

Voorraadbeheer in magazijnen



Groep BC10

Xavier Claerhoudt

Bram De Smet

Robbe De Vilder

Garben Tanghe

Begeleiders

Prof. D. Colle

Prof. E. De Poorter

Prof. M. Pickavet

Ir. J. Rossey

Ir. P. Stroobant

Ir. J. Vanhie



Project in het kader van het Vakoverschrijdend Projectvak in de Bachelor Computerwetenschappen

2017 - 2018

Inhoudsopgave

Inhoudsopgave	i
Lijst van figuren	iii
Lijst van tabellen	iv
1 Analyse van de probleemstelling	1
2 Planning	2
2.1 Initiële planning	2
2.2 Taakverdeling	2
2.3 Finale planning	2
3 Hardware	5
3.1 Ultra Wide Band	5
3.2 Controller	5
3.2.1 DecaWave DWM1001	5
3.2.2 Raspberry Pi Zero W	6
3.2.3 Lithium-ion Polymeer Batterij en Power Supply	6
3.2.4 Totaal	6
3.3 Drone	6
3.4 Setup	7
4 Software	9
4.1 Verbinding tussen de Ultra Wide Band Location Anchors en de Decawave .	9
4.2 Verbinding tussen de Decawave en de Raspberry Pi	9
4.3 Verbinding tussen de Raspberry Pi en de drone	10
4.4 Verbinding tussen de Raspberry Pi en de server	10
4.5 Setup	10
4.6 Full-mesh	10
5 Resultaten	12
5.1 Kosten	12

Lijst van figuren

2.1	Gantt chart van de initiële planning.	3
2.2	Gantt chart van de finale planning.	4
3.1	Hardware setup	8
4.1	Hardware setup, aangevuld met protocollen	11

Lijst van tabellen

2.1 Taakverdeling	3
5.1 Kosten	13

Samenvatting

Het doel van dit project is om commercieel beschikbare drones te voorzien van een functionele on-board controller en gebruik te maken van een controlebord dat instaat voor de drone-aansturing en -lokalisatie.

Daarnaast is de controller verantwoordelijk voor communicatie met een centraal controlepunt en met naburige toestellen.

Tegen het eind van het project moeten drones in staat zijn om autonoom en probleemloos een vanuit het controlepunt verzonden route af te leggen.

Hoofdstuk 1

Analyse van de probleemstelling

Voorraadbeheer in een magazijn gebeurt traditioneel door werknemers die op een vorklift handmatig barcodes inscannen. Zoals u wel kan aanvoelen is dat een zeer langzame, dure en gevaarlijke aanpak. Een kosteneffectievere oplossing is het gebruikmaken van automatisch aangestuurde drones die met een camera de barcodes kunnen inscannen.

Het doel van dit vakoverschrijdend project is om een basis te leggen voor dit systeem. In eerste instantie is het doel om een drone een vooraf bepaalde route autonoom te laten vliegen in een magazijn. In tweede instantie is het doel om het systeem uit te breiden naar meerdere drones die zonder accidenten door elkaar kunnen laten vliegen om zo het proces te versnellen, of grotere magazijnen te onderhouden.

Hoofdstuk 2

Planning

2.1 Initiële planning

De initiële planning is terug te vinden op figuur 2.1.

