzetten, van kanaal veranderen en het geluid kunnen aanpassen. Ook moet er een klok aanwezig zijn, die als screensaver over het scherm beweegt om inbranding van de LCD te voorkomen. Om dergelijke dynamische GUI te maken kunnen we gebruik maken van Embedded Wizard, een populaire IDE van TARA Systems’. Het is een WYSIWYG (What You See Is What You Get) Editor, waar zelfs niet visuele elementen zoals variabelen en klassen worden voorgesteld als afzonderlijke pictogrammen. De workflow van Embedded Wizard is daardoor niet zo eenvoudig, en vergt dus enige tijd om onder de knie te krijgen.

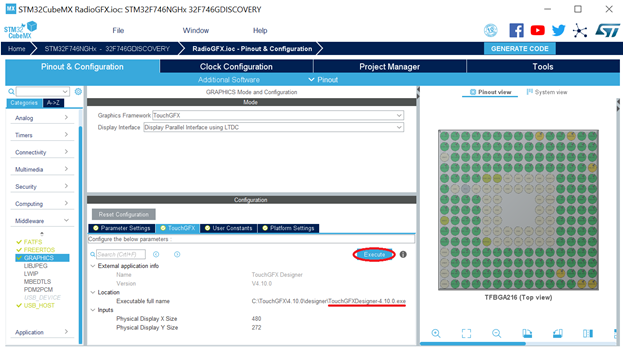


#### CubeMX

STM32CubeMX van ST is de SDK bij uitstek om de hardware van de STM32-microcontrollers te configureren. Embedded Wizard bevat echter geen ready-to-use environment dat steunt op CubeMX. Dit betekent dat het Embedded Wizard project manueel geïntegreerd moet worden in het CubeMX project, wat niet eenvoudig is.

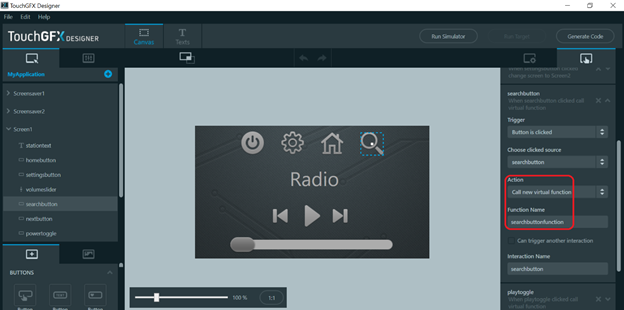
#### TouchGFX

CubeMX kan wel gebruik maken van de addon TouchGFX. TouchGFX is een tool vergelijkbaar met Embedded Wizard, waarbij je met ‘Drag & Drop’ een interface kan ontwerpen. Met een correct ingesteld CubeMX project kan de TouchGFX applicatie in CubeMX geopent worden.



Nu de hardware correct geconfigureerd is en we een GUI kunnen ontwerpen met TouchGFX, laten we het Embedded Wizard project achterwege.

De interface kunnen we vormgeven met verscheidene widgets: buttons, toggles, sliders, images, text, etc. Aan deze widgets kunnen we een virtual function toewijzen, die we in een editor kunnen programmeren.



Na de code generatie van TouchGFX en CubeMX kunnen we de interface functionaliteit geven. Dit doen we met de code editor IAR Embedded Workbench (EWARM). Er moeten nog ontbrekende BSP files toegevoegd worden, en er zijn enkele aanpassingen nodig in het linkerscript om het project werkent te krijgen.

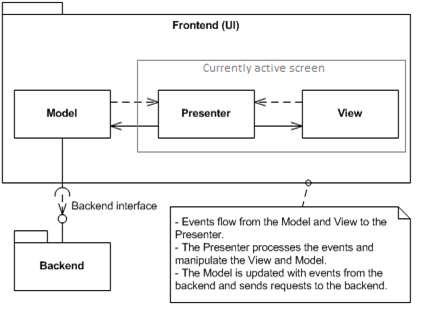
#### STM32CubeProgrammer

Om de binary (gegenereerd door EWARM) naar de controller te downloaden, gebruiken we STM32CubeProgrammer. Indien we hiervoor EWARM gebruiken worden afbeeldingen niet correct geflashed, en krijgen we verstoorde afbeeldingen.

## Klok

#### Applicatie Architectuur

Om de werking van de klok uit te leggen, moeten we eerst de structuur van een TouchGFX applicatie toelichten.



