

Inhoudstafel

[Inhoudstafel 1](#_Toc72319418)

[Voorwoord 2](#_Toc72319419)

[Inleiding 3](#_Toc72319420)

[1 Hardware 4](#_Toc72319421)

[1.1 Blokschema 4](#_Toc72319422)

[1.2 Schema’s 5](#_Toc72319423)

[1.3 Beschrijving 7](#_Toc72319424)

[1.3.1 Schakelende voeding 7](#_Toc72319425)

[1.3.3 LDO Voltage Regulator 9](#_Toc72319426)

[1.3.4 µC 9](#_Toc72319427)

[1.3.5 RGB LED’s 10](#_Toc72319428)

[1.3.6 Voltage Level Translator 11](#_Toc72319429)

[1.3.7 Rotary Encoder 12](#_Toc72319430)

[1.3.8 EEPROM 13](#_Toc72319431)

[besluit…. 3](#_Toc72319432)

Voorwoord

TEKST TEKST TEKST ...

Inleiding

Deze bundel gaat over het project dat ik gemaakt heb voor het vak Practice Enterprise 2, hierbij waren de minimale vereisten dat ik een PCB ontwerp, Software schrijf (liefst in C op een µController) en ten slotte het project een nuttige toepassing heeft. Het is ook belangrijk dat er een veilig project gekozen werd en de kostprijs niet te hoog lag.

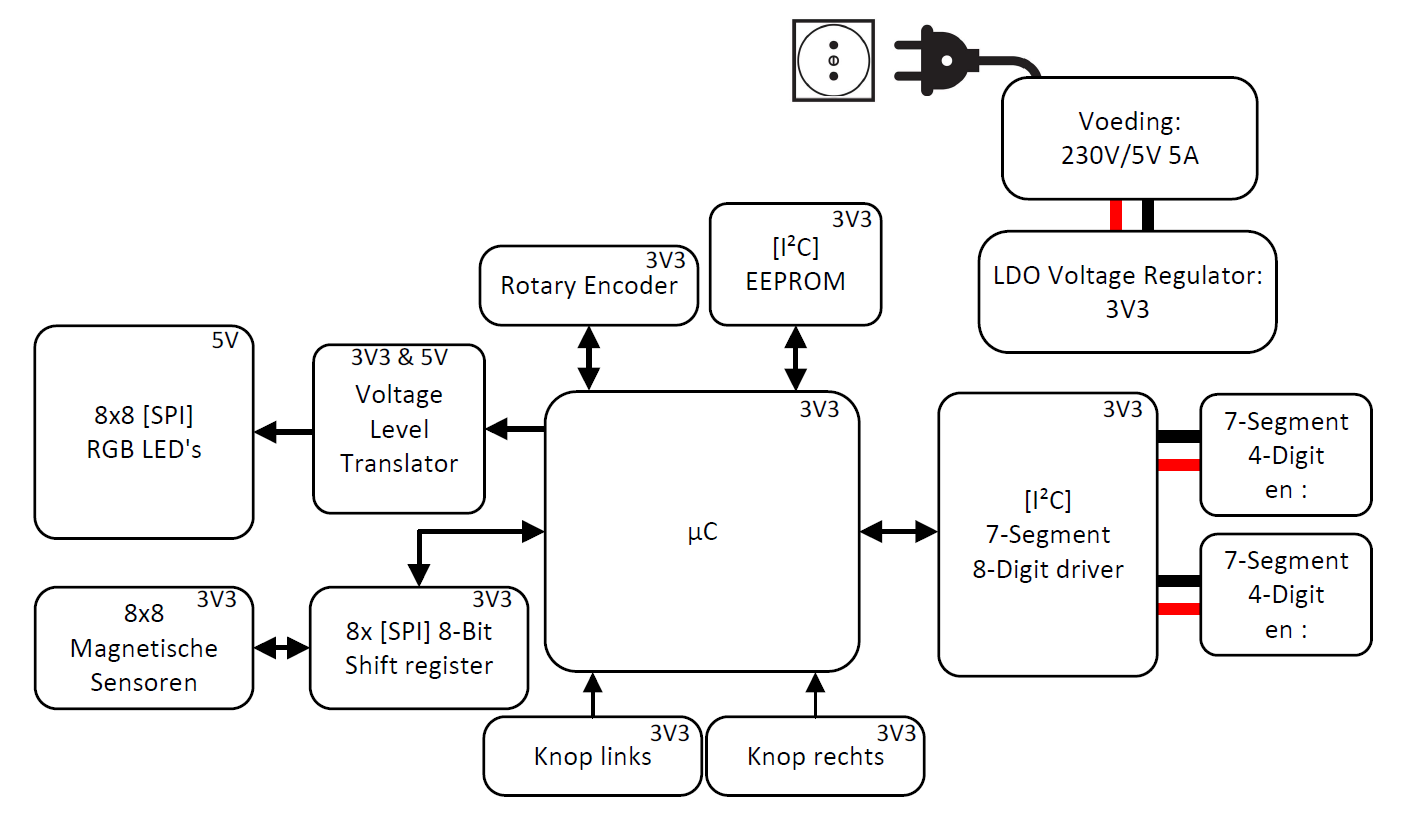
Na enkele projectjes dat ik heb voorgesteld koos ik om een elektronisch educatief schaakbord te maken. Hiermee zouw u dan de mogelijkheid krijgen om te leren hoe men iedere pion zet en de werking van een schaakklok. Daarnaast bevat het schaakbord over een puntensysteem dat bijgehouden wordt.