2.2 Taakverdeling

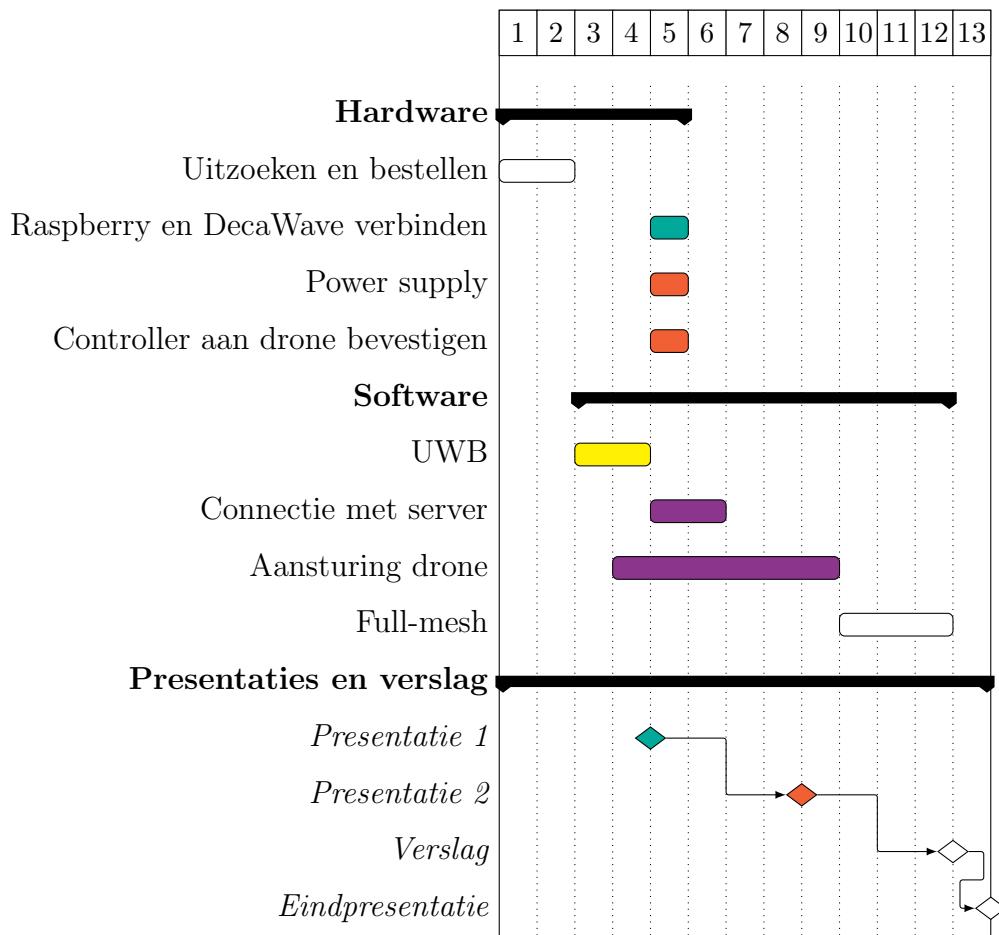
Wie doet wat? Dit kun je terug vinden in tabel 2.1.

2.3 Finale planning

De finale planning is terug te vinden op figuur 2.2.

De grootste verschillen tussen de initiële planning en de finale planning zijn ...

De motivatie van de belangrijkste wijzigingen zijn ...



Figuur 2.1: Gantt chart van de initiële planning.

Wat?	Wie?	Kleur
Uitzoeken en bestellen	Iedereen	WIT
Raspberry en DecaWave verbinden	Bram en Garben	JUNGLEGROEN
Power supply	Robbe en Xavier	ROODORANJE
Controller aan drone bevestigen	Robbe en Xavier	ROODORANJE
UWB	Bram en Xavier	GEEL
Connectie met server	Garben en Robbe	FUCHSIA
Aansturing drone	Garben en Robbe	FUCHSIA
Full-mesh	Garben en Xavier	SEPIA
Presentatie 1	Bram en Garben	JUNGLEGROEN
Presentatie 2	Robbe en Xavier	ROODORANJE
Verslag	Iedereen	WIT
Eindpresentatie	Iedereen	WIT

Tabel 2.1: Taakverdeling.

1	2	3	4	5	6	7	8	9	10	11	12	13
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Figuur 2.2: Gantt chart van de finale planning.

Hoofdstuk 3

Hardware

In wat volgt, worden alle componenten uitvoerig besproken en wordt een verantwoording gegeven van de ontwerpskeuzes.

3.1 Ultra Wide Band

In theorie is het mogelijk om de drone op een gekende locatie te laten vertrekken en een voorgeprogrammeerde route mee te geven. In de praktijk zorgt dat zeker en vast voor problemen. Denk bijvoorbeeld maar aan een onbekend obstakel dat plots het pad van de drone kruist, of een ventilatieschacht die hem uit positie blaast. Daarom is het nodig dat z'n precieze lokatie in de ruimte op elk moment gekend is. Veelgebruikte localisatietools zoals gps, wifi en bluetooth zijn te onnauwkeurig voor deze toepassing. Als de drone tussen 2 rekken met een doorgang van 1 m moet kunnen vliegen, dan moet de localisatie veel nauwkeuriger gebeuren.

Ultra Wide Band komt deze noden tegemoet. Dit is een vrij recente techniek met een nauwkeurigheid in de grootteorde van 0.10 m, wat volstaat om de drone indoor te kunnen lokaliseren.

