MeshCore

Complete Manual

Off-grid mesh communication

Text-based communication via LoRa radio

Version 2.0 - January 2026

By PE1HVH / ChatGPT / Claude.ai

Table of Contents

Foreword

My introduction to MeshCore coincided with my retirement a month ago. So I had all the time to dive into this. Which was also necessary, as I knew nothing about this at all.

Using the Internet and AI tools such as ChatGPT (OpenAI), Claude (Anthropic) and Perplexity.ai, I delved into the subject matter. While these tools helped me structure information, translate technical concepts into understandable language, and consistently format the document, the content choices and practical experience are my own.

Ik ben begonnen met het aankopen van SensCap Solar Node, SensCap T1000e en de Lilygo T-Deck.
In mijn zoektocht naar de informatie - die veelal te vinden is op internet, maar wel heel versnippert - is hier samengebracht. Samen met de ervaringen van het installeren en de problemen waar ik tegen aanliep is het geworden tot dit document.

Disclaimer

The AI tools used can hallucinate: this means they present information with great confidence that is incorrect. This is a well-known phenomenon in AI, caused by limitations in training data and statistical patterns that do not represent factual knowledge, or are simply wrong.

Although checking is done, errors may still occur. When in doubt, always consult the official MeshCore documentation. MeshCore is still actively in development. Information in this manual may therefore be outdated. If you have found errors, suggestions for improvements, or want to contribute, send an email.

73 de PE1HVH

Januari 2026

meshcore@pe1hvh.nl

Reading Guide

This manual is divided into four parts, each with its own focus:

Part A: User Manual

In addition to background, it provides information on how to use MeshCore. From initial installation to practical applications, hardware choices and group communication. Start here if you want to get started right away.

Part B: Technology Explained

For those who want to understand how LoRa and MeshCore work under the hood. From bits to chirps, dechirp analysis, modulation, demodulation, packet structures and advanced features such as remote control. Technical but accessibly written.

Part C: Abbreviations and Terminology

Alfabetisch naslagwerk van alle technische termen en afkortingen die in deze handleiding voorkomen.

Part D: References and Sources

Comprehensive source list for further study: datasheets, academic papers, official documentation and educational resources.

You can read the manual from front to back, or jump directly to the relevant section. The table of contents and header on each page help you navigate.

PART A

User Manual

1. Introduction: What is MeshCore?

MeshCore - a combination of Mesh (network of interconnected devices) and Core - is a wireless communication system that operates completely independently of the internet, mobile networks and other central infrastructure.

Het combineert LoRa radio-technologie met geavanceerde mesh-netwerken om betrouwbare communicatie mogelijk te maken over afstanden van tientallen kilometers.

Important: MeshCore is a text-based communication system. It is specifically designed for sending text messages, GPS locations, telemetry data and control commands. Voice calls or video are not possible due to the limited bandwidth of LoRa radio.

1.1 What makes MeshCore unique?

Unlike traditional communication systems, MeshCore does not need a central server or access point. Each device in the network is simultaneously:

- A transmitter and receiver - for direct communication
- A router - that forwards messages for others
- An autonomous computer - that makes decisions without human intervention

This architecture makes the network robust: if one device fails, the network automatically finds an alternative route.

1.2 Who is MeshCore for?

- Families - who want to communicate with each other during travel or emergencies
- Clubs and associations - such as morse clubs or hiking groups
- Amateur radio operators - for digital experiments and emergency communication
- Outdoor enthusiasts - hikers, campers, adventurers

2. Origin and History

De technologische evolutie van radartechniek naar off-grid mesh-communicatie

2.1 Chirp Spread Spectrum (1940s)

De wortels van MeshCore liggen in Chirp Spread Spectrum (CSS), een modulatietechniek ontwikkeld in de jaren '40 voor radarapplicaties. CSS verspreidt een signaal over een brede bandbreedte door de frequentie lineair te laten stijgen of dalen (chirps). Dit maakt het signaal robuust tegen multipath-fading, interferentie en jamming.

Multipath fading A radio signal can reach the receiver via multiple paths: directly, but also via reflections from buildings, mountains or other objects. These signals arrive at slightly different times and can reinforce or cancel each other out (fading). CSS is robust against this because the frequency changes continuously - a reflection that arrives a bit later has a different frequency and therefore does not interfere destructively with the direct signal.

Interferentie Other radio sources on the same or nearby frequencies can interfere with your signal. Think of WiFi, other LoRa transmitters, or industrial equipment. CSS spreads the signal over a wide bandwidth and a long time. A short interfering transmitter only affects a small part of your chirp, and the FFT can still reconstruct the symbol from the remaining samples.

Jamming Intentional interference by a transmitter continuously emitting noise or a strong signal on your frequency. CSS is difficult to jam because:

- Je moet de hele 64 kHz (in NL) bandbreedte tegelijk verstoren (niet één frequentie)
- De processing gain zorgt dat je signaal nog steeds detecteerbaar is zolang de jammer niet extreem sterk is
- Different spreading factors are orthogonal, so a jammer on SF7 does not disturb SF12

Orthogonal betekent in dit verband "onafhankelijk" of "niet-interfererend".
For a more detailed explanation see section 14.4

2.2 LoRa: from idea to chip (2009-2012)

In 2009 begonnen twee Franse ingenieurs, Nicolas Sornin en Olivier Seller, met de ontwikkeling van een long-range, low-power modulatietechniek gebaseerd op CSS. In 2010 voegde François Sforza zich bij hen en richtten ze samen Cycleo op in Frankrijk. In mei 2012 werd Cycleo overgenomen door Semtech Corporation, die de technologie commercialiseerde onder de merknaam LoRa (Long Range).

2.3 Accessible hardware (2016-2018)

De echte doorbraak voor hobbyisten kwam met de combinatie van Semtech's LoRa-chips (SX1276/SX1262) en Espressif's ESP32-microcontroller. Fabrikanten als Heltec en LILYGO brachten geïntegreerde development boards op de markt voor ~$20-30.

2.4 Meshtastic: the first mesh wave (2019-2020)

In 2019, American software enginee Kevin Hester (GitHub: geeksville) started the Meshtastic project. Meshtastic uses a flooding mesh protocol where each device forwards messages. In large networks, this led to congestion.

2.5 MeshCore: intelligent routing (2024-2025)

Eind 2024 begon de Australische ontwikkelaar Scott Powell (Ripple Radios) aan een nieuw protocol. Begin 2025 lanceerde hij samen met Andy Kirby (UK) en Liam Cottle (NZ) het MeshCore-project met:

- Hybride routing: eerste contact via flood, daarna geleerde routes voor efficiëntie
- Role separation: Companion Radios, Repeaters and Room Servers as separate functions
- Scalability: up to 64 hops, state-aware network, AES-128 encryption
- Lightweight C++: geen dynamische geheugenallocatie, embedded-first design

2.6 Timeline

Year	Milestone	Meaning
1940s	CSS ontwikkeld voor radar	Basis modulatietechniek
2009	Sornin & Seller start LoRa development	CSS voor datacommunicatie
2010	Cycleo opgericht (Frankrijk)	Commercialisatie begint
2012	Semtech acquires Cycleo	LoRa brand born
2015	LoRa Alliance opgericht	LoRaWAN standard
2016+	ESP32 + LoRa boards (Heltec e.a.)	Hardware toegankelijk
2019	Kevin Hester start Meshtastic	Eerste off-grid mesh
2024	Scott Powell starts MeshCore	Intelligent routing
2025	MeshCore publicly launched	Scalable architecture

3. Node Types (Firmware Profiles)

MeshCore distinguishes different node types based on their function in the network. These types are determined by the firmware profile you flash.
Deze worden in de volgende paragraven beschreven.

3.1 Companion Radio

This is the most commonly used type for end users. The node acts as a radio interface for a smartphone via Bluetooth (BLE), USB or WiFi. The MeshCore Companion App controls the node.

3.2 Repeater

A node configured as a repeater has one primary task: forwarding messages to extend the network range. Typically placed at strategic locations with good antenna position such as roof edges or tall buildings. Works fully autonomously without smartphone.

3.3 Room Server

A Room Server manages one or more Rooms and provides extensive store-and-forward functionality. Stores messages for offline recipients. Requires Ultra license for remote management.

3.4 Standalone Device

Some hardware such as the T-Deck Plus can function completely independently without an external smartphone. With built-in screen and keyboard you can type and read messages directly.

3.5 Telemetry Node

Een node specifiek geconfigureerd voor het verzenden van sensordata zoals temperatuur, luchtvochtigheid, batterijspanning. Uitbreidbaar via GPIO, I²C of SPI interfaces.

3.6 Typical network

In de praktijk kan een netwerk bestaan uit een combinatie van deze node types. Een typisch familienetwerk zou kunnen bestaan uit:

- 2-4 Companion Radios for family members
- 1 Repeater at a high point for better coverage
- 1 Room Server at home for store-and-forward

4. Getting Started: First Installation

4.1 Requirements

- MeshCore-compatible hardware (e.g. Heltec V3, T-Deck Plus)
- Antenna for 868 MHz
- Android or iOS phone with MeshCore Companion App
- USB cable for initial configuration

4.2 Step 1: Flash firmware

Go to flasher.meshcore.co.uk in Chrome or Edge. Select your device, choose 'Companion Radio BLE' firmware, and click Flash.

4.3 Step 2: Connect with the app

Start the MeshCore Companion App, choose Scan, select your node and tap Connect.

4.4 Step 3: Configure

Set: frequency 868.300 MHz, bandwidth 125 kHz, spreading factor SF11, power 14 dBm (EU limit). Encryption is enabled by default.