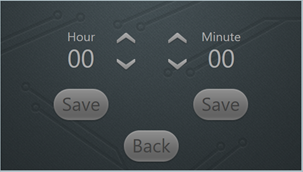
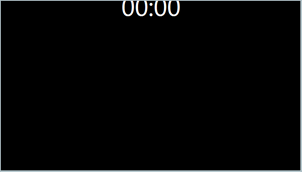
Elk scherm van onze UI heeft een View en een Presenter. Wanneer de gebruiker in de UI van scherm veranderd, worden de View en Presenter van het nieuwe scherm geallocceerd en de View en Presenter van het vorige scherm gedealloceerd . De View en Presenter kunnen dus niet gebruikt worden voor data dat over meerdere schermen beschikbaar moet zijn. Hiervoor gebruiken we de Model.

De Model bevat toestandsinformatie over de UI. Er is slechts één Model voor heel het project.

De View bevat de widgets die we in TouchGFX ontworpen hebben.

De Presenter linkt de Model en View aan elkaar. Het ontvangt backend events van de Model en UI events van de View. Met deze data kan de Presenter vervolgens beslissen welke acties er ondernomen moeten worden.

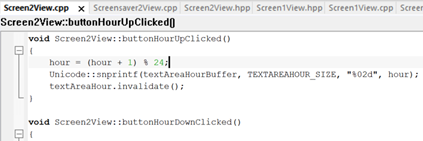
#### Klok

In TouchGFX hebben we een Scherm aangemaakt voor de klokinstellingen (Screen2) en een scherm voor de klok zelf. (Screensaver1)

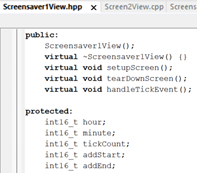
In Screen2View.hpp declareren we onze functies die we in TouchGFX hebben aangemaakt als ‘virtual functions’.



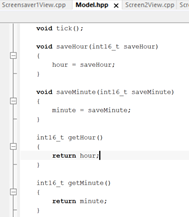
In Screen2View.ccp kunnen we ze vervolgens oproepen en hen functionaliteit geven.

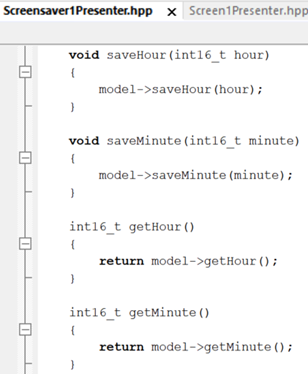
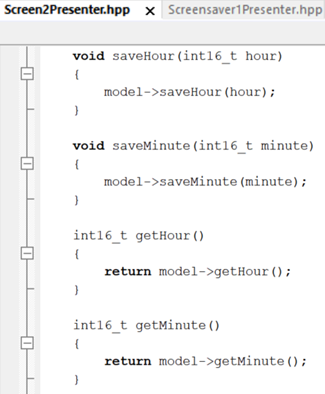


Hetzelfde doen we voor de screensaver.

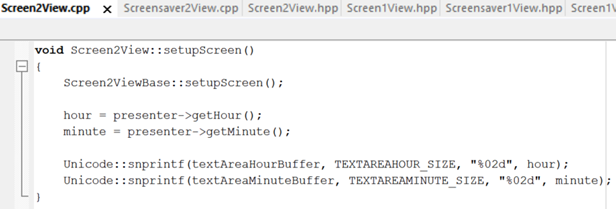


De Model moet de informatie van de klok bijhouden en kunnen returnen.

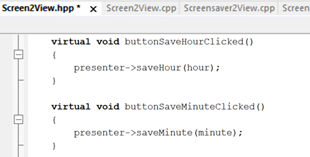


Om de data van het Model te gebruiken in de View moet de Presenter functies voorzien voor het laden en opslaan van de data. Idem voor de screensaver.

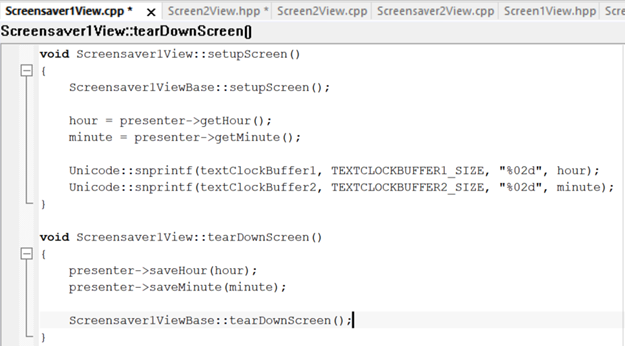
Nu kunnen we in screen2 (klokinstellingen) de huidige tijd weergeven met behulp van de Presenter.