# Hardware

## Blokschema

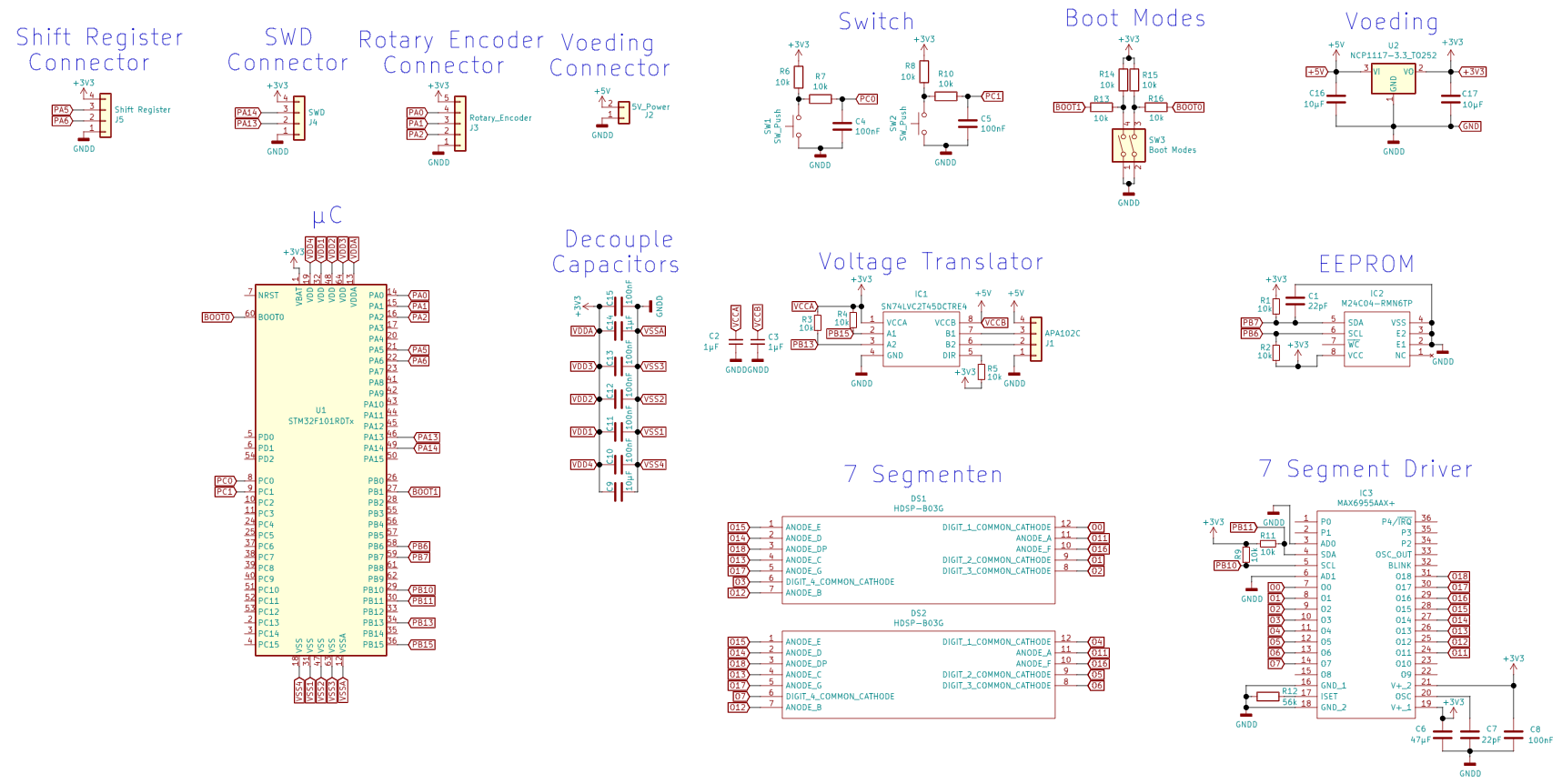
In het blokschema zien we hoe de verschillende blokken verbonden zijn met elkaar maar dit in een vereenvoudigde weergave. Hierbij kan je zien hoe alles logisch met elkaar verbonden is d.m.v. voedingsaansluitingen en communicatielijnen.