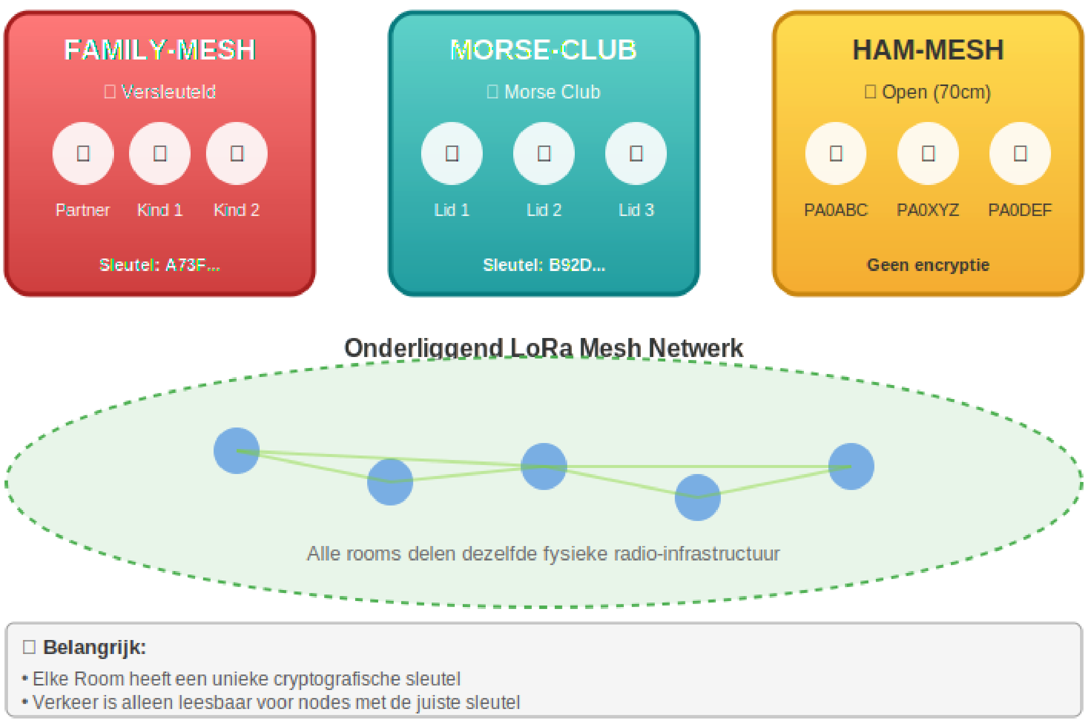
5. Groepscommunicatie: Channels en Room Servers

MeshCore has two ways to communicate in groups: Channels and Room Servers. These are fundamentally different concepts.

Channel - A shared cryptographic key (PSK for AES-256 encryption). Nodes with the same key can read each other's messages. There is no central server or member list. Messages are real-time: miss it, and it's gone.

Room Server - A physical node with server firmware that works as a BBS (Bulletin Board System). Users log in with a password. Messages are stored (store-and-forward) and you can come back later to retrieve missed messages (up to 32 messages).

MeshCore Rooms - Versleutelde Groepen



5.1 Channel-typen

Public Channel (#public)

Het standaard kanaal dat automatisch wordt toegevoegd bij elke MeshCore installatie. Dit is het "marktplaats" kanaal waar iedereen op luistert. Handig voor eerste contact maken, maar geen privacy.

Hashtag Channel (#naam)

Community-kanalen voor specifieke onderwerpen of regio's, zoals #switzerland, #berlin, of #morsecode. De sleutel wordt berekend uit de naam, dus iedereen die de naam kent kan meeluisteren. Handig voor open communities.

Private Channel (own key)

A channel with a self-chosen, random key that you only share with the intended participants. This provides true privacy - only those who have the key can listen.

Type	Key	Privacy	Use
#public	Universally known	Geen	Algemene communicatie
#hashtag	Berekend uit naam	Geen (afleidbaar)	Community, regio, onderwerp
Private	Self-chosen, secret	Full	Familie, werk, gevoelig

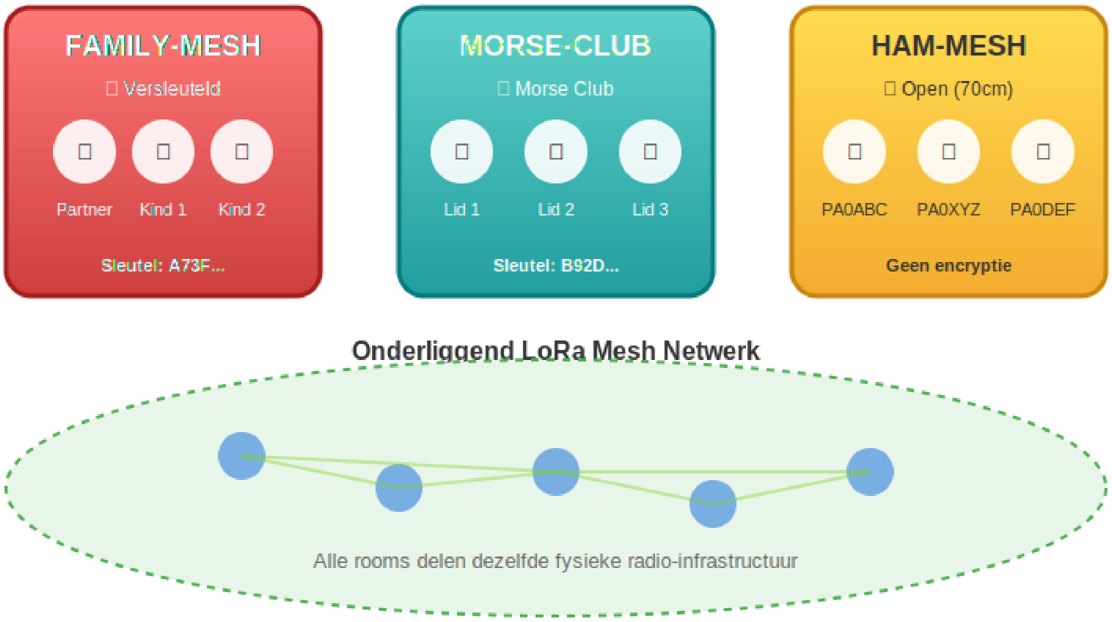
5.2 Room Servers

A Room Server is a node that is always online and offers additional features for reliable communication:

- Store-and-forward: messages are stored until the recipient comes online
- Ledenlijst: je ziet wie er in de Room zit
- Beheer: moderators kunnen leden toevoegen/verwijderen
- Persistence: up to 32 messages are stored

Functie	Channel	Room Server
Send messages	Alleen naar online nodes	Also store for offline nodes
Store-and-forward	Beperkt (alleen in transit)	Full (long-term storage)
Bereikbaarheid	Depends on participants	24/7 via fixed node
Beheer op afstand	No	Ja (met Ultra-licentie)

MeshCore Rooms - Versleutelde Groepen



Belangrijk:
- Elke Room heeft een unieke cryptografische sleutel
- Verkeer is alleen leesbaar voor nodes met de juiste sleutel

6. Direct Messages (DM)

In addition to group communication via Rooms, MeshCore also supports Direct Messages (DMs): private messages between two specific nodes. DMs are end-to-end encrypted and can only be read by the sender and receiver.

6.1 Verschil tussen Rooms en DM's

Aspect	Room (Public)	Direct Message
Requirement	Lid van dezelfde Room	Receiver's public key
Encryption	Shared Room key	End-to-end (public key)
Key-uitwisseling	Via QR code or Room Server	Via advertisements (beacons)

6.2 Hoe DM's werken

- Node A broadcasts an advert with its public key
- Node B receives the advert and stores the public key
- Node B can now send an encrypted DM to Node A
- For two-way traffic, Node B must also broadcast an advertisement

Belangrijk: Adverts worden NIET doorgestuurd door repeaters. Beide nodes moeten elkaar direct kunnen "horen" voor de key-uitwisseling.

6.3 Troubleshooting DM's

Problem	Solution
DM fails after 3 retries	Controleer of beide nodes elkaars advert hebben ontvangen. Zet "Auto-add nodes to contacts" aan.
Node verschijnt niet in "Heard" lijst	Adverts worden niet geforward. Zorg voor direct bereik tussen de nodes.
Contact is in list but DM fails	Contact may have been added via public message. Delete and have re-added via advert.
Public messages work, DM does not	Public goes via Room, DM requires public key. Send adverts and wait 15 seconds.

7. Privacy en Beveiliging

7.1 What is always visible?

Elke node moet beacons uitzenden voor routing. Andere nodes zien dat er een node bestaat en actief is.

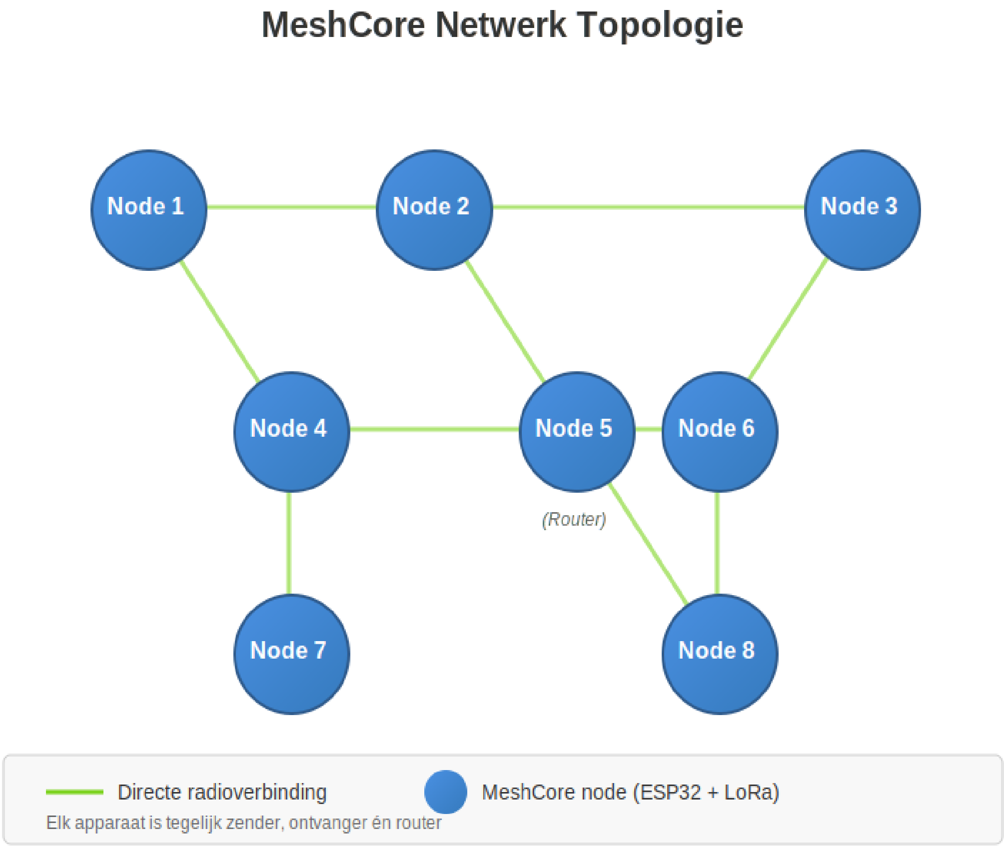
7.2 Wat is NOOIT zichtbaar (ISM-modus)?