En aangepaste waardes kunnen terug aan de model doorgegeven worden.



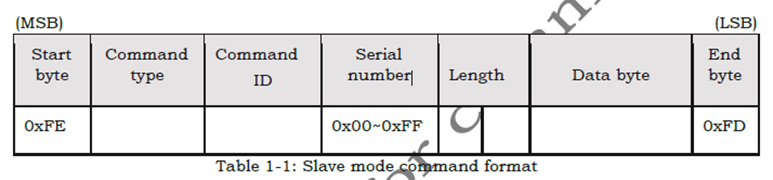
Ook in de screensaver moeten de waardes vanuit de Model, via de Presenter, naar de View geladen worden. Bij het verlaten van de screensaver moeten de nieuwe waardes in de Model geüpdatet worden. (De screensaver blijft bijvoorbeeld een half uur draaien, wanneer de gebruiker de screensaver wil verlaten moet deze nieuwe tijd in de Model terechtkomen.)



## Module aanspreken

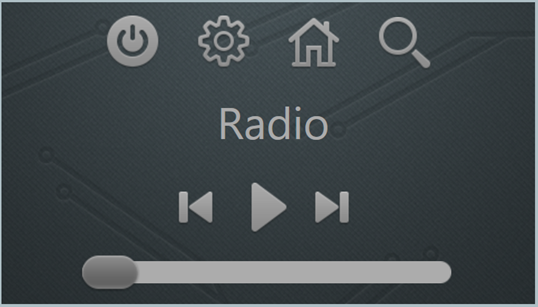
De T3B is geconfigureerd als Slave, de STM32F7 gedraagt zich als de Master.

#### Command format



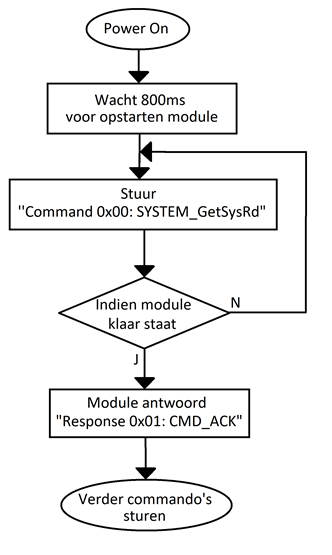
Elk commando begint met een identieke start byte (0xFE). Gevolgd door een byte voor het Command Type, die definieert tot welke groep van commandos de Command ID toebehoort. ( 0x00: System, 0x01: Stream, 0x07: Notify) De command ID bevat een unieke code voor elk mogelijk commando. Met het Serial Number kunnen sessies tussen verschillende modules onderscheiden worden. (Bvb. Master zendt Serial Number 0 naar Slave, Slave antwoordt met Serial Number 0.) De Lenght (2 bytes lang) duidt het aantal data bytes aan dat verwacht mag worden. In de Data Byte zitten gegevens van Master of van de Slave. (Bvb. Indien de Master van radiokanaal wilt veranderen, stuurt hij in de data bytes het gewenste kanaalnummer.) Het Commando wordt afgesloten met de End Byte (0xFD).

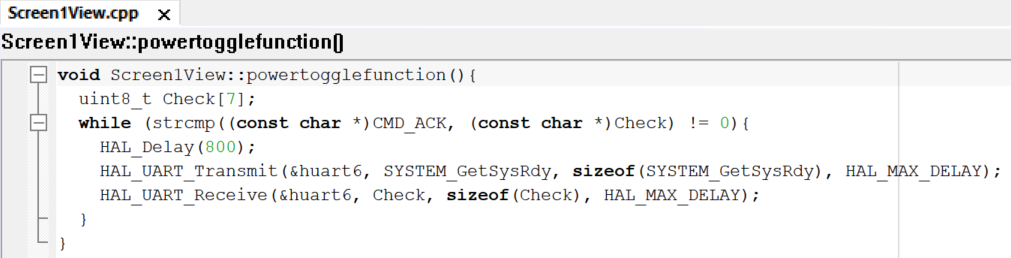
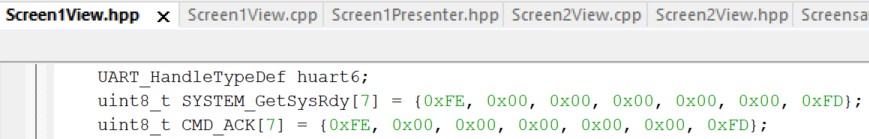
Onze radio wordt aangestuurd door middel van Screen1 die we ontworpen hebben in TouchGFX.



