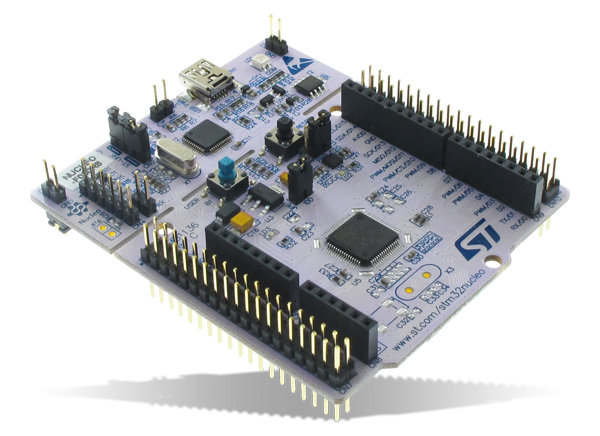
GPIO challenge Embedded systems

Jim Wanten, 06-10-2022



# Inleiding

Voor deze challenge heb ik gekozen om met een rotary encoder 2 leds te besturen. Als er naar rechts gedraaid wordt gaat de rechter led aan en andersom. Hierbij heb ik gebruik gemaakt van de stm32, KY-040 en leds.

# Inhoudsopgave

[Inhoudsopgave 2](#_Toc116042781)

[Onderzoek 2](#_Toc116042782)

[Code & design 3](#_Toc116042783)

[Design 3](#_Toc116042784)

[Code 3](#_Toc116042785)

[Proof 6](#_Toc116042786)

[Reflectie 6](#_Toc116042787)

[Bronnen 6](#_Toc116042788)

# Onderzoek

Om het project te beginnen heb ik als eerst onderzocht welke pins ik kon gebruiken. Hiervoor heb ik gebruik gemaakt van de presentatie “GPIO”. Hierbij heb ik ontdekt dat voor de rotary encoder de pinnen A0 en A1 zullen voldoen. Voor de 2 leds zijn dit pinnen d3 en d4. Deze pinnen worden allemaal nog niet gebruikt in de stm32 waardoor ze veilig zijn voor mij om te gebruiken. Zodra ik wist welke pinnen ik kon gebruiken heb ik de pinout van de stm32 nulceo f303re in de UM1724 User manual gebruikt.

Graphical user interface

Description automatically generated with medium confidence

Pin A0 en A1 zijn bij dit bord dus PA0 en PA1. Pin D3 en D4 zijn hierbij PB3 en PB5. Dit betekent dat ik port A en B beide moet gebruiken. Port A voor de rotary encoder en port B voor de leds.

# Code & design

## Design

Hieronder is een schematic te zien hoe ik de verschillende onderdelen heb aangesloten.

Graphical user interface, diagram

Description automatically generated

Figuur 1 schematic

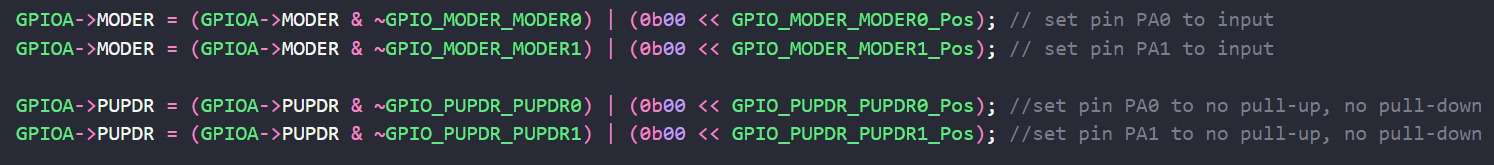
Ook heb ik een class gemaakt genaamd “led”. Dit heb ik gedaan zodat het in de toekomst makkelijker is om een led aan te sluiten.