Figuur , Blokschema

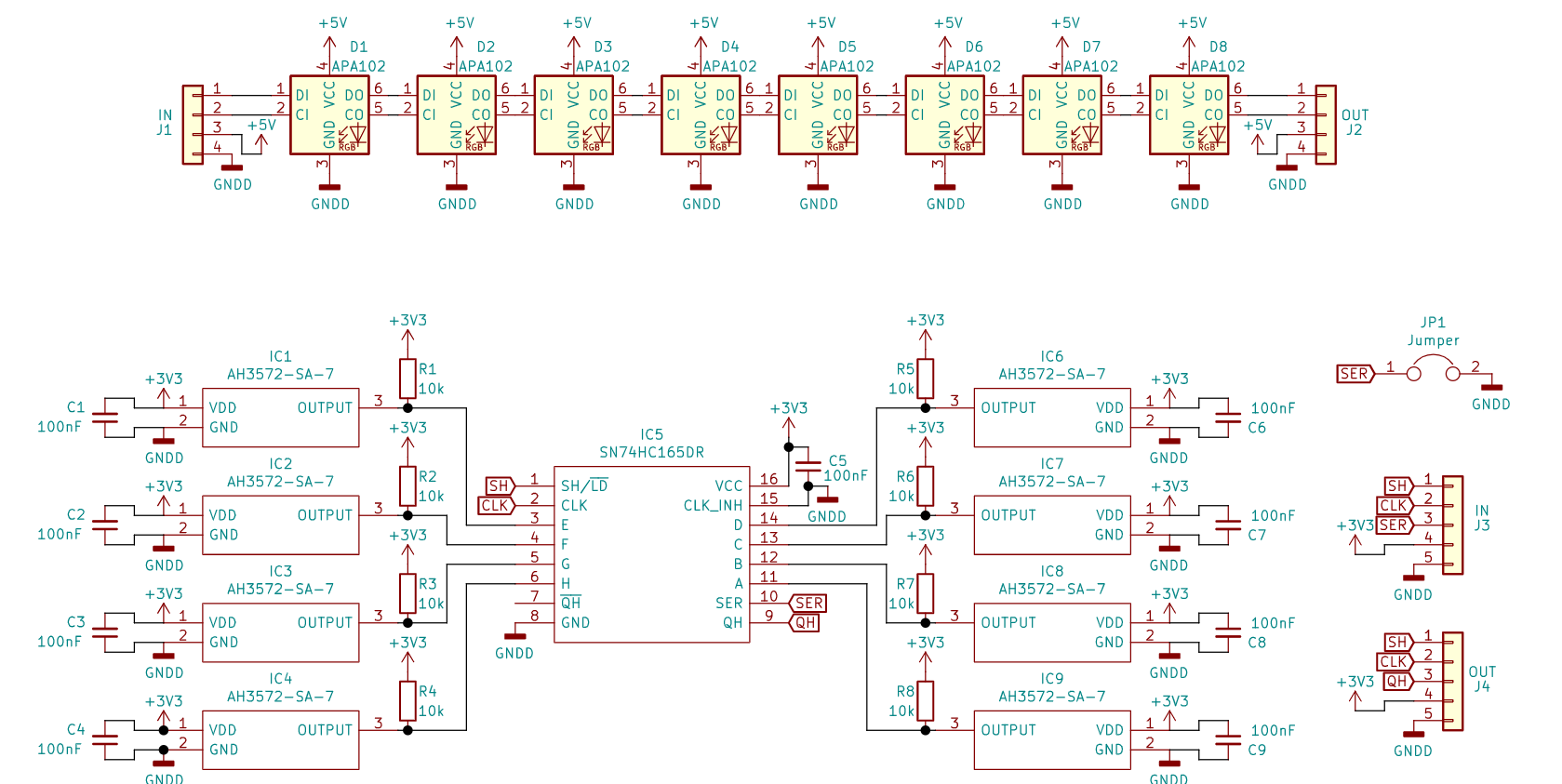


## Schema’s

Dit is het aansluiting schema van het hoofdbord, hierin zit: de µC, 7 Segmenten, Rotary Encoder ... Later wordt ieder deel nog besproken.



Figuur , Schema hoofdbord

Dit is het aansluiting schema van de borden die zich onderaan de pionnen liggen, Later wordt ieder deel besproken.