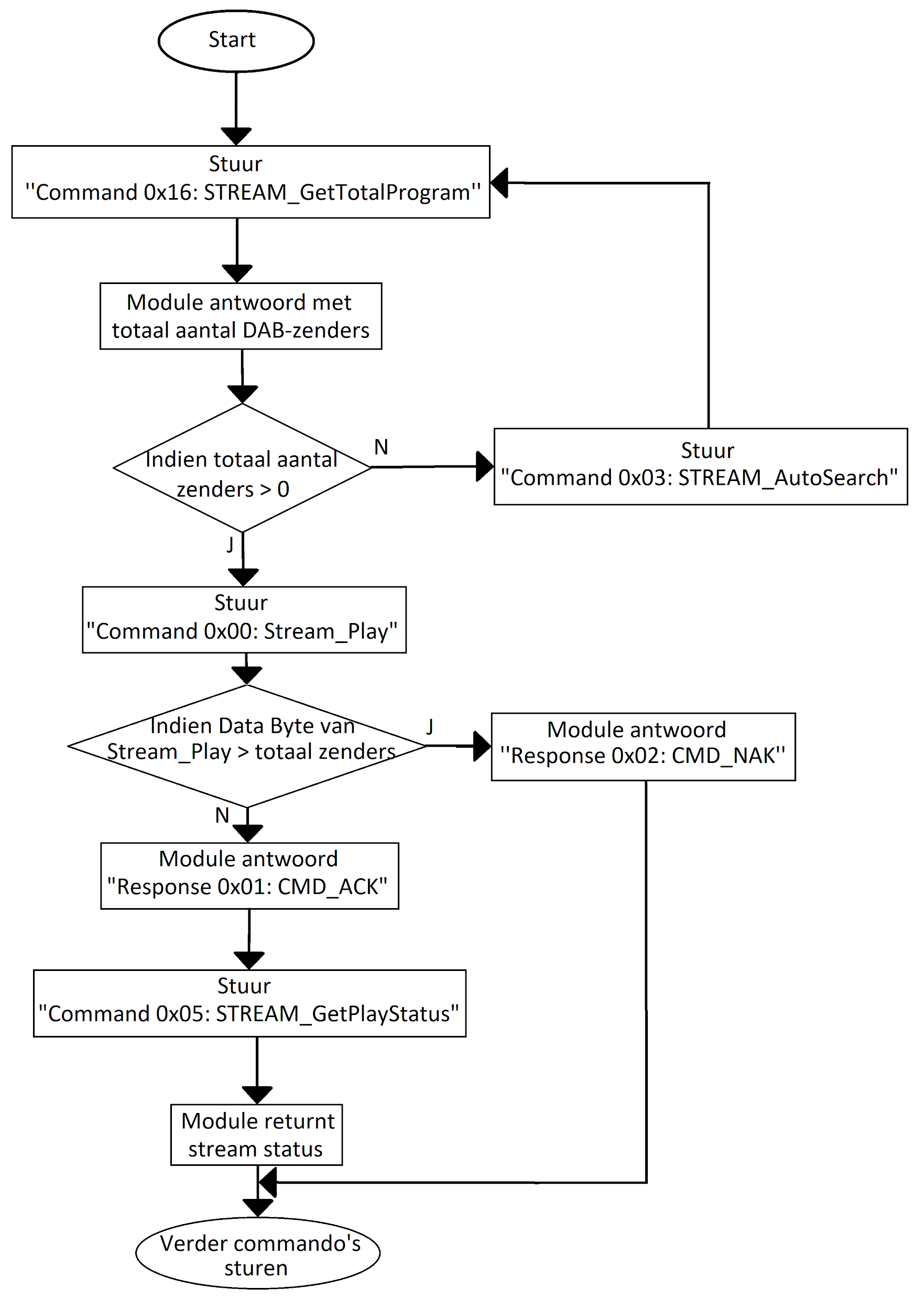
In Screen1View.hhp en Screen1View.cpp kunnen we nu functionaliteit geven aan onze widgets.