- Who you communicate with
- Which Rooms you are in
- The content of your messages
- Even the existence of your Rooms

7.3 HAM vs ISM

Property	HAM-modus	ISM-modus
Frequency	70 cm (430-440 MHz)	868 MHz
License	Required	Not required
Encryption	Not allowed	Required
Identification	Callsign	Anonymous possible

8. Practical Applications



8.1 Familie-mesh

All family members have a MeshCore device in Room 'FAMILY-MESH'. Use: short messages, GPS tracking, appointments. Works without mobile network on vacations, festivals or in remote areas.

8.2 Morse-club

Clubleden coördineren CW-activiteiten, velddagen en spotmeldingen via een gezamenlijke Room. Leden zien elkaar op de kaart en kunnen real-time QSO-informatie delen.

8.3 Amateur Radio Mesh

On the 70 cm band with callsigns. Digital QSOs, emergency communication, self-routing repeaters. Each node identifies itself with callsign according to amateur regulations.

8.4 Remote Station

Node on roof terrace with good antenna, node inside with BLE to phone. Control your radio infrastructure via your own mesh. No VPN or internet connection needed.

9. Hardware Overzicht

MeshCore supports a wide range of hardware devices. The data below is taken from various internet sources. In my limited knowledge built up in a short time, I can say that the Standalone T-Deck is not very suitable for family use in our case.
The combination of companion app with the SensCap T1000e was experienced as much more user-friendly.

9.1 Hardware Categorieën

- Companion Radios - require smartphone for control via Bluetooth/USB/WiFi
- Standalone Devices - complete communication devices with keyboard and screen
- Repeaters and Servers - nodes that extend the network

9.2 LilyGO T-Deck Plus (€70-80)

Best for: Standalone use without smartphone

Complete standalone oplossing met ESP32-S3, 2.8 inch LCD kleurenscherm, fysiek QWERTY toetsenbord, ingebouwde GPS, trackball navigatie en SMA antenneconnector.

Advantages: Fully independent, physical keyboard, robust housing, GPS built-in, external antenna possible.

Disadvantages: Trackball sometimes sensitive, reset button can be accidentally pressed.

9.3 Heltec WiFi LoRa 32 V3/V4 (€20-40)

Best for: Budget entry with smartphone

Goedkoopste manier om met MeshCore te starten. ESP32-S3 met SX1262 LoRa, klein 0.96 inch OLED scherm. V4 heeft hoger zendvermogen (28dBm).

Advantages: Very affordable, compact, widely supported, suitable as companion and repeater.

Disadvantages: Requires smartphone, small screen, no housing standard.

9.4 RAK WisBlock RAK4631 (€40-60)

Best for: Energy-efficient and modular solutions

Gebaseerd op nRF52840 met extreem laag stroomverbruik - weken tot maanden op batterij. Modulair systeem met uitbreidingsmodules voor GPS, sensoren, etc.

Advantages: Extremely low consumption, modular, perfect for solar, professional quality.

Disadvantages: More expensive, no WiFi (Bluetooth only), more complex setup.

9.5 Seeed Studio T1000-E (€30-40)

Best for: Compact companion radio with GPS

Extreem compact creditcard-formaat met ingebouwde GPS. nRF52840 gebaseerd voor laag verbruik.

Advantages: Very compact, built-in GPS, low consumption, robust.

Disadvantages: No display or buttons, requires smartphone for all interaction.

9.6 Hardware Vergelijkingstabel

Specificatie	T-Deck Plus	Heltec V3/V4	RAK4631	T1000-E
Price	€70-80	€20-40	€40-60	€30-40
Standalone	Yes	No	No	No
Display	Yes	Yes	No	No
GPS ingebouwd	Yes	No	Optional	Yes
Batterijduur	2-3 days	1-2 days	Weeks	Weeks
Externe antenne	Ja (SMA)	Ja (IPEX)	Ja (U.FL)	No
Best voor	Standalone	Budget	Solar/IoT	Compact/GPS

Note: Always check that the device has the correct frequency for Europe (868 MHz, not 915 MHz).

9.7 Offline Kaarten op T-Deck

De T-Deck Plus kan offline kaarten weergeven via een microSD-kaart. Dit is handig voor navigatie zonder internetverbinding. Op internet vond ik de volgende gegevens.

9.7.1 Benodigdheden

- microSD card (max 32 GB, formatted as FAT32)
- Computer met Python 3 geïnstalleerd
- Internetverbinding voor het downloaden van kaarten

9.7.2 De verschillende methoden

Method 1: Map Tiles Downloader (recommended)
De eenvoudigste methode is de Map Tiles Downloader van OM7TEK:

Windows: Open PowerShell and run:

- pip install pipx
- python -m pipx ensurepath
- pipx install mt-downloader
- mt-downloader

macOS: Open Terminal and run:

- brew install pipx
- pipx ensurepath
- pipx install mt-downloader
- mt-downloader

Het programma toont een tekstinterface waar je continenten, landen en regio's kunt selecteren. De tegels worden gedownload naar een 'tiles' map.

Method 2: tdeck-maps script
Alternatively you can use the tdeck-maps Python script:

- Installeer dependencies: pip install pillow requests
- Clone repository: git clone https://github.com/JustDr00py/tdeck-maps.git
- Download tegels voor een stad:

python3 meshtastic_tiles.py --city "Amsterdam, Netherlands" --min-zoom 8 --max-zoom 12

Method 3: Buy ready-made maps
Pre-downloaded map packages are available at https://buymeacoffee.com/ripplebiz for Europe and other regions. This supports the developer and saves time.

Method 4: Download ready-made maps with higher zoom level.
For this method you need a lot of patience.
Via the URL below you can download all of Europe with a very high zoom level, note that copying to SD card can take more than a day.

https://www.reddit.com/r/meshtastic/comments/1j1chem/meshtastic_26_map_tiles_with_higher_zoom_levels/.

9.7.3 Installatie op SD-kaart

- Formatteer de microSD-kaart als FAT32 (niet exFAT)
- Kopieer de 'tiles' map naar de root van de SD-kaart
- De mapstructuur moet zijn: /tiles/[zoomlevel]/[x]/[y].png
- Werp de SD-kaart veilig uit
- Insert the SD card into the T-Deck (on the left side)
- Herstart de T-Deck

9.7.4 Kaart gebruiken

- Navigate to the Maps screen via the map icon
- Use the trackball or touchscreen to pan
- Zoom in/out with the 'W' and 'S' keys or the buttons on the screen
- Tap the cross to return to your current location

Let op: De SD-kaart moet geplaatst zijn vóór het inschakelen van de T-Deck.

9.7.5 Zoomniveaus

Zoom level	Detail	Use	Storage per region
8-10	Laag	Large region overview	Klein (~50 MB)
11-12	Gemiddeld	Cities and roads	Gemiddeld (~200 MB)
13-14	Hoog	Street level	Groot (~1 GB+)

Tip: Start with zoom level 8-12 for a good balance between detail and storage space. Deeper zoom (13+) requires an Ultra license.

10. Useful Links and Resources

10.1 Official Websites

- https://meshcore.co.uk/ - Officiële website
- https://github.com/meshcore-dev/MeshCore - GitHub broncode

10.2 Web Tools

- https://flasher.meshcore.co.uk/ - Web Flasher voor firmware
- https://analyzer.letsmesh.net/ - Packet Analyzer
- https://meshcore.nz/ - Web Companion Client
- https://map.meshcore.dev/ - Officiële MeshCore kaart

10.3 Apps

- https://meshcore.co.uk/apps.html - Overzicht alle apps
- MeshCore Companion App - Android en iOS

10.4 Community

- https://discord.gg/ZVH2ujy9ex - MeshCore Discord server
- https://forum.meshcore-net.nl/ - Nederlandstalig forum

11. Open Source and Paid Licenses

MeshCore is an open-source project. The core software is freely available under the MIT license, but some advanced features require a paid license.

11.1 What is open-source?

De open-source kern van MeshCore omvat: mesh-routing, end-to-end encryptie, room-systemen, basis mesh-functionaliteit. Iedereen kan de broncode bekijken, aanpassen en eigen builds maken.

11.2 MeshCore Ultra Software License (£8)

De MeshCore store verkoopt unlock-licenties voor extra functies op bepaalde apparaten. De Ultra-licentie kost £8 (eenmalig) en ontgrendelt:

- Higher zoom levels on maps
- Remote Repeater profile (node as fixed repeater)
- Room server admin (manage rooms remotely)
- Telemetry node (send/receive sensor data)
- Store-and-forward (store messages and deliver later)

11.3 What always remains free?

- Full mesh routing
- Encrypted chat
- Rooms and group communication
- Basic GPS
- LoRa networks
- Custom firmware builds

PART B

Technology Explained

12. From Text to Chirp

Hoe LoRa data codeert - een stap-voor-stap uitleg van bits naar radiosignaal

12.1 Je wilt "Test" versturen

Laten we beginnen met iets concreets: je wilt het woord "Test" versturen via LoRa. Hoe wordt dat een radiosignaal dat kilometers ver kan reizen?