Table

Description automatically generated

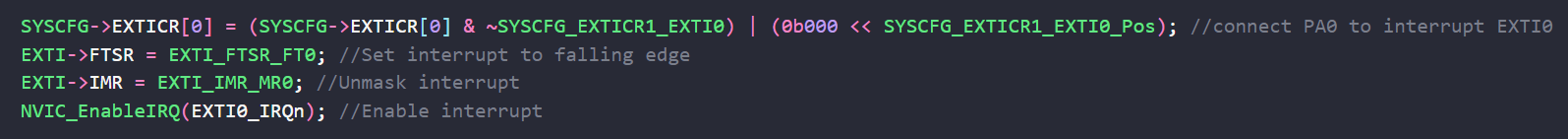
## Code

Als eerste moet ik de juiste pins voor de rotary encoder initialiseren, dit doe ik met het register MODER om de pins op input te zetten, hierbij moet ik de eerste 4 bits van het MODER register op 0 zetten. Om te voorkomen dat ik eerdere aanpassingen in een ander gedeelte van het register verander gebruik ik een mask zodat alleen de eerste 4 bits aangepast worden. Het register PUPDR gebruik ik om de pins non pull-up en non pull-down te maken. Dit gebeurt exact hetzelfde als bij MODER alleen dan voor het PUPDR register.



Figuur 2 Initialize PA0 & PA1 as non pull-up non pull-down input

Nadat de pins correct geintialiseerd zijn moet een van deze pins geconnect worden aan de juiste interrupt. Hiervoor heb ik EXTI0 gebruikt omdat dit de eerste is. In de 2e regel code wordt de interrupt op falling edge gezet om de input van de rotary encoder juist op te vangen. Na deze stap wordt de interrupt geunmasked. En als laatste wordt de interrupt aan gezet zodat deze gebruikt kan worden.



Figuur 3 Connect en intialize the correct pin to the correct interrupt

Ook heb ik low power gebruikt om minder power te gebruiken. Hiervoor heb ik de functie \_\_WFI gebruikt.

A picture containing graphical user interface

Description automatically generated

Figuur 4 low power

Hieronder is de led class te zien. Deze class heeft 4 functies. De eerste functie is de constructor, deze functie zorgt ervoor dat de data die meegegeven is op de juiste manier opgeslagen wordt en het object correct geintialiseerd wordt. De functie init initialiseerd de juiste pin, dit gebeurt door de pinnumber en port mee te geven. Door deze functie hoeft je niet bij elke led in de datasheet te kijken wat er aangepast moet worden. Hier kun je het pinnummer invullen en de port en de pin wordt automatisch ingesteld op push-pull output. De functies on en off doen niks anders dan de pin met het juiste pinnummer aan of uit zetten.

Text

Description automatically generated

Figuur 5 led class

Als laatste is er een interrupt handler nodig, deze interrupt handler zorgt er elke keer dat er aan de rotary encoder gedraaid wordt dat de juiste led aan gaat.

Text

Description automatically generated

Figuur 6 interrupt handler

# Proof

A picture containing text

Description automatically generated

Figuur 7 Rotary encoder test

Om te testen of de rotary encoder werkte heb ik in de interrupt handler waar normaal van led geswitcht zou worden een print gezet. Hierdoor kon ik in de serial monitor zien of de rotary encoder het deed.

Ook heb ik een vergelijking gedaan tussen low power en niet low power. Dit heb ik gedaan met Teun Engels. Hiervoor hebben we een multimeter gebruikt om de stroomsterkte over de chip te meten. Het meetresultaat met WFI aan was 0.11ma, zonder WFI was het meetresultaat 0.21 ma. Hier is dus te zien dat de chip in low power mode gaat. Het verschil tussen de 2 zou groter kunnen zijn als alle interrupts uit zouden staan zoals de system clock etc.

# Reflectie

Deze challenge was een combinatie van alle onderdelen die ik hiervoor los gemaakt heb. Hierbij heb ik vooral veel geleerd over het werken met registers. In deze challenge is dit vooral het generaliseren van het gebruik van deze registers. Door de led class te maken heb ik geleerd hoe ik informatie uit een datasheet te gebruiken om een functie te maken waarmee veel verschillende pinnen gemakkelijk te besturen zijn.

# Bronnen

Presentatie “GPIO”, René Bakx

Presentatie “interrupts”, René Bakx

Presentatie “low\_power”, René Bakx

User Manual DM00105823, STMicroelectronics

Reference manual DM00043574, STMicroelectronics