Figuur , Schema pionnen

## Beschrijving

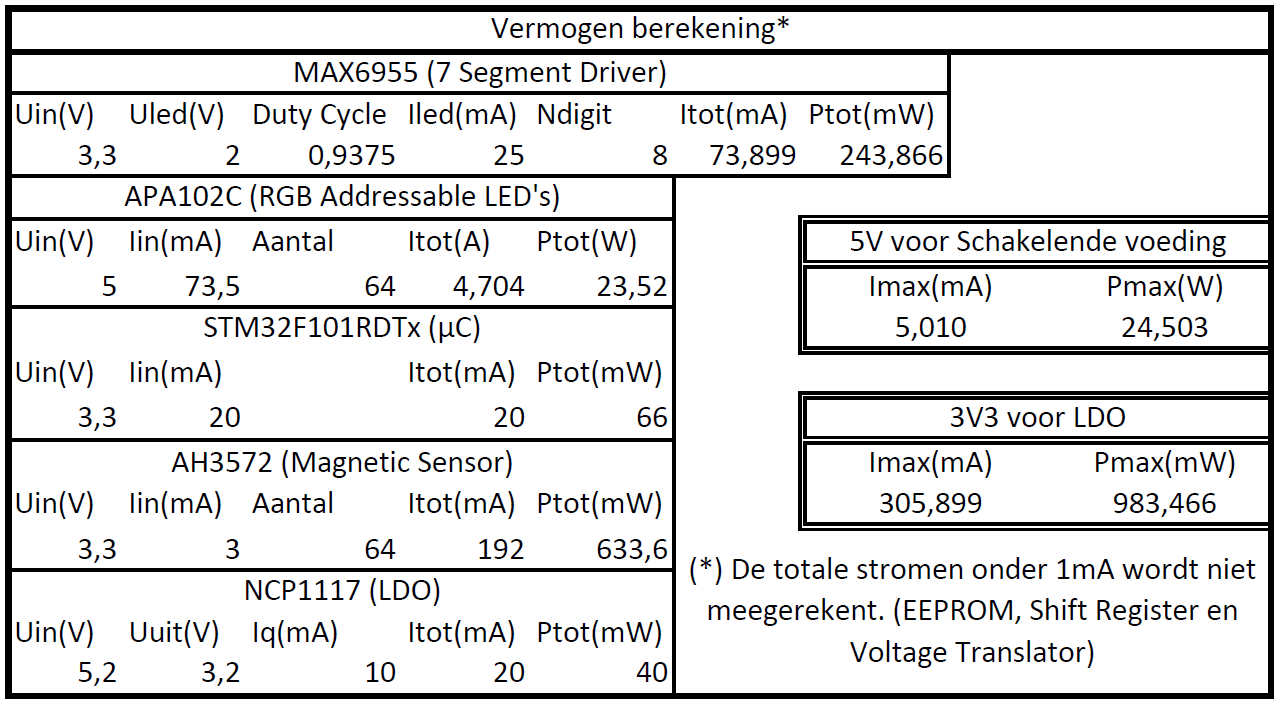
Hierin ga ik ieder deel bespreken en uitleggen hoe ik eventueel iets berekend heb.

### Schakelende voeding

Het E-Schaakbord wordt gevoed door de 230V AC netspanning die wordt omgezet naar 5V DC d.m.v. een schakelende voeding. De voeding die ik gekozen/berekend heb kan maximaal 5A leveren of in andere woorden .

Dit heb ik berekend a.d.h.v. het maximaal vermogen dat gebruikt kan worden op te tellen van ieder component. Daarmee heb ik gebruik gemaakt van de maximale spanning en stromen die uit de datasheet komen van ieder component.