#### Flowcharts

Voor er commando’s kunnen gestuurd worden moet de module eerst geïnitialiseerd worden volgens onderstaande flowchart.

In code ziet dit er zo uit:

Deze flowchart toont aan hoe je een DAB programma kan afspelen:



Wanneer de master STREAM\_GetTotalProgram stuurt, zal de slave antwoorden met het aantal programma's dat in zijn database zit. Indien deze database nog leeg is, moet de master STREAM\_AutoSearch sturen met een start en stop frequency in de data bytes. De Slave zal tussen deze frequenties alle zichtbare DAB stations in zijn database opslaan. Met het STREAM\_Play commando kan vervolgens een station gekozen worden. STREAM\_GetPlayStatus kan de status van de stream opvragen. (0: Playing, 1: Searching, 2: Tuning, 3: Stop, 4: Sorting change, 5: Reconfiguration)

# Problemen en hun oplossingen

## PCB pin headers te groot

Na het bestellen van het PCB merkten we dat de gaten van de pin headers te groot waren voor de T3B module en dat de gaten voor de pinheaders voor het STM32F7 board te ver uiteen stonden. We moesten dus helaas het PCb terug opnieuw bestellen met de aanpassingen.

## Voedingsspanningen

Na enkele metingen te doen merkten we dat de voedingsspanningen voor de 3.3v niet op het juiste niveau zaten. Eerst gaf deze voedingslijn 3.3v aan zonder belasting maar als we de module aanlegde zakt deze voedingslijn naar 2.4v. Na even zoeken kwamen we erachter dat bij deze component , de ADP1712, de ground lijn niet verbonden was met de rest van de ground lijnen. Dit was dus een fout in het PCB, maar deze hebben we dan verbonden met een draadje. Nadat we dit deden gaf de voedingslijn zonder belasting 5v aan, na enkele aanpassingen te doen en alle componenten even na te kijken gaf deze voedingslijn 1v. Na de component in kwestie te vervangen was dit probleem nog steeds niet opgelost. We hebben er dan voor geopteerd om de voedingslijn aan te sluiten op de 3.3v die we kregen van het STM32F7 board. Dit kon wel voor problemen zorgen omdat we dan de ADP1712 niet meer gebruikte voor een soft-start en beide signalen, 1.2v en 3.3v, samen aankomen.

## Te veel stroom

Nadat we enkele problemen hadden opgelost sloten we alles aan. De DAB+ module plaatsten we in de shield en de shield plaatsten we op het STM32F7 board. Dit alles sloten we dan aan via een micro- usb kabel aan een computer. Maar als we dit deden begon het scherm te flikkeren en merkten we dat er iets mis was. Na enkele datasheets door te nemen en de stroom te meten via een labovoeding merkten we dat we meer als 500mA nodig hadden, dit zorgt voor een probleem omdat de usb aansluiting van het STM32F7 board gelimiteerd is tot 500mA. Dit hebben we opgelost door een voedingsstekker van 12V te gebruiken en deze zetten we dan met een 7805 om tot 5V die we dan op een externe ingang van het board plaatsen.

### Pinheaders verkeerd op PCB

Bij het testen van de DAB+ module ontdekten we dat er bij het maken van het PCB een fout is gebeurd bij het nummeren van de pinheaders voor de DAB+ module. Zo is er een pin header die gespiegeld is en eentje die achterstevoren staat. Dit hebben we dan moeten oplossen met enkele draadjes.

## Grounds niet verbonden

Door het testen merkten we ook dat enkele grounds niet verbonden waren met de gemeenschappelijke ground. Dit komt ook door een fout tijdens het maken van het PCB, maar het was reeds te laat om een nieuwe te bestellen. Daarom hebben we deze grounds ook met enkele draadjes aan de gemeenschappelijke ground moeten hangen.