Letters worden bits

Elke letter heeft een ASCII-code, en die code is een getal dat we als bits kunnen schrijven:

Letter	ASCII	Binair
T	84	01010100
e	101	01100101
s	115	01110011
t	116	01110100

Samen is "Test" dus 32 bits.

12.2 Bits become symbols

Now the Spreading Factor (SF) comes into play. At SF12 we group bits in groups of 12.

Why 12 bits?

SF12 means: each symbol carries 12 bits of information. Those 12 bits together form a number from 0 to 4095 (because 2^{12} = 4096 possible values).

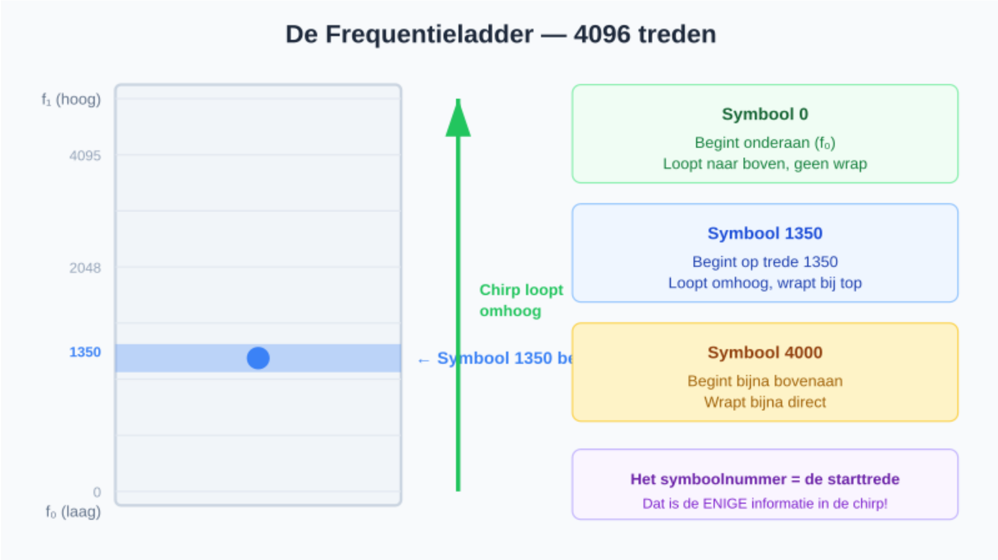
We pakken onze 32 bits en verdelen ze in groepjes van 12. Het woord "Test" wordt zo drie symbolen: 1350, 1395 en 1860.

12.3 Symbols become chirps

Now the crucial step: how does a number (for example 1350) become a radio signal?

De frequentieladder

Imagine the 125 kHz bandwidth as a ladder with 4096 rungs. Each symbol number corresponds to a starting position on that ladder.



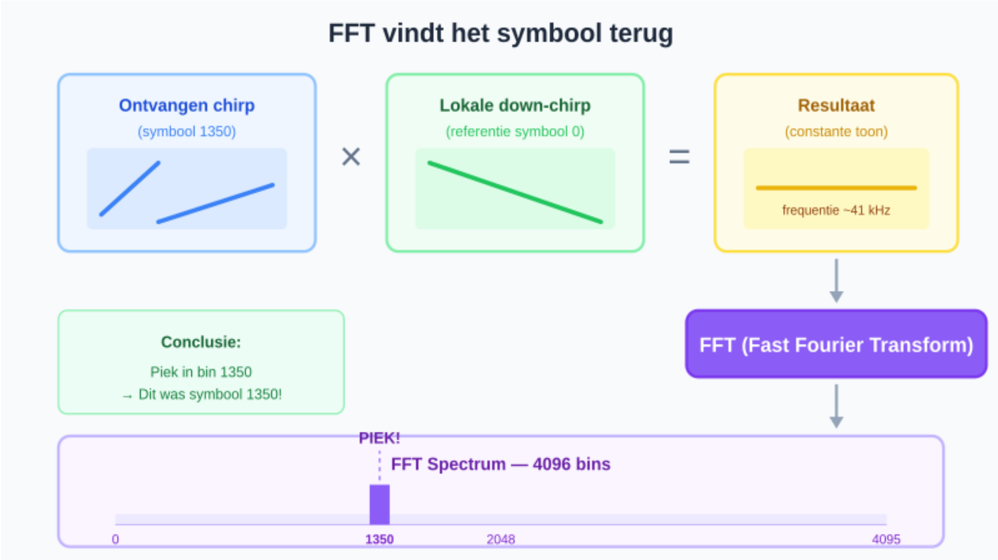
De chirp loopt de ladder op

Een chirp begint op zijn startpositie en loopt dan alle treden af, omhoog. Bij de top wrapt hij naar beneden en gaat verder tot hij weer bij zijn startpunt is.

12.4 How does the receiver know which symbol it was?

De ontvanger doet een slimme wiskundige truc: hij vermenigvuldigt de ontvangen chirp met een lokaal gegenereerde down-chirp (dalende frequentie).

Rising frequency × falling frequency = constant tone. The pitch of that tone depends on where the original chirp started.



De FFT (Fast Fourier Transform) analyseert de toon en geeft een spectrum met 4096 bins. De bin waar de energie geconcentreerd is = het symboolnummer.

Fouttolerantie: Processing Gain

De kracht van LoRa zit in redundantie. Bij SF12 worden 12 bits verspreid over 4096 frequentiestappen. Dit is 341× meer bandbreedte dan strikt nodig. Deze "processing gain" (~36 dB) maakt detectie onder de ruisvloer mogelijk.

12.5 Summary

De complete keten van tekst naar radiosignaal:

Written;

- Text → ASCII → Bits
- Bits → Groepjes van SF bits → Symbolen (0-4095)
- Symbol → Start position on frequency ladder
- Chirp doorloopt alle treden, wrappend bij de top
- Receiver multiplies with down-chirp → constant tone
- FFT vindt de piek → symboolnummer teruggevonden

13. Dechirp Analysis

Waarom de simpele tabel faalt en hoe FFT piekdetectie werkt

13.1 Introduction

Deze analyse hoort bij een poging om beter te begrijpen waarom LoRa zo goed werkt. Het doel is om op een eenvoudige manier kenbaar te maken wat het dechirp proces doet en hoe de FFT daaruit het symbool haalt.

Om het duidelijk te houden, wordt een sterk vereenvoudigd voorbeeld gebruikt: een "bandbreedte" van 10 frequentiestappen (0-9), waarbij elk sample 1 seconde duurt. In de praktijk zijn de tijden veel korter (milliseconden) en het aantal samples per symbool veel groter (128-4096), maar het principe blijft hetzelfde.

13.2 The problem with the simple table

Bij mijn eerste poging om LoRa dechirp met een eenvoudige som-tabel te laten zien, gaat het mis zodra er een wrap in de frequentie optreedt. Eerst het geval zonder verschuiving.

Symbol 0 (no shift) - Works

Hier lopen de TX (up-chirp) en RX (down-chirp) netjes tegenover elkaar, en de som blijft constant.

TX (up)	RX (down)	Som
0	10	10
1	9	10
2	8	10
3	7	10
4	6	10
5	5	10
6	4	10
7	3	10
8	2	10
9	1	10

✓ Constante som = 10. Hier werkt het idee "constante som = goed".

Symbol 3 (shifted) - Fails after wrap

Then the same table for the shifted version (symbol 3):

TX (start=3)	RX (down)	Som - 10	Status
3	10	3	✓
4	9	3	✓
5	8	3	✓
6	7	3	✓
7	6	3	✓
8	5	3	✓
9	4	3	✓
0 (wrap)	3	-7	✗ FOUT
1	2	-7	✗ FOUT
2	1	-7	✗ FOUT

After the wrap, the calculation no longer works in this simple approach.

13.3 Why wrapping is necessary

The frequency band is physically limited. In this example we use 10 steps (0-9), in practice for example 64 kHz bandwidth with a fixed channel width.

De frequentie mag niet buiten de band omdat:

- Het buiten het toegewezen spectrum zou vallen (illegaal)
- Het andere diensten zou verstoren
- De ontvanger het signaal dan niet kan volgen

Daarom loopt de tijd/positie netjes door (0-9), maar springt de frequentie terug naar 0 wanneer hij voorbij 9 zou gaan: dat is de wrap.

Belangrijk: de wrap is een praktische beperking van de frequentieband, niet een echte "reset" van de onderliggende fase-ontwikkeling van het signaal.

13.4 The correct interpretation

De oplossing is om naar het verschil te kijken in plaats van naar de som, en om de TX-frequentie na de wrap wiskundig door te laten lopen. Dan blijft het verschil constant.

Position	TX freq	TX (wiskundig)	RX ref	Difference
0	3	3	0	3
1	4	4	1	3
2	5	5	2	3
3	6	6	3	3
4	7	7	4	3
5	8	8	5	3
6	9	9	6	3
7	0 (wrap)	10	7	3
8	1	11	8	3
9	2	12	9	3

Het verschil blijft constant = 3, ongeacht de wrap!

Door de TX-frequentie "door te tellen" (10, 11, 12, ...) zie je dat het verschil met de RX-ref overal 3 blijft, ook na de wrap. Dat constante verschil is precies wat de FFT als piek terugvindt.