Tabel , Vermogen berekening



Figuur , Voorbeeld voeding

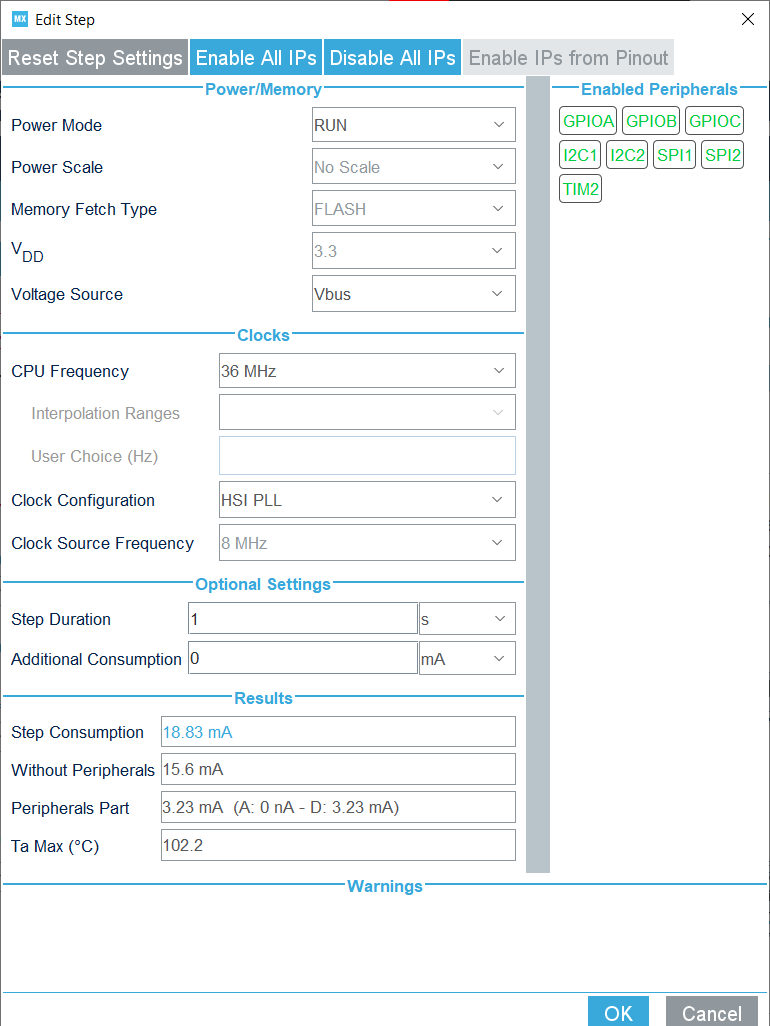
Zoals u merkt heb is mijn voeding’s keuze redelijk krap genomen, dit is doordat ik het maximale berekend heb en daarmee juist gekozen heb voor 5A. Het verbruik wordt beperkt d.m.v. de software die ik er insteek, omdat ik nooit al mijn RGB LED’s tegelijk de kleur wit ga maken zal de stroom dus beperkt blijven.

APA102C (RGB Addressable LED’s): kleur Wit

APA102C (RGB Addressable LED’s): kleur Blauw

MAX6955 (7 Segment Driver):

STM32F101RDTx (µC):



Itot is gesimuleerd a.d.h.v. CubeMX zijn PCC Tool.

Hierbij heb ik alles ingesteld op het maximum dat ik kan gebruiken i.v.m. het schaakbord: HSI PLL, de CPU Frequency, I/O en meer.

Figuur , Simulatie stroomverbruik

AH3572 (Magnetic Sensor):

NCP1117 (LDO Voltage Regulator):

### LDO Voltage Regulator

Nadat de 230V AC omgezet is naar 5V DC zal ik deze 5V omzetten naar 3V3, hiermee zal ik de andere componenten voeden buiten de RGB LED’s.

Ik heb gekozen voor een LDO omdat:

* Het maximaal vermogen onder de 1W was en dus dit vermogen nog gedissipeerd kan worden d.m.v. een LDO.
* Het voordeel is ook dat LDO’s qua structuur simplistisch zijn en dus onafhankelijk zijn van een externe spoel i.t.t. een Buck-Converter.

A.d.h.v. de vorige berekeningen (Tabel 1) kwam ik uit dat de LDO een minimaal vermogen van ±983mW gedissipeerd wordt. Hiermee ben ik dan aan de slag gegaan en opzoek gegaan naar een LDO en kwam ik uit op de NCP1117.

Deze heb ik gekozen door de volgende berekeningen te maken:

Deze berekeningen heb ik geraadpleegd via een handig document van Microchip**<link>**

### µC

Dit is een zeer belangrijk keuze die gemaakt moet worden, daarmee heb ik genoeg research gedaan op het internet d.m.v. online guides op te zoeken en meer.

De eerste vraag die ik mezelf stelde is of ik een 8-bit of 32-bit microcontroller kies. Ik koos voor een 32-bit microcontroller wegens volgende redenen:

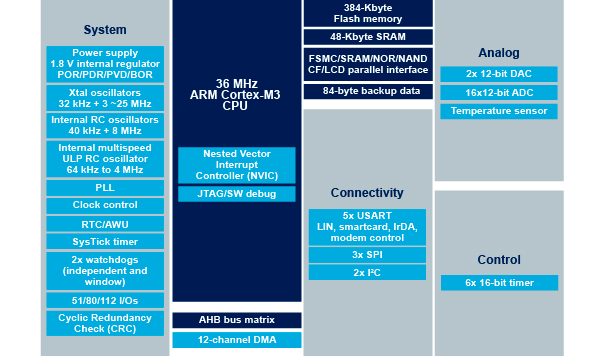
* Communicatieprotocollen zoals SPI & I²C zijn eenvoudiger te implementeren.
* Om lengte/grootte van software te beperken.
* Eenvoudiger te debuggen.

