



**Amsterdam University
of Applied Sciences**

Sensornetwerken

ISO [DRAFT]

Auteurs: Tycho Jöbsis

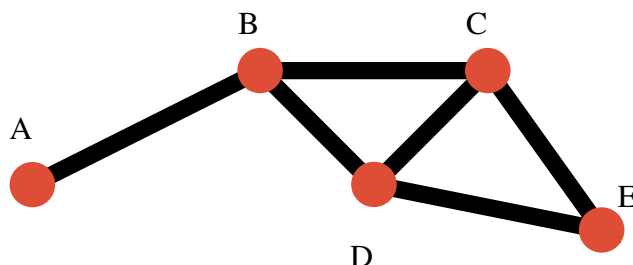
Datum: 11 oktober 2023

Inhoudsopgave

1	Nodes	3
2	Draadloze communicatie eisen	3
3	Netwerk ontdekking	3
3.1	Online komen	4
3.2	Snapshots	4
3.2.1	Via in snapshot laten vallen	4
3.2.2	Maximaal aantal hops	4
3.2.3	Snapshot packet opbouw	5
3.3	Updates	5
4	Berichten versturen en doorsturen	7
4.1	Data voor een ander	7
5	Timing	7
6	Adressen	7
A	Adres allocatie 25-09-2023	7

1 Nodes

Het netwerk bestaat uit verschillende node. Deze nodes kunnen in directe verbinding staan met elkaar of via een andere node. Dit is te zien in afbeelding 1. De nodes in het netwerk dat beschreven wordt



Figuur 1: Simpele graaf van een netwerk

door deze ISO communiceren draadloos met elkaar. De eisen die aan deze draadloze communicatie worden gesteld zullen verder worden toegelicht in sectie 2.

2 Draadloze communicatie eisen

De nodes in het netwerk moeten kunnen communiceren met een NRF24L01+ die is ingesteld met de volgende instellingen:

Setting	Instelling
Data snelheid	2Mbps
Kanaal	38
CRC	16 bits ¹

Suggestie

Voor het testen van een systeem wordt er aangeraden om op -18dBm te zenden.

3 Netwerk ontdekking

In deze sectie zal in worden gegaan op hoe een node voor het eerst online komt en hoe een node moet omgaan met een verandering in het netwerk. Een verandering kan zijn dat een connectie wordt verbroken of dat er een nieuwe connectie ontstaat. Al deze netwerk ontdekkingen zullen plaats vinden op de broadcast pipe.

¹Dit onder voorwaarde dat het niet gebruik maakt van de 32 byte dataframes die de NRF24L01+ kan versturen.

3.1 Online komen

Zodra een node online komt opent deze de pipes die in tabel 1 staan om uit te luisteren. Na een periode van ± 1 seconden na het aan gaan zal de node een snapshot versturen op de broadcast pipe. Op ontvangen berichten zal er geen auto acknowledge door de NRF24L01p worden gegeven².

Pipe	ADDR
Broadcast	'4' - '2' - '0' - 'B' - 0x00
Privé	'4' - '2' - '0' - 'P' - {ID}

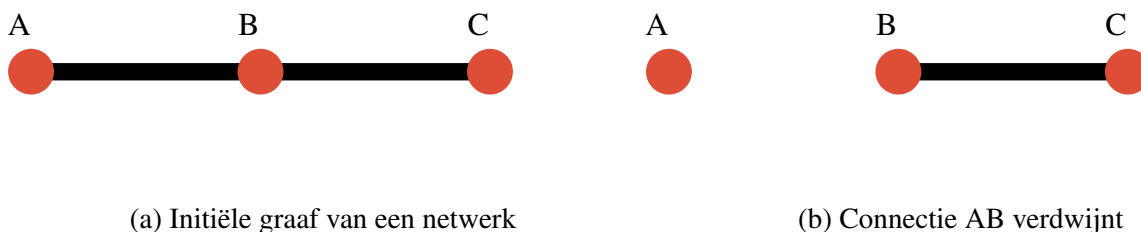
Tabel 1: Pipes en de IDs

3.2 Snapshots

Een snapshot is een manier om de routing die in een node bekend is te delen met andere nodes. Een snapshot kan uit een enkel packet bestaan of uit meerdere, afhankelijk van de hoeveelheid routing informatie die bekend is in de node. Wanneer een snapshot uit meerdere packets bestaat worden deze packets zo snel mogelijk achter elkaar verstuurd.

3.2.1 Via in snapshot laten vallen

Figuur 2a laat een eenvoudig netwerk bestaande uit drie nodes zien. Als vlak na dat B een snapshot heeft gestuurd de verbinding tussen A en B verdwijnt ontstaat de situatie die in figuur 2b is te zien. Zodra C in deze situatie een snapshot verstuurt zal hierin staan dat A via C in 1 stap te bereiken is. Om te voorkomen dat B en C samen gaan optellen elke keer dat ze een snapshot versturen zal B geen via data voor A accepteren totdat B een snapshot heeft gestuurd



Figuur 2: Voorbeeld van hoe een race conditie kan ontstaan

3.2.2 Maximaal aantal hops

In sectie 3.2.1 is er een methode beschreven om de kans op race condities in via data te verminderen. Deze methode werkt alleen op het moment dat er geen packet loss is. Om dit communicatie systeem robuster te maken voor packet loss wordt er een dynamisch maximaal aantal hops vastgesteld.