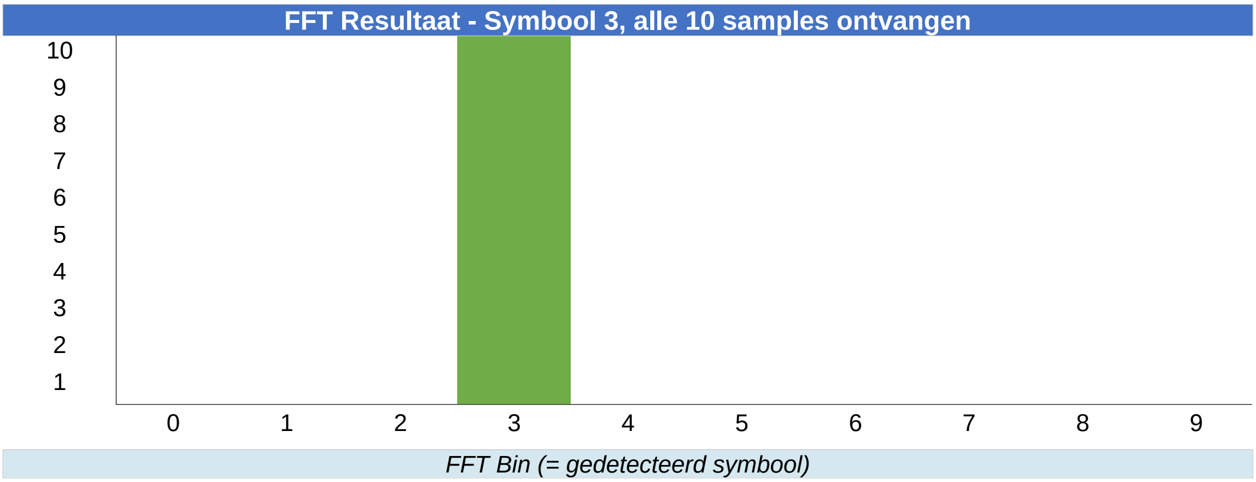
Note: this is a simplified representation intended to make the working principle of dechirp and FFT peak detection intuitive; the actual LoRa implementation uses a more complex, but mathematically equivalent description.

13.5 FFT peak detection - Graphical representation

De FFT telt in feite hoe vaak elk mogelijk verschil voorkomt; elk verschil komt in een eigen bin terecht. Voor symbool 3 leveren alle 10 samples verschil 3 op, dus stapelt alle energie zich in bin 3.

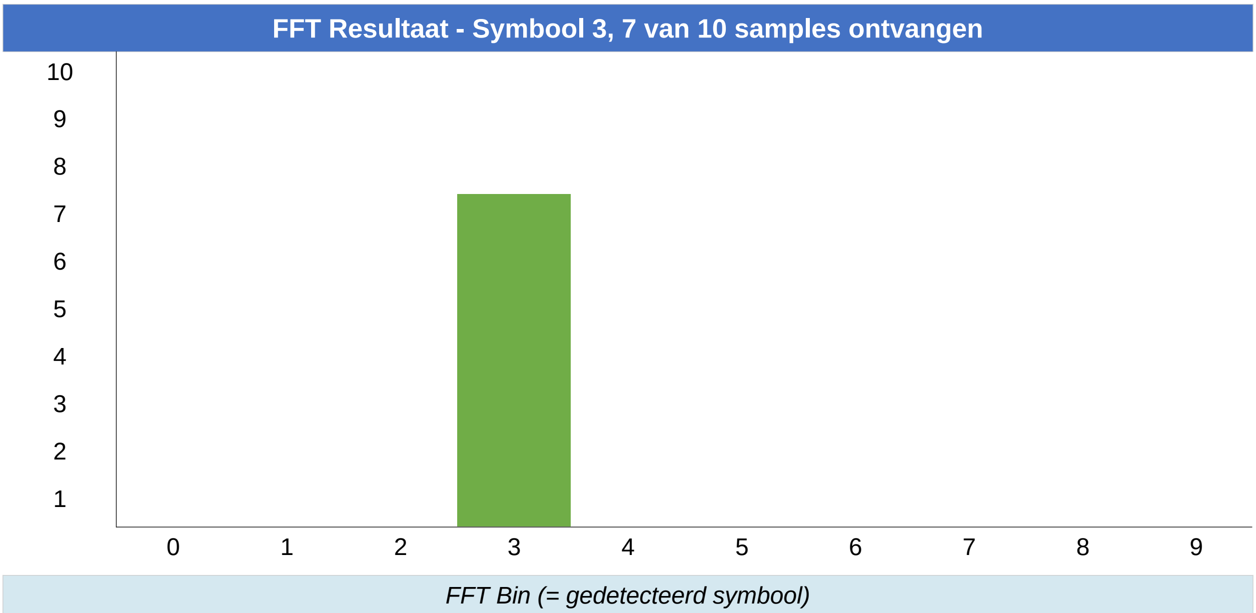
Een bin is letterlijk een "bakje" waarin de energie voor dat specifieke frequentieverschil wordt opgeteld.

Ideaal signaal (geen verlies)



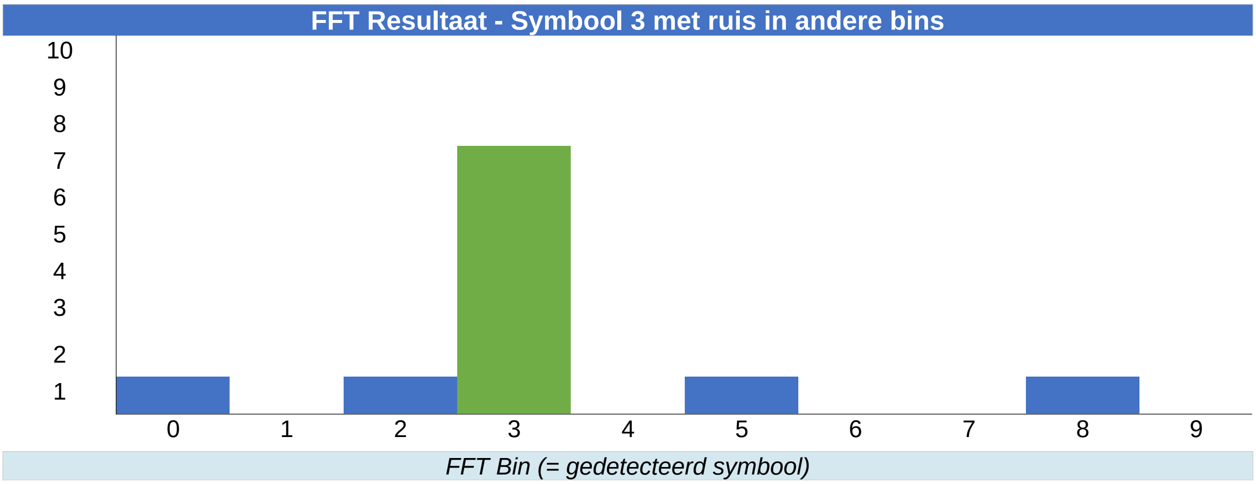
- Alle 10 samples dragen bij aan bin 3
- Bin 3 krijgt daardoor een piekhoogte van 10, de andere bins blijven laag

With 30% sample loss



Even with 3 missed samples, the peak at bin 3 remains dominant. The symbol is correctly detected.

With noise/interference



Noise and errors land in random bins. They are spread out and cannot exceed the signal peak.

13.6 The core principle

SIGNAAL	RUIS
Alle samples → dezelfde bin	Fouten → willekeurige bins
= GECONCENTREERDE ENERGIE	= VERSPREIDE ENERGIE

Daarom kan incidentele vervuiling de signaalpiek nooit overstijgen.

13.7 How many samples can you miss?

De fouttolerantie hangt af van de Spreading Factor:

Spreading Factor	Samples per symbol	~30% loss tolerable
SF7	128	~38 samples
SF10	1024	~307 samples
SF12	4096	~1229 samples

Daarnaast voegt de Coding Rate (CR) nog extra foutcorrectie toe:

Coding Rate	Overhead	Fouttolerantie
CR 4/5	25%	Basis
CR 4/6	50%	Moderate
CR 4/7	75%	Goed
CR 4/8	100%	Maximum

13.8 Processing gain - The power of LoRa

De "magie" van LoRa zit in de processing gain: door het signaal te spreiden over vele samples, kun je signalen detecteren die onder de ruisvloer liggen.

In our example with 10 samples:

- Signal energy concentrates in 1 bin
- Noise distributes over 10 bins
- Processing gain ≈ 10× (10 dB)

Bij SF12 met 4096 samples: processing gain ≈ 4096× (36 dB)!

Dit verklaart waarom LoRa verbindingen kan maken die met conventionele radio onmogelijk zouden zijn.

13.9 Conclusion

De oorspronkelijke tabel-aanpak mislukte omdat:

- Er met een som werd gewerkt in plaats van met een verschil
- De wrap werd gezien als een echte reset, in plaats van een praktische beperking van de frequentieband
- Instantane frequenties werden vergeleken, terwijl LoRa werkt met faseverschillen die via de FFT als frequentieverschil zichtbaar worden

De FFT ziet het constante verschil als een duidelijke piek:

- Alle juiste samples stapelen zich in dezelfde bin (signaal)
- Noise and errors are spread across other bins
- De piek van het signaal blijft daardoor dominant, zelfs bij aanzienlijk sample-verlies

In MeshCore, the FFT is used as a decision maker: the bin with the highest energy is the decoded symbol. This keeps the implementation relatively simple, scalable and robust against errors.

14. LoRa Modulation

Van OSI-model tot radiosignaal - CSS, Spreading Factors en Demodulatie

14.1 LoRa in the network layers

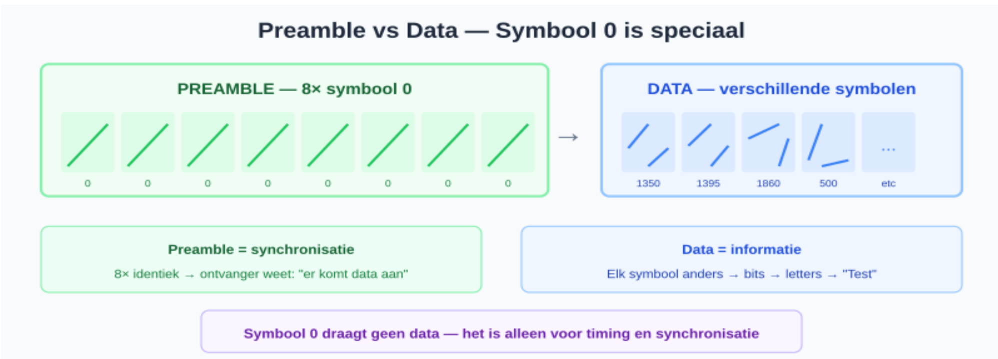
LoRa and MeshCore can be fitted into the well-known OSI model:

OSI layer	LoRa/MeshCore equivalent
7. Applicatie	MeshCore Companion App, chat, GPS
6. Presentatie	— (not present)
5. Sessie	— (no persistent sessions)
4. Transport	MeshCore ACKs, retries, fragmentation
3. Network	MeshCore routing, hops, Room addressing
2. Datalink	LoRa packet: preamble, sync word, header, CRC
1. Physical	LoRa PHY: chirps, SF, BW, the radio itself

14.2 Synchronization without handshake

Er is geen handshaking bij LoRa. De zender zendt, en wie luistert, luistert. Synchronisatie gebeurt via de preamble - identieke chirps aan het begin van elk packet.