Nu was de keuze voor welke fabrikant ik zal gaan, hier waren vooral 2 uitblinkers bij de STM32 of PIC32. Ik koos voor de STM32 wegens volgende redenen:

* Meer gekend in hoe deze ge programmeert wordt en debugt.
* Zeer stabiel en handige Toolchain/IDE ondersteuning.

Ten slotte ben ik opzoek gegaan naar een STM32 die een genoeg aantal I²C en SPI-protocollen had en die genoeg geheugen heeft.

Toen kwam ik uit op de STM32F101RDT6, deze bevindt zich in een LQFP-64 package dat nog handsoldeerbaar is en ook de mogelijkheid heeft in dezelfde package een groter geheugen te kiezen of een andere reeks.



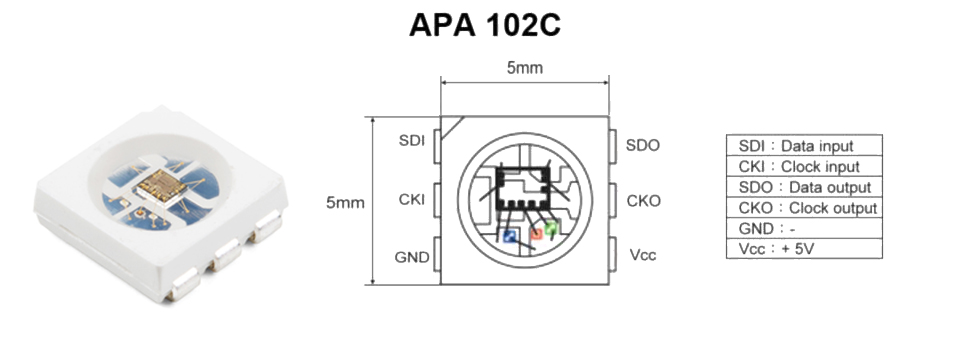
Figuur 6, Circuit Diagram

### RGB LED’s

Om de schaakvakken een kleur te geven maak ik gebruik van adresseerbare RGB LED’s, hierdoor kan ik elk vak individueel een andere kleur geven wat dat belangrijk is voor aan te duiden wat de pion hun mogelijke zetten zijn en meer.

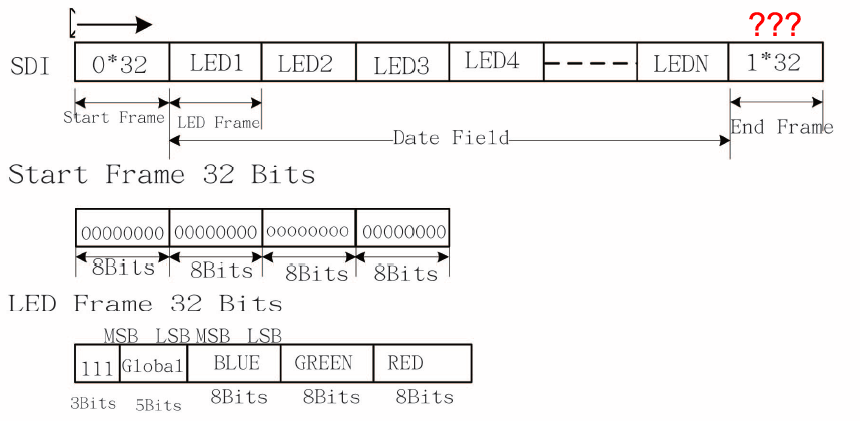
Ik heb gekozen om gebruik te maken van de APA102C LED’s omdat:

* Worden aangestuurd via het standaard SPI-Protocol.
* Heeft een snellere CLK-frequentie maximaal 1,2MHz.
* Beschikt over een aparte PWM-instelling om de lichtsterkte te regelen.



Figuur , Illustratie APA102C

Zoals u hierboven ziet wordt de APA102C’s aangestuurd:



Figuur , APA102C SPI-Protocol

Eerst wordt er een Start Frame gestuurd met 4 bytes die ‘0’ zijn.

Daarna wordt er de data van de eerste LED gestuurd met als inhoud:

1. “111” om te beginnen, 3 bits.
2. Daarna de data wat de licht intensiteit is van al de LED’s, 5bits.
3. De lichtintensiteit van de kleur blauw, 1 byte.
4. De lichtintensiteit van de kleur groen, 1 byte.
5. De lichtintensiteit van de kleur rood, 1 byte.