Om deze lokatiebepaling via UWB te kunnen doen zijn 2 verschillende hardwarecomponenten noodzakelijk, namelijk enkele anker nodes (die op gekende locaties in het magazijn worden opgehangen) en een mobiele tag (die als onderdeel van de controller op de drone wordt bevestigd). Over de mobiele tag is meer te vinden in sectie 3.2.1.

3.2 Controller

3.2.1 DecaWave DWM1001

De mobiele tag kan dan om de beurt de verschillende ankers aanspreken, en vragen hoe ver hij van hen verwijderd is. Wanneer er enkele afstanden gekend zijn, kan hij zijn locatie

bepalen ten opzichte van de ankers.

De DecaWave DWM1001 bevat dezelfde UWB chip als de Pozyx tag, maar is goedkooper, compacter, en lichter. Voor de gebruikte drone is het minieme gewichtsverschil niet echt een probleem. Men moet echter wel in het achterhoofd houden dat meer massa de stabiliteit en vliegminuten in negatieve zin beïnvloedt.

RX Peak Current: 154 mA RX Mean Current: 134 mA TX Peak Current 111 mA TX Mean Current 82 mA, 2.8 V-3.6 V, 0.55 W

3 g

3.2.2 Raspberry Pi Zero W

150 mA, 5.0 V, 0.75 W

9 g

3.2.3 Lithium-ion Polymeer Batterij en Power Supply

De Lithium-ion Polymeer Batterij (LiPo) moet de controller gedurende ongeveer een kwartier van stroom kunnen voorzien.

Een batterij met 150 mA h kan gedurende 15 min zo'n 600 mA aan de controller leveren, de controller heeft ongeveer 350 mA nodig.

3.7 V

$$\frac{1.30 \text{ W}}{3.7 \text{ V}} = 350 \text{ mA}$$

$$350 \text{ mA} * 0.25 \text{ h} = 87.5 \text{ mA h}$$

5 g

LiPo SHIM???