Symbol 0 is the reference - a chirp that starts at the lowest frequency and runs neatly upward. The preamble consists of 8× symbol 0, followed by the data.



What the receiver learns from the preamble

Informatie	Hoe
Timing	Chirps komen op regelmatige intervallen
Frequentie-offset	Preamble chirps are symbol 0, deviation = drift
SF confirmation	Chirp-lengte matcht verwachte SF
Netwerk-ID	Sync word must match

14.3 Demodulation: down-chirp mixing

De ontvanger vermenigvuldigt het ontvangen signaal met een lokaal gegenereerde down-chirp (dalende frequentie). Dit proces heet dechirping.

Demodulatie stappen

- 1. Receive up-chirp with unknown start frequency
- 2. Multiply with local down-chirp
- 3. Result: constant tone (frequency depends on start position)
- 4. Apply FFT to the dechirped signal
- 5. Vind de piek in het spectrum
- 6. Piekpositie = symboolwaarde

14.4 SF Orthogonality

Different spreading factors are (almost) orthogonal: an SF10 receiver ignores SF12 signals, and vice versa. This is because the chirp duration differs per SF:

SF	Symbolen	Chirp-duur (BW=125kHz)	Bits/symbool
SF7	128	1.02 ms	7
SF8	256	2.05 ms	8
SF9	512	4.10 ms	9
SF10	1024	8.19 ms	10
SF11	2048	16.38 ms	11
SF12	4096	32.77 ms	12

14.5 Chirp Spread Spectrum (CSS)

LoRa uses Chirp Spread Spectrum (CSS): a modulation technique where the frequency runs linearly through the bandwidth. This makes the signal very robust.

Why CSS is robust

- Multipath: reflections are shifted in time, not in frequency evolution
- Interferentie: andere signalen correleren niet met de chirp
- Doppler: frequentieverschuiving verschuift de hele chirp, niet de vorm

14.6 The Encoding Pipeline

LoRa has a multiple encoding pipeline that converts data to chirps:

- 1. Data → Whitening (pseudo-random XOR to prevent DC bias)
- 2. Whitened data → Hamming FEC (add error correction)
- 3. FEC codewords → Interleaving (spread bits over symbols)
- 4. Interleaved data → Gray coding (minimize bit errors)
- 5. Gray-coded symbolen → Chirp modulatie
- 6. Preamble + Sync + Header + Payload + CRC → Packet

- 7. Packet → Radiosignaal

14.7 Bits, Symbols and Chips

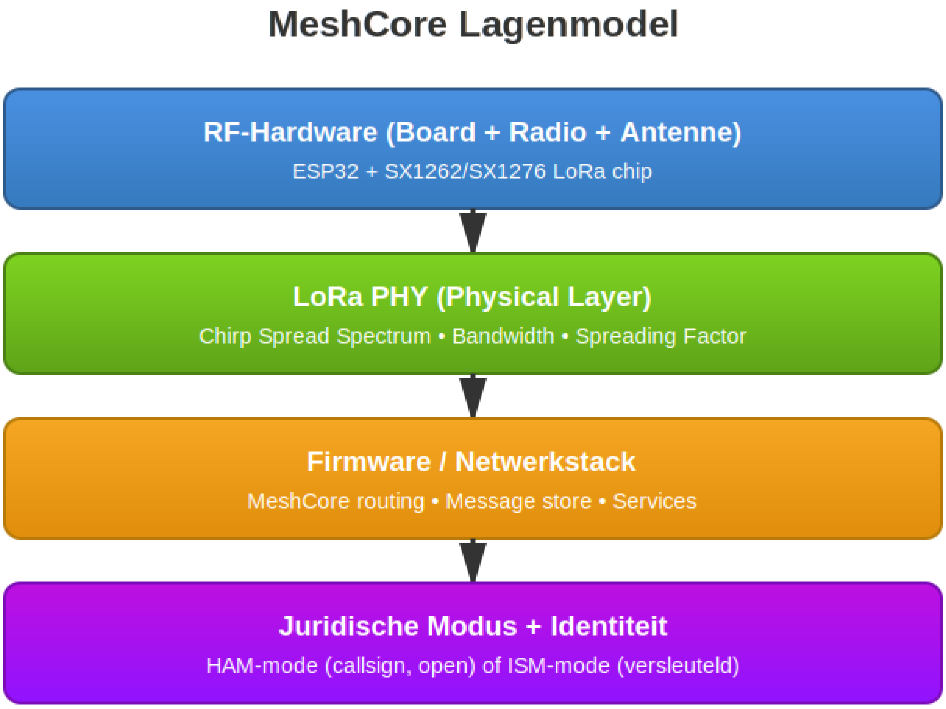
In spread spectrum terminologie:

- Bit: de kleinste informatie-eenheid
- Symbol: a group of SF bits (e.g. 12 bits at SF12)
- Chip: één frequentiestap in de chirp (2^SF chips per symbool)

Bij SF12 worden 12 bits verspreid over 4096 chips. Dit geeft een processing gain van ~21 dB, waardoor signalen onder de ruisvloer nog decodeerbaar zijn.

15. The Layer Model of MeshCore

MeshCore is built from four strictly separated layers, each with a specific function:



15.1 Layer 1: RF hardware

Microcontroller (ESP32/nRF52), LoRa chip (SX1262), antenna, power supply.

15.2 Layer 2: LoRa PHY

Modem layer: bits → radio signals via CSS. BW, SF and CR determine range vs speed.

15.3 Layer 3: Firmware and Network Stack

MeshCore firmware handles routing, message handling and network management. Each node has its own identity (4-byte Node-ID) and can learn routes to other nodes. Intelligent routing instead of pure flooding.

15.4 Layer 4: Legal mode

Bepaalt of het apparaat draait in HAM-modus (70 cm-band, geen encryptie, callsign verplicht) of ISM-modus (868 MHz, encryptie, anoniem mogelijk).

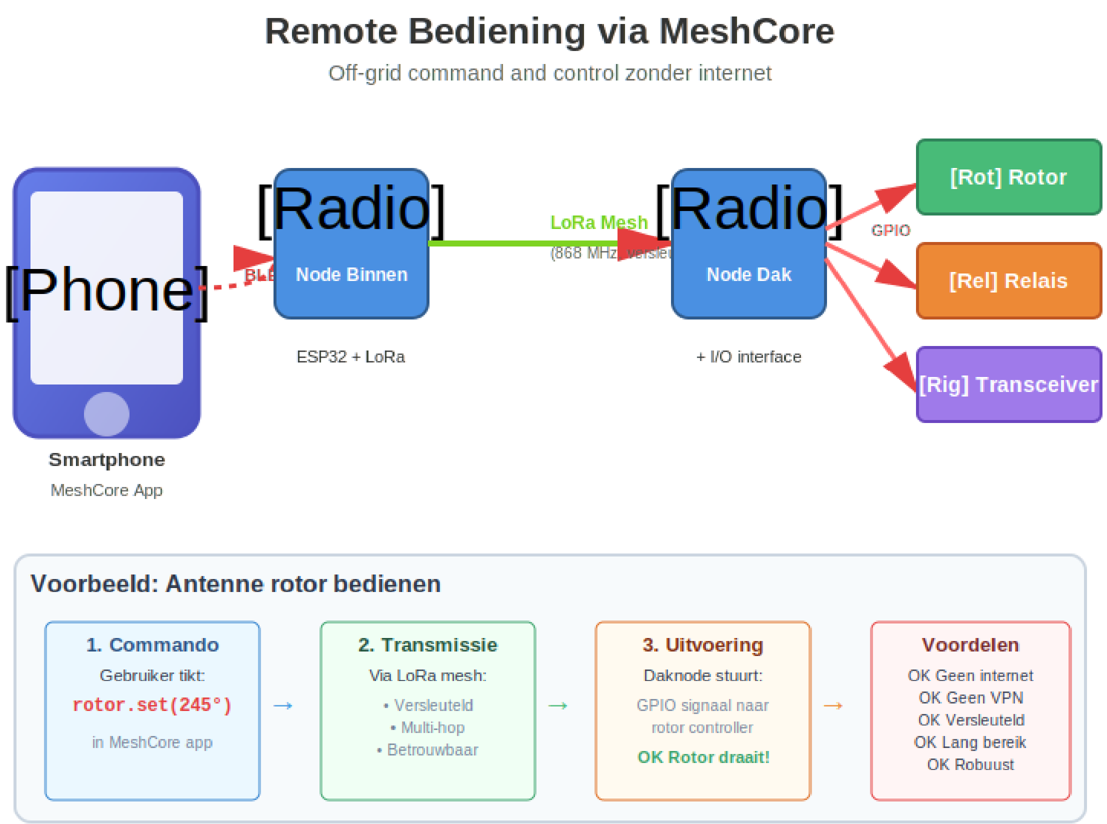
15.5 Link Budget - Why LoRa reaches so far

Link budget is het totaal aan signaalverlies dat een verbinding kan verdragen en toch nog te decoderen. LoRa haalt 150+ dB link budget, waardoor:

- Long distances possible with low power
- Connections below the noise floor can still be decodable
- Meshes with few nodes can already scale regionally (via hops)

16. Advanced Features: Remote Control

MeshCore is not just a chat system. It can be used as a fully off-grid, encrypted command-and-control network.