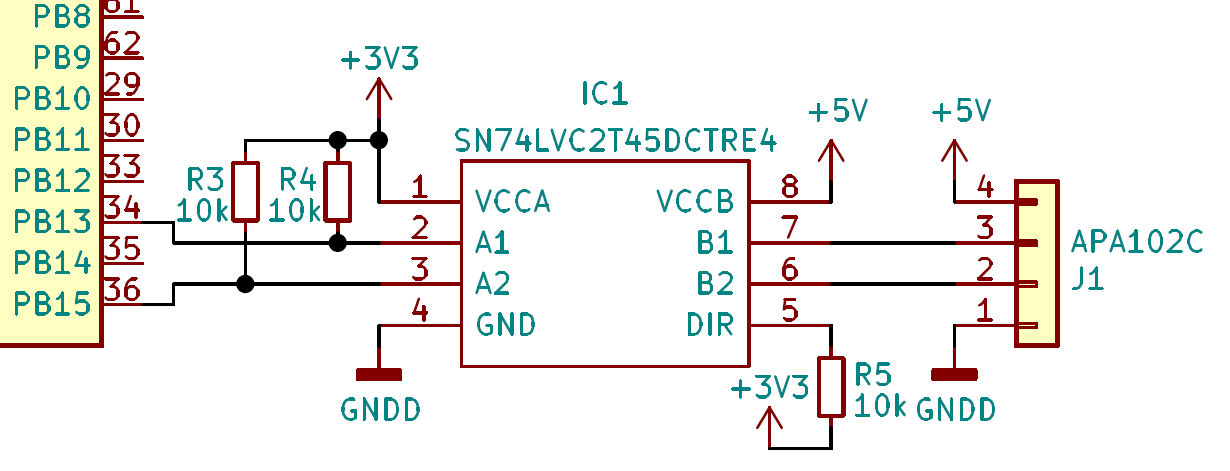
Vervolgens stuurt men de data door van de tweede LED.

Ten slotte kan men een eind frame sturen die bestaat uit allemaal enen, de grote van deze frame is gelijk aan het totaal aantal LED’s delen door 2.

Dus het schaakbord bestaat uit 64 RGB LED’s, dat wil zeggen dat er 32 keer 4 bytes die ‘1’ zijn (eind frame) verstuurt moet worden. Het is gemakkelijker om op het einde een Start frame sturen om terug opnieuw te beginnen waardoor ik zo min mogelijk tijdverlies.

### Voltage Level Translator

Om de APA102C’s te laten communiceren met de µC zal ik de 3V3 logica moeten omzetten naar 5V logica. Om deze om te zetten maak ik gebruik van een Voltage Level Translator. Ik heb gekozen voor de SN74LVC2T45, hiermee zal ik de SCK (seriële clock) en de SDI (seriële data in) omzetten naar 5V logica.

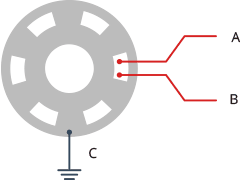
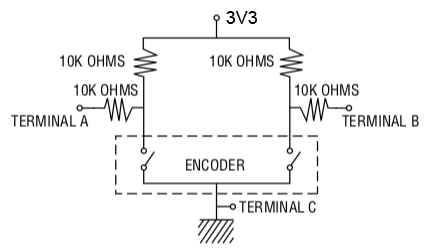
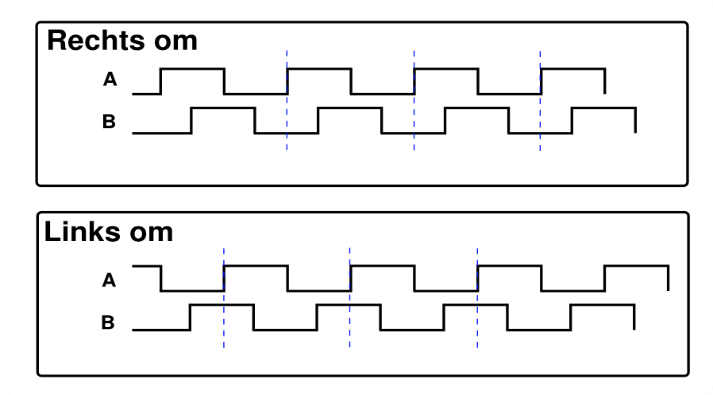
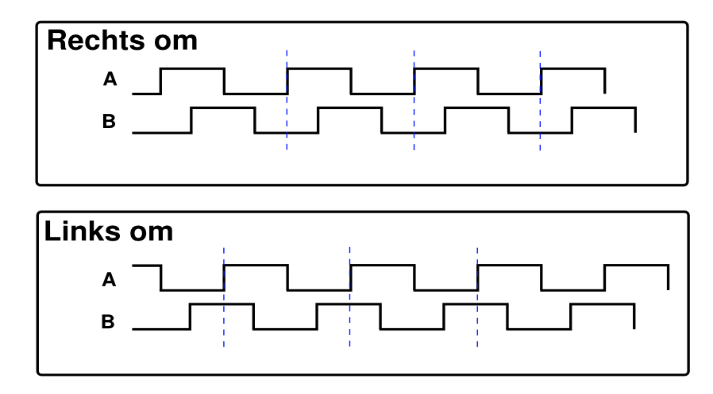