## Datasheet verkeerd geïnterpreteerd

Toen we audio probeerden af te spelen stootten we opnieuw op een fout in het PCB. Je kon namelijk enkel audio afspelen als er een male aux in de female aux stak en dan kwam de audio via de speakers. Deze fout kwam door het verkeerd interpreteren van de datasheet van de amplifier. Deze geeft namelijk de optie om audio via een aux aansluiting af te spelen, hiervoor moet er een aux aansluiting zijn en een verbinding tussen beide speakers. Dit laatste was niet het geval op onze PCB. Daarom hebben we de aux aansluiting verwijderd en alle pads met elkaar verbonden, zo kan er wel audio afgespeeld worden.

## Geen reset

Enkele pogingen later om de DAB+ module te laten werken lazen we in de datasheet dat de module een reset puls nodig heeft. Deze hadden we niet ingerekend in ons design en deze moesten we dan ook later aanbrengen. Dit doen we met een switch tussen een 3.3V verbinding die we halen van het STM32F7 board.

## Geen opslag

Wegens tijdgebrek en allerlei problemen die opdoken hebben we de opslag van een nieuwsbericht niet kunnen integreren in het eindproduct.

## Geen communicatie met DAB module

Door alle problemen met het PCB hebben we geen connectie kunnen maken met de DAB module. De problemen begonnen met verkeerde pin headers die we dan moesten verbinden met draadjes. Alsook ondervonden we dat we te veel stroom nodig hadden dan de usb poort op het SMT32F7 board aan kon. Hiervoor moesten we dan ook een externe voeding voor maken.

Daarna ontdekten we dat er enkele grounds niet aan de gemeenschappelijke ground verbonden waren. Dit hebben we dan ook moeten verbinden met draadjes.

We ontdekte later dan ook dat er een reset signaal aangelegd moest worden, dit hebben we dan trachten te doen met een switch maar dit hielp ook niet naar het verkrijgen van een antwoord van de DAB module.

Als we alle signalen juist aanleggen en enkele commando’s sturen krijgen we geen antwoord en komt er op de audio lijnen 1.6V te staan dat af en toe zakt. Dit gebeurt niet als we het reset signaal niet aanleggen.

# Logboek

## Aäron

Totaal aantal gelogde uren: 223,5 uren

## Ward

# Besluit & Reflectie

Als we een besluit moeten trekken over ons project dan moeten we eerlijk toegeven dat het niet af is. We hebben een mooie dynamische interface, met een instelbare klok. Op het vlak van audio is alles in orde, we kunnen perfect een audio signaal afspelen en van volume veranderen. Hier hebben we ook een lange tijd aan bezig geweest omdat er vanalles op het PCB niet klopte.

Waar het project niet compleet is en wat eigenlijk onze key factor zou moeten zijn is ons signaal. We hebben niet kunnen communiceren met de DAB+ module waardoor we gaan audiosignaal konden opvangen. Al onze code om de juiste commando’s te sturen staat klaar om te gebruiken als we de connectie kunne leggen tussen het STM32F7 board en de DAB+ module.

De grootste oorzaak van het incompleet zijn van ons project is de vele problemen met het PCB. Het was namelijk de eerste keer dat we een PCB moesten maken en zonder enige voorkennis hebben we dit proberen te doen en dit heeft geleid tot de vele problemen. We hadden dit kunnen voorkomen door meer hulp te vragen aan de leerkrachten en begeleiders om te kijken of onze PCB in orde was.

Daarnaast heeft tijd ook een grote rol gespeeld. We hebben er namelijk wel veel tijd ingestoken maar dit had zeker nog meer kunnen zijn. We zijn vastberaden om toch nog na dit jaar het project af te werken in onze vrije tijd.

Op vlak van samenwerking verliep alles vlot, ook al was het niet altijd even handig om met twee een probleem op te lossen. De taken waren echter wel goed verdeeld, Ward focuste zich eerder op de software terwijl Aäron zich focuste op de hardware en de probleem oplossing.

# 

# Bijlagen

## T3B module datasheets

* Command Specification of Slave Mode UART.pdf
* datasheet T3B.pdf
* F19008.pdf
* T3B I2C English.pdf
* T3B\_Antenna Selection Guide\_V0.2.pdf

## DAB+ shield componenten datasheets

* 21889b-64653.pdf
* ADP1712\_1713\_1714-1503310.pdf
* RK097-1370742.pdf
* sj2-3581x-smt-1313218.pdf
* tpa6021a4.pdf

## DAB+ shield PCB

* DAB.sch
* DAB.kicad\_pcb

## STM32F746g board

* Reference Manual .en.DM00124865.pdf
* User Manual Bord .en.DM00190424.pdf