16.1 Technically speaking

Elke MeshCore-node is tegelijkertijd:

- A LoRa radio - for wireless communication
- A router - for mesh network functions
- A computer (ESP32) - with processing power
- With I/O interfaces - UART, GPIO, I2C, SPI, USB

Dit betekent dat je letterlijk apparaten kunt aanhangen:

- Relays - for switchable power
- Arduinos and Raspberry Pis - for more complex tasks
- Transceivers - via CAT interface
- Sensors - temperature, humidity, air pressure
- Motors and rotors - for antenna direction
- Power switches - for remote on/off

16.2 How does it work?

MeshCore has a service layer on top of the mesh. A node can offer services such as gpio.toggle, uart.send or i2c.read. Other nodes can call these services via encrypted messages.

16.3 Concrete example: Roof terrace

Je zet een node bij je antennes op het dak. Die node zit in Room 'REMOTE-MESH' en heeft GPIO verbindingen met je antennerotator en transceiver. Binnen heb je een node met BLE verbonden aan je telefoon.

Commando versturen: Phone → BLE → binnennode → mesh → daknode → GPIO → rotor draait. Geen wifi nodig, geen internetprovider, geen cloud.

16.4 Why is this so powerful?

- Cannot be jammed with a single transmitter
- No central point that can fail
- No cloud needed
- Fully encrypted
- Tens of kilometers range

17. MeshCore Packet Structure

17.1 General Packet Structure

Dit document beschrijft twee lagen die samenwerken:

LoRa Fysieke Laag	Radio hardware (SX1262/SX1276)	Preamble, Sync Word, MIC, CRC
MeshCore Protocol	MeshCore firmware (ESP32/nRF52)	Header + Payload (alle andere velden)

De LoRa chip voegt automatisch Preamble/Sync Word/CRC toe. MeshCore software verwerkt alleen Header en Payload.

Elk MeshCore-pakket volgt een vaste structuur die bestaat uit de volgende onderdelen:

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Synchronization for radio receiver
Sync Word	2	`12 34`	Network identifier (configurable)
Header	12-16	`(zie details)`	Routing and type information
Payload	0-200	`(variabel)`	Message content (type-dependent)
MIC	4	`XX XX XX XX`	Message Integrity Code (auth)
CRC	2	`XX XX`	Foutdetectie (CRC-16)

Total maximum packet size: 232 bytes

17.2 Header structure

De header bevat essentiële routing- en type-informatie voor elk pakket:

Field	Bytes	Hex Example	Description
Version	1	`01`	Protocol version (0x01)
Flags	1	`00-FF`	Packet flags (encrypt, ack, etc.)
Type	1	`01-06`	Message type identifier
Hop Count	1	`00-0F`	Aantal hops (max 15)
Packet ID	4	`4F 3A 2B 1C`	Unique packet identifier
Source ID	4	`DE AD BE EF`	Sender Node-ID
Dest ID	4	`CA FE BA BE`	Receiver Node-ID (or FF FF FF FF)

Flags byte breakdown:

```
Bit 0: Encrypted (1=ja)
Bit 1: ACK Required (1=ja)
Bit 2: Is ACK (1=dit is een ACK)
Bit 3: Is Broadcast (1=broadcast)
Bit 4-7: Reserved
```

Type values:

```
0x01 = TEXT_MESSAGE (text/chat)
0x02 = POSITION (GPS location)
0x03 = TELEMETRY (sensor data)
0x04 = NODEINFO (node identification)
0x05 = ROUTING (route discovery)
0x06 = ACK (acknowledgment)
0x07 = BEACON (periodic announcement)
0x10 = COMMAND (remote control)
```

17.3 Message types

Code	Type	Encryption	Routing
0x01	TEXT_MESSAGE	E2E / Room	Unicast / Room
0x02	POSITION	Optional	Broadcast
0x03	TELEMETRY	Optional	Broadcast
0x04	NODEINFO	No	Broadcast
0x05	ROUTING	No	Broadcast
0x06	ACK	E2E	Unicast
0x07	BEACON	No	Broadcast
0x10	COMMAND	E2E (verplicht)	Unicast

17.4. The text messages explained

17.4.1 TEXT_MESSAGE(0x01)

Text messages for user communication. Can be private (encrypted) or public.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`03`	Encrypted + ACK required
Type	1	`01`	TEXT_MESSAGE
Hop Count	1	`00`	Eerste hop
Packet ID	4	`4F 3A 2B 1C`	Unieke ID
Source ID	4	`DE AD BE EF`	Sender node
Dest ID	4	`CA FE BA BE`	Receiver node
Room ID	4	`00 00 00 00`	Room (0=DM)
Timestamp	4	`65 A1 B2 C3`	Unix timestamp
Text Length	1	`10`	Payload length (16 bytes)
Text Data	var	`44 69 74 20 69 73 ...`	"Dit is een test!"
MIC	4	`AB CD EF 12`	Authenticatie
CRC	2	`34 56`	Checksum

Complete Hex Dump:

```
AA AA AA AA AA AA AA AA 12 34 01 03 01 00 4F 3A 2B 1C DE AD BE EF CA FE BA BE 00 00
00 00 65 A1 B2 C3 10 44 69 74 20 69 73 20 65 65 6E 20 74 65 73 74 21 AB CD EF 12 34
56
```

17.4.2 POSITION(0x02)

GPS-coördinaten van een node. Wordt automatisch broadcast op ingesteld interval.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`08`	Broadcast, no ACK
Type	1	`02`	POSITION
Hop Count	1	`00`	Eerste hop
Packet ID	4	`12 34 56 78`	Unieke ID
Source ID	4	`DE AD BE EF`	Sender node
Dest ID	4	`FF FF FF FF`	Broadcast
Latitude	4	`42 50 24 DD`	52.0705° (float)
Longitude	4	`40 89 9A 44`	4.3007° (float)
Altitude	2	`00 05`	5 meter
Precision	1	`0A`	GPS HDOP ×10
Timestamp	4	`65 A1 B2 C3`	Unix timestamp
Speed	2	`00 00`	Speed km/h
Heading	2	`00 B4`	Direction 180°
MIC	4	`78 9A BC DE`	Authenticatie
CRC	2	`F0 12`	Checksum

Complete Hex Dump:

```
AA AA AA AA AA AA AA AA 12 34 01 08 02 00 12 34 56 78 DE AD BE EF FF FF FF FF 42 50
24 DD 40 89 9A 44 00 05 0A 65 A1 B2 C3 00 00 00 B4 78 9A BC DE F0 12
```

17.4.3 TEXT_MESSAGE(0x03)

Sensor data such as battery percentage, temperature and voltage.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`08`	Broadcast
Type	1	`03`	TELEMETRY
Hop Count	1	`00`	Eerste hop
Packet ID	4	`AB CD EF 01`	Unieke ID
Source ID	4	`DE AD BE EF`	Sender node
Dest ID	4	`FF FF FF FF`	Broadcast
Battery %	1	`57`	87% batterij
Voltage	2	`0E D8`	3.80V (×1000)
Temperature	2	`00 DC`	22.0°C (×10)
Humidity	1	`41`	65% RH
Pressure	2	`27 D6`	1018.2 hPa (×10)
Uptime	4	`00 01 51 80`	86400 sec (1 dag)
TX Count	2	`00 C8`	200 transmissions
RX Count	2	`01 2C`	300 received
MIC	4	`11 22 33 44`	Authenticatie
CRC	2	`55 66`	Checksum

Complete Hex Dump:

```
AA AA AA AA AA AA AA AA 12 34 01 08 03 00 AB CD EF 01 DE AD BE EF FF FF FF FF 57 0E
D8 00 DC 41 27 D6 00 01 51 80 00 C8 01 2C 11 22 33 44 55 66
```

17.4.4 NODEINFO(0x04)

Information about the node: name, hardware model and firmware version.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`08`	Broadcast
Type	1	`04`	NODEINFO
Hop Count	1	`00`	Eerste hop
Packet ID	4	`FE DC BA 98`	Unieke ID
Source ID	4	`DE AD BE EF`	Sender node
Dest ID	4	`FF FF FF FF`	Broadcast
Name Len	1	`08`	Name length (8)
Node Name	var	`4E 6F 64 65 2D 41 42 43`	"Node-ABC"
Model ID	2	`00 03`	Heltec V3 (model 3)
FW Major	1	`01`	Firmware v1
FW Minor	1	`02`	.2
FW Patch	1	`03`	.3 → v1.2.3
Capabilities	2	`00 1F`	Feature flags
Public Key	32	`04 AB CD ... (32 bytes)`	Ed25519 pubkey
MIC	4	`AA BB CC DD`	Authenticatie
CRC	2	`EE FF`	Checksum

Model ID values:

```
0x0001 = T-Deck Plus
0x0002 = Heltec V2
0x0003 = Heltec V3
0x0004 = RAK4631
0x0005 = T1000-E
0x0006 = T-Beam
```

17.4.5 ROUTING(0x05)

Routing messages for network discovery and route table updates.