Figuur , Schema Voltage Translator

### Rotary Encoder

Voor het schaakbord gebruiksvriendelijker te maken bevindt er zich een rotary encoder tussen de 2 drukknoppen op de schaakklok. Hiermee kan voor het spel begint de tijden ingesteld worden en de kleur van de schaakvakken.

Een rotary encoder werkt als volgend:

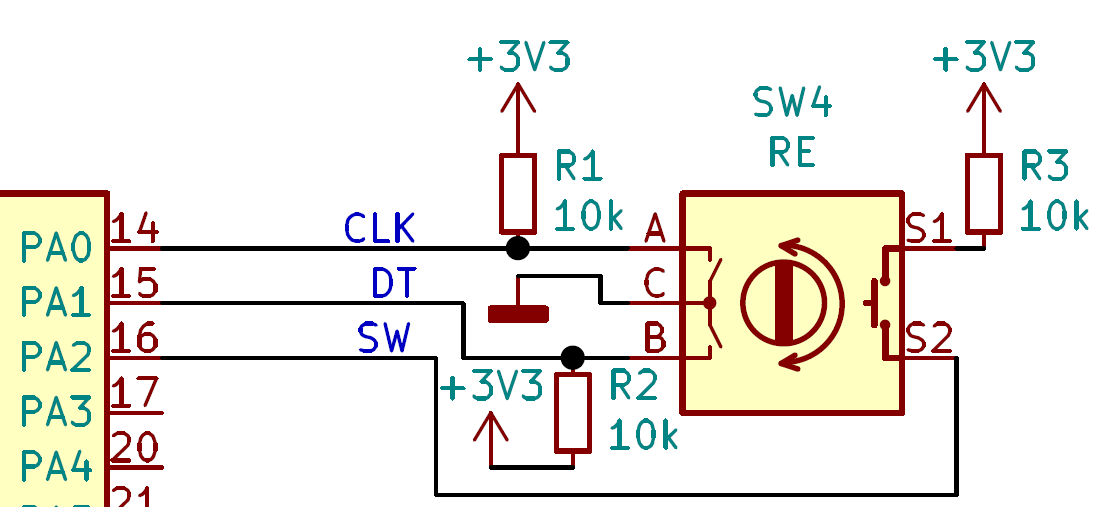


Figuur , Werking Rotary Encoder

Wanneer er één stand in wijzerzin (↻) gedraaid wordt zal Terminal A 90° voor-ijlen omdat deze voor Terminal B contact zal maken met de GND of de 3V3. Als daar na gelijk er één stand in wijzerzin (↻) opnieuw gedraaid wordt zal nu Terminal B op hetzelfde contact van Terminal A staan.

Dit is omgekeerd als men één stand in tegenwijzerzin (↺) draait zal Terminal B 90° voor-ijlen omdat deze voor Terminal A contact zal maken met de GND of de 3V3. hetzelfde als er daar na gelijk één stand in tegenwijzerzin (↺) opnieuw gedraaid wordt zal nu Terminal A op hetzelfde contact staan als dat van Terminal B.

Deze terminals worden vervolgens zo aangesloten op de STM32:



Figuur , Schema Rotary Encoder

In de software zal nu gewoon iedere pin continu uitgelezen worden en als er een verandering plaats vindt zal men te weten komen welke richting er gedraaid werd.

De rotary encoder bevat ook een schakelaar die schakelt wanneer op de knop geduwd wordt, hiermee is alles wat meer gebruiksvriendelijk zoals om een actie te voltooien.

### EEPROM

EEPROM of Electrically Erasable Programmable Read-Only Memory is een geheugen IC die data opgeslagen houdt terwijl de IC niet gevoed wordt, dit heet men ook permanent geheugen. EEPROM’s kunnen ook hun blokken data (1 byte) individueel her programmeren waardoor er niet alles verwijderd moet worden. Het nadeel is wel dat deze niet oneindig keer herschreven kan worden, maar tegenwoordig kunnen de meeste meer dan 1 miljoen keer herschreven worden.

Ik heb gekozen voor de M24C04 omdat deze via I²C te werk gaat en een geheugen heeft van 4kbits of 512bytes. Hierin zal ik de score zetten van het vorig spel.

besluit….

Typ hier je besluit