Een via entry in een snapshot zal als invalide worden beschouwd als het aantal hops van die via entry groter is dan het hop limiet. Het hop limiet wordt als volgt berekend: Aantal bekende nodes in het netwerk + 2.

²Kan worden aangepast op basis van resultaten van aankomende week.



3.2.3 Snapshot packet opbouw

byte	0	1	2	3	i	31
betekenis	My ID	Type	Via ID 1	Hops via ID 1	Via ID i	Hops via ID i

Tabel 2: Snapshot packet structuur

- **ID**

Deze byte bevat het ID van de zender.

- **Type**

Hieronder wordt beschreven welke informatie er in de **type** byte wordt doorgegeven.

bit	7	[6:0]
betekenis	U/S	Lengte

Tabel 3: Type

- **U/S**

Bepaal of het packet een snapshot of een update is. Voor een snapshot is dit bit 0 en voor een update is dit bit een 1.

- **Lengte**

Lengte geeft aan hoeveel via nodes er in de snapshot zijn.

- **Via ID n**

Bevat een node ID die via de node die de snapshot verstuurt te bereiken is.

- **Hops via ID n**

Geeft aan hoeveel andere nodes er tussen de node die de snapshot verstuurd en Via node n in zitten.

3.3 Updates

Een update kan worden gebruikt om aan te geven dat er iets in het netwerk is veranderd voordat er een snapshot wordt verstuurd. Een verandering in het netwerk kan een toevoeging of verwijdering van een node zijn.

byte	0	1	2	3	i
betekenis	Source ID	Type	Count	ID 1	ID i

Tabel 4: Update packet structuur

- **Source ID**

Het source ID is het ID van de node die als eerste de update verstuurd. Als in figuur 1 de connectie tussen A en B verdwijnt kan B een update versturen waarbij **Source ID** op B gezet kan worden. Als deze update vervolgens door C en D herhaalt worden blijft **Source ID** B.

- **Type**

Hieronder wordt beschreven welke informatie er in de type byte wordt doorgegeven.

bit	7	[6:0]
betekenis	U/S	Verval

Tabel 5: Type

- **U/S**

Bepaal of het packet een snapshot of een update is. Voor een snapshot is dit bit 0 en voor een update is dit bit een 1.

- **Verval**

Verval begint met een waarde van 5 waarvan elke keer dat een update wordt doorgestuurd 1 wordt afgehaald. Op het moment dat een node een update ontvangt waarvan de **Verval** waarde 0 is zal deze node de update niet nog een keer herhalen.

- **Count**

bit	[7:4]	[3:0]
betekenis	Toevoeging	Verwijdering

Tabel 6: Count

- **Toevoeging**

De waarde van **Toevoeging** geeft aan hoeveel Adressen zijn toegevoegd aan het netwerk. Het eerste adres dat is toegevoegd staat in byte 3 onder voorwaarde dat **Toevoeging** groter dan 0 is.

- **Verwijdering**

De waarde van **Verwijdering** geeft aan hoeveel Adressen zijn verwijderd in het netwerk. Het eerste adres dat is verwijderd staat in byte 3 + **Toevoeging** onder voorwaarde dat **Verwijdering** groter dan 0 is.

- **ID n**

ID n is het nde adres. Er kunnen maximaal 29 adressen worden meegegeven in een update.



4 Berichten versturen en doorsturen

In de vorige secties is besproken hoe de netwerk structuur wordt bij gehouden in het DWSN, in deze sectie zal in gegaan worden op hoe informatie wordt verstuurd in het DWSN.

Alle privé berichten worden over privé pipes gestuurd³ naar de doel node. Als deze node er niet is kan het bericht naar een volgende node die **Dest ID** in zo min mogelijk aantal stappen kan bereiken worden gestuurd.

byte	0	[1:31]
betekenis	Dest ID	Data

Tabel 7: Data bericht packet structuur

- **Dest ID**

Dest ID is het ID waar het bericht naar toe moet worden gestuurd.

- **Data**

Data is de data die naar de **Dest ID** node gestuurd wordt.

4.1 Data voor een ander

Als een node een data packet ontvangt waarvan de **Dest ID** niet zijn eigen ID is, zal de node dit data packet doorsturen naar een volgende node die **Dest ID** in zo min mogelijk aantal stappen kan bereiken. Als er geen nodes bekend zijn die **Dest ID** kunnen bereiken zal het packet vergeten worden en hoeft deze node niks meer te doen met het packet.

5 Timing

Nodes verzenden snapshots elke seconde. Er mogen maximaal 5 updates per node per snapshot worden verstuurd.

6 Adressen

Het adres van een node is een enkele byte. De enige uitzonderingen op valide adres waardes zijn 0x00 en 0xFF. In appendix A kan de adres allocatie gezien worden zoals deze is afgesproken op 25-09-2023.

A Adres allocatie 25-09-2023

³De privé pipes kunnen bepaald worden met tabel 1

eigenaar	begin adres	eind adres
Vrij	0x01	0x1F
Groep 1	0x20	0x3F
Groep 2	0x40	0x5F
Groep 3	0x60	0x7F
Groep 4	0x80	0x9F
Groep 5	0xA0	0xBF
Vrij	0xC0	0xDF
Vrij	0xE0	0xFE

Tabel 8: Uitgedeelde adres blokken