3.2.4 Totaal

Totaal verbruik: 1.30 W

Totaal gewicht: 25 g

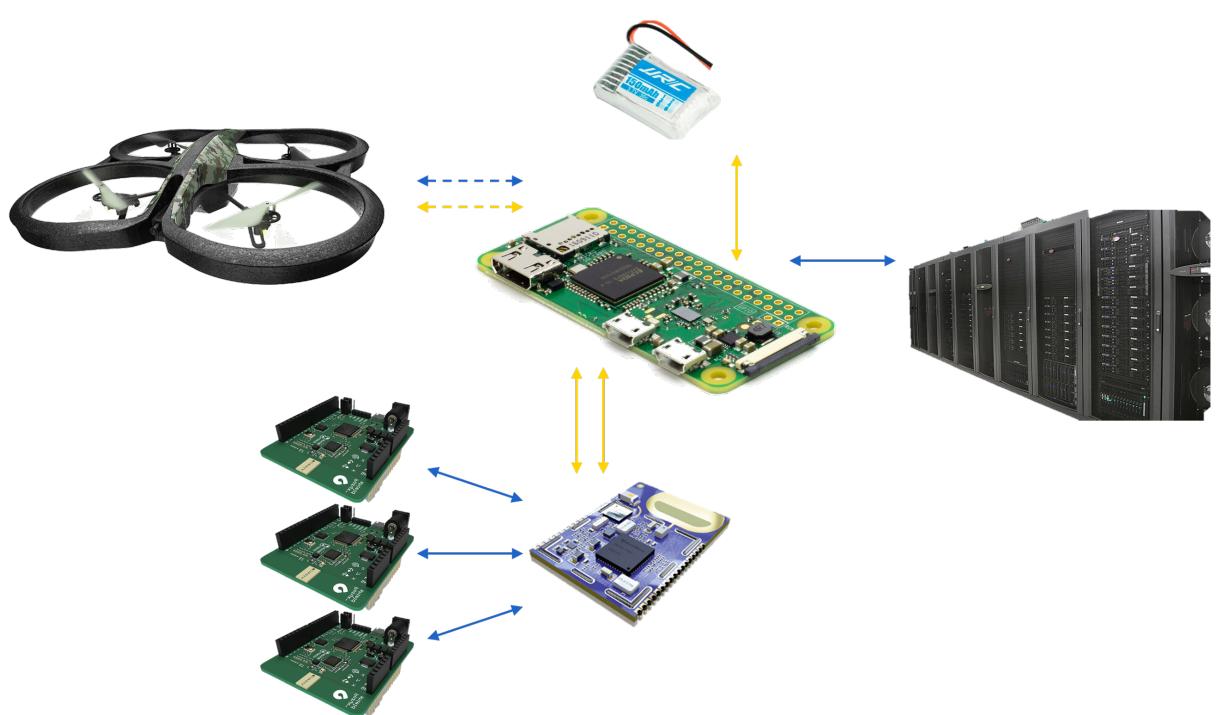
3.3 Drone

Parrot AR.Drone 2.0 Elite Edition

De camera op de drone zou kunnen dienen om barcodes in te scannen, maar dat onderdeel werd niet onder de doelstellingen van dit project gedefinieerd.

3.4 Setup

Op figuur 3.1 vindt u de hardware setup.



Figuur 3.1: Hardware setup.

Hoofdstuk 4

Software

In wat volgt, wordt het ontwerpproces van de software uitvoerig besproken en wordt een verantwoording gegeven van de ontwerpskeuzes.

4.1 Verbinding tussen de Ultra Wide Band Location Anchors en de Decawave

Ultra Wide Band (UWB) [1].

Wanneer de controller afstanden ontvangt tot gekende ankers, moeten deze nog verwerkt worden tot de correcte locatie. Hiervoor kunnen we gebruik maken van een Mosquitto server. Een Mosquitto server werkt via het MQTT protocol.

Een ander belangrijk aspect buiten nauwkeurigheid, is de snelheid. We zullen dus ook een algoritme moeten schrijven die bepaalt op welke ankers hij zich basseert. Het zou een beetje over-kill zijn als hij alle ankers in het volledige magazijn aanspreekt voor zijn locatie. Een mogelijke manier zou zijn dat we op voorhand bepalen dat hij gebruik maakt van het anker waar hij zich het dichtst bij bevindt en 3 ankers die zich in de buurt bevinden.

Nog een extra ontwerp die we in rekening moeten brengen is dat de server enkel 2D ondersteunt, wat dus niet voldoet voor een drone die op verschillende hoogtes kan vliegen. Gelukkig zit er in de drone een ultrasone sensor en een barometer ingebouwd die zijn hoogte kan bepalen.

4.2 Verbinding tussen de Decawave en de Raspberry Pi

I²C / TWI

4.3 Verbinding tussen de Raspberry Pi en de drone

De drone heeft een eigen wifi-netwerk met ESSID adrone2_xxx en geeft zichzelf vaak het IP-adres 192.168.1.1. Als de Raspberry Pi Zero W verbindt met het netwerk van de drone, krijgen het een IP-adres tussen 192.168.1.2 en 192.168.1.5 (met de grenzen inbegrepen) toegekend. Indien de drone een ander IP-adres aan zichzelf toegekend heeft, zullen de gebruikers één van de 4 volgende adressen toegekend krijgen. Het besturen van de drone gebeurd door het versturen van *AT commands* op UDP poort 5556. De frequentie waarmee de commando's moeten doorgestuurd worden ligt rond de 30 Hz of met een tussenperiode van ongeveer 30 ms, om de gebruiker een ervaring van voldoende hoge kwaliteit te voorzien. Wanneer er tussen 2 opeenvolgende commando's meer dan 2 s zitten, zal de AR Drone denken dat de verbinding verbroken is.

Informatie over de drone (zoals status, positie, snelheid, snelheid van de rotoren, ...) wordt naar de gebruiker gestuurd op UDP poort 5554. De frequentie waarmee deze *navdata* wordt verstuurd ligt tussen de 15 Hz (in demo mode) en 200 Hz in full (debug) mode. Om belangrijke data, zoals informatie voor de configuratie, te versturen maakt men geen gebruik van UDP, maar van TCP. Dit gebeurd via de *control port* 5559. [2]

Syntax van AT commands en navdata is terug te vinden in hoofdstuk 6 van ARDrone_Developer_Guide.pdf (Project → ARDrone_SDK_2_0_1 → Docs)!

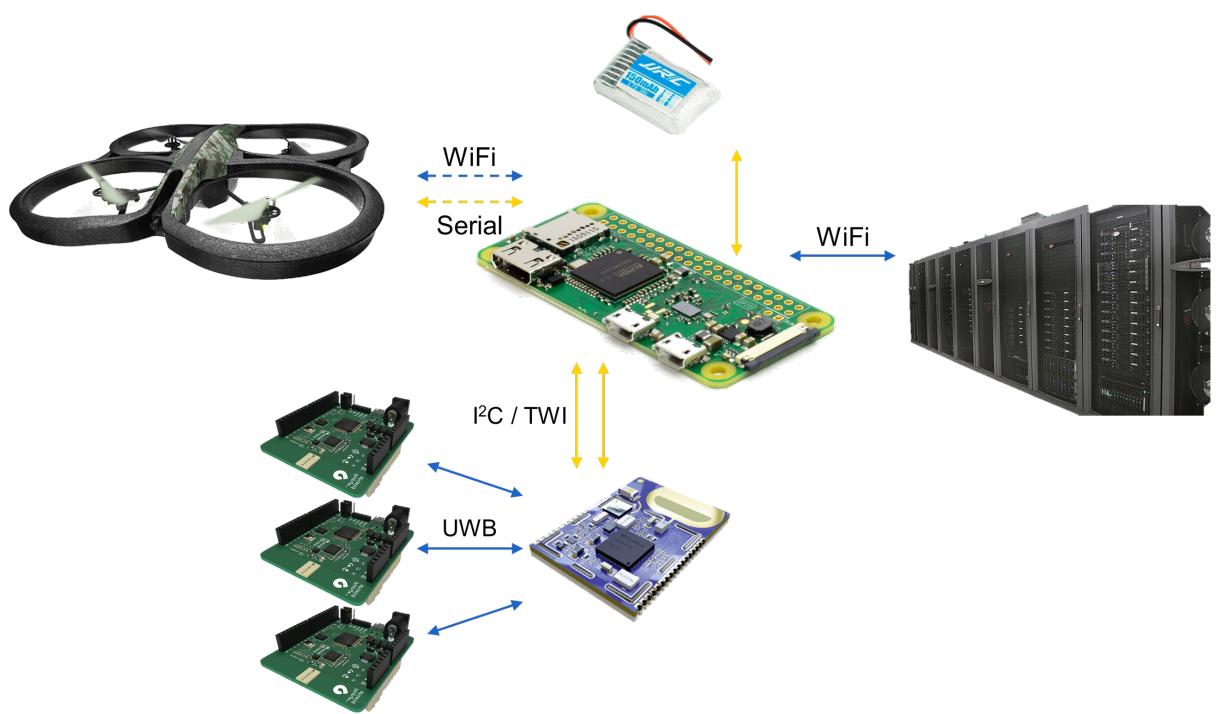
Configuratie van de drone is terug te vinden in hoofdstuk 8 van ARDrone_Developer_Guide.pdf (Project → ARDrone_SDK_2_0_1 → Docs)!

4.4 Verbinding tussen de Raspberry Pi en de server

4.5 Setup

Op figuur 4.1 vindt u de hardware setup, aangevuld met protocollen.

4.6 Full-mesh



Figuur 4.1: Hardware setup, aangevuld met protocollen.

Hoofdstuk 5

Resultaten

De resultaten van het project zijn ...

De openstaande problemen van het project zijn ...

De tekorten van het project zijn ...

Wat waren de foute beslissingen?

5.1 Kosten

Hier komt de verantwoording van gemaakte kosten...

Meer info in tabel 5.1

Product	Prijs (€)	Aantal	Totaal (€)
Parrot AR.Drone 2.0 Elite Edition	116.71	1	116.71
Micro USB OTG	1.32	2	2.64
LiPo batterijen en lader	12.47	1	12.47
Totaal			131.82

Tabel 5.1: Verantwoording van gemaakte kosten.

Referenties

- [1] A. Alarifi, A. Al-Salman, M. Alsaleh, A. Alnafessah, S. Al-Hadhrami, M. A. Al-Ammar, and H. S. Al-Khalifa, “Ultra wideband indoor positioning technologies: Analysis and recent advances,” *Sensors*, vol. 16, no. 5, p. 707, 2016.
- [2] S. Piskorski, N. Brulez, P. Eline, and F. D’Haeyer, “Developer Guide SDK 2.0,” p. 124, March 2012.