Field	Bytes	Hex Example	Description
Preamble	8	AA AA AA AA AA AA AA AA	Radio synchronization
Sync Word	2	12 34	MeshCore netwerk ID
Version	1	01	Protocol v1
Flags	1	08	Broadcast
Type	1	05	ROUTING
Hop Count	1	00	Eerste hop
Packet ID	4	11 22 33 44	Unieke ID
Source ID	4	DE AD BE EF	Sender node
Dest ID	4	FF FF FF FF	Broadcast
Route Type	1	01	Route announce
Neighbor Cnt	1	03	3 neighbors
Neighbor 1 ID	4	11 11 11 11	Buur node 1
Neighbor 1 RSSI	1	C8	-56 dBm (signed)
Neighbor 1 SNR	1	0A	10 dB SNR
Neighbor 2 ID	4	22 22 22 22	Buur node 2
Neighbor 2 RSSI	1	B0	-80 dBm
Neighbor 2 SNR	1	05	5 dB SNR
Neighbor 3 ID	4	33 33 33 33	Buur node 3
Neighbor 3 RSSI	1	9C	-100 dBm
Neighbor 3 SNR	1	02	2 dB SNR
MIC	4	12 34 56 78	Authenticatie
CRC	2	9A BC	Checksum

Route Type values:

```
0x01 = Route Announce (periodic)        0x03 = Route Reply (response)
0x02 = Route Request (find route)       0x04 = Route Error (route broken)
```

17.4.6 ACK(0x06)

Acknowledgement: confirmation that a message was received.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`04`	Is ACK flag
Type	1	`06`	ACK
Hop Count	1	`00`	Eerste hop
Packet ID	4	`99 88 77 66`	ACK packet ID
Source ID	4	`CA FE BA BE`	ACK zender (orig ontvanger)
Dest ID	4	`DE AD BE EF`	ACK ontvanger (orig zender)
ACK'd Pkt ID	4	`4F 3A 2B 1C`	Original packet ID
Status	1	`00`	0=OK, 1=Error
MIC	4	`DE AD C0 DE`	Authenticatie
CRC	2	`FA CE`	Checksum

Complete Hex Dump:

```
AA AA AA AA AA AA AA AA 12 34 01 04 06 00 99 88 77 66 CA FE BA BE DE AD BE EF 4F 3A
2B 1C 00 DE AD C0 DE FA CE
```

17.4.7 ROOM MESSAGE

Encrypted group message to all Room members via Room Key.

Field	Bytes	Hex Example	Description
Preamble	8	AA AA AA AA AA AA AA AA	Radio synchronization
Sync Word	2	12 34	MeshCore netwerk ID
Version	1	01	Protocol v1
Flags	1	09	Encrypted + Broadcast
Type	1	01	TEXT_MESSAGE
Hop Count	1	00	Eerste hop
Packet ID	4	AA BB CC DD	Unieke ID
Source ID	4	DE AD BE EF	Sender node
Dest ID	4	FF FF FF FF	Broadcast (Room filtered)
Room ID	4	D0 CW A0 01	DOCWA Room ID
Timestamp	4	65 A1 B2 C3	Unix timestamp
Nonce	12	XX XX ... (12 bytes)	Encryptie nonce
Enc Payload	var	(encrypted)	Encrypted text
Auth Tag	16	XX XX ... (16 bytes)	AES-GCM auth tag
MIC	4	12 34 56 78	Packet authentication
CRC	2	9A BC	Checksum

Encryptie: AES-256-GCM met Room Key

17.4.8 COMMAND MESSAGE

Command messages for remote control of devices via GPIO/UART/I2C.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`03`	Encrypted + ACK required
Type	1	`10`	COMMAND (0x10)
Hop Count	1	`00`	Eerste hop
Packet ID	4	`C0 MM AA ND`	Unieke ID
Source ID	4	`DE AD BE EF`	Command zender
Dest ID	4	`DA K0 DE 01`	Target node (roof)
Service ID	2	`00 01`	GPIO service
Command	2	`00 02`	SET command
Target	1	`05`	GPIO pin 5
Value Len	1	`02`	Value length
Value	var	`00 F5`	245° (rotor position)
Nonce	12	`XX XX ...`	Encryptie nonce
Auth Tag	16	`XX XX ...`	AES-GCM auth
MIC	4	`CO DE CA FE`	Packet auth
CRC	2	`12 34`	Checksum

Service IDs:

```
0x0001 = GPIO (pin control)
0x0002 = UART (serial passthrough)
0x0003 = I2C (sensor/device bus)
0x0004 = CAT (transceiver control)
0x0005 = POWER (relay/switch)
0x0006 = ROTOR (antenna rotation)
```

Command codes:

```
0x0001 = GET (read value)
0x0002 = SET (write value)
0x0003 = TOGGLE (flip state)
0x0004 = PULSE (momentary)
```

17.4.9 BEACON MESSAGE

Periodic announcement messages for mesh routing and node discovery.

Field	Bytes	Hex Example	Description
Preamble	8	`AA AA AA AA AA AA AA AA`	Radio synchronization
Sync Word	2	`12 34`	MeshCore netwerk ID
Version	1	`01`	Protocol v1
Flags	1	`08`	Broadcast, no encrypt
Type	1	`07`	BEACON (0x07)
Hop Count	1	`00`	Original beacon
Packet ID	4	`BE AC 0N 01`	Beacon sequence
Source ID	4	`DE AD BE EF`	Beacon node
Dest ID	4	`FF FF FF FF`	Broadcast
Interval	2	`01 2C`	300 sec (5 min)
TX Power	1	`0E`	14 dBm
Capabilities	2	`00 1F`	Feature flags
Uptime	4	`00 01 51 80`	Node uptime
Load	1	`20`	CPU load 32%
Queue	1	`02`	2 packets queued
MIC	4	`BE AC 0N ED`	Authenticatie
CRC	2	`FA ST`	Checksum

Capability flags:

```
Bit 0: GPS aanwezig
Bit 1: Telemetry sensoren
Bit 2: Repeater mode
Bit 3: Store & Forward
Bit 4: GPIO services
Bit 5-15: Reserved
```

PART C

Abbreviations and Terminology

18. Abbreviations and Terminology

Alfabetisch overzicht van alle technische termen en afkortingen.

Term	Meaning
70 cm-band	Amateur radio band 430-440 MHz, license required
868 MHz	ISM-band frequentie voor Europa
ACK	Acknowledgement - confirmation that a message was received
Advert/Beacon	Periodic signal for presence announcement and public key exchange
AES	Advanced Encryption Standard - versleutelingsalgoritme (128/256-bit)
BLE	Bluetooth Low Energy - energiezuinige verbinding tussen node en smartphone
BW	Bandwidth - bandbreedte in kHz (125/250/500), smaller = robuuster
Callsign	Callsign - unique identification for amateur radio operators (e.g. PE1HVH)
Channel	Shared cryptographic key (PSK) for group communication
Chirp	Frequentiesweep van laag naar hoog (up-chirp) of hoog naar laag (down-chirp)
Companion App	Smartphone application to control the MeshCore node
CR	Coding Rate - foutcorrectieniveau (4/5 tot 4/8), meer = betrouwbaarder
CSS	Chirp Spread Spectrum - the modulation method that LoRa uses
dBm	Decibel-milliwatt - unit for transmit power (14 dBm = 25 mW)
Dechirp	Demodulatie door ontvangen chirp te vermenigvuldigen met lokale down-chirp
DFU	Device Firmware Update - firmware updaten via Bluetooth
DM	Direct Message - private message between two nodes, end-to-end encrypted
Duty Cycle	Percentage zendtijd dat is toegestaan (EU: max 1% op 868 MHz)
E2E	End-to-End encryption - only sender and

	receiver can read messages
EIRP	Effective Isotropic Radiated Power - effectively radiated power incl. antenne
ESP32	Popular microcontroller from Espressif with WiFi and Bluetooth
FFT	Fast Fourier Transform - algoritme voor frequentieanalyse van gedechirpt signaal
Firmware	Software that permanently runs on a node's microcontroller
Flashen	Installing or updating firmware on a device
GPIO	General Purpose Input/Output - connection pins for external devices
GPS/GNSS	Global Navigation Satellite System - satellite navigation for positioning
HAM	Amateur radio modus - licentie vereist, geen encryptie toegestaan
Hop	Eén sprong tussen twee nodes in het mesh-netwerk
I²C	Inter-Integrated Circuit - bus for connecting sensors and displays
IPEX/U.FL	Small click-on antenna connector for internal antennas
ISM-band	Industrial, Scientific, Medical - free frequency band, 868 MHz in Europe
Key Rotation	Periodically replacing cryptographic keys for extra security
Link Budget	Total signal loss that a connection can tolerate and still be decodable
LoRa	Long Range - patented modulation technique for long-range communication
MCU	Microcontroller Unit - the central processor of a node
Mesh	Network where each device can forward messages to others
Meshtastic	Alternatieve LoRa mesh firmware - niet compatibel met MeshCore
MIC	Message Integrity Code - packet authentication (4 bytes)
MIT-licentie	Open-source license for free use, modification and distribution
Node	A device/node in the MeshCore network

Node-ID	Unique identification of a node in the network (4 bytes)
nRF52840	Nordic microcontroller with ultra-low power consumption and Bluetooth
OTA	Over-The-Air - wireless firmware update
PHY	Physical Layer - the physical radio layer that converts bits to radio signals
Preamble	Series of identical chirps at the beginning of each packet for synchronization
Private Key	Private key - kept secret, for decrypting messages
Processing Gain	Signal amplification through spreading over many samples (SF12: ~36 dB)
Public Key	Public key - freely shareable, for encrypting messages
Repeater	Node that forwards messages to extend network range
Room Server	Physical node with BBS function for store-and-forward (up to 32 messages)
Routing	Determining the best route for a message through the network
SF	Spreading Factor - determines range vs speed (SF7-SF12), higher = further
SMA	SubMiniature version A - screw-on antenna connector
SNR	Signal-to-Noise Ratio - ratio between signal and noise in dB
SPI	Serial Peripheral Interface - fast bus for LoRa chip and SD card
Store-and-forward	Store messages until recipient is reachable
SX1262	Semtech LoRa radio chip - newer, more efficient version
SX1276	Semtech LoRa radio chip - older but still used version
Sync Word	Network identifier (2 bytes) to separate different networks
Telemetry	Measurement data from sensors sent through the network
UART	Universal Asynchronous Receiver-Transmitter - serial communication interface
Web Flasher	Browser tool to install firmware without

	special software
Wrap	Frequentie springt terug naar begin van de band (0) na bereiken van maximum

PART D

References and Sources

19. References and Sources

Comprehensive source list organized by category.

19.1 Datasheets

Titel:	SX1276/77/78/79 Datasheet
URL:	https://cdn-shop.adafruit.com/product-files/3179/sx1276_77_78_79.pdf
Bron:	Semtech
Beschrijving:	Official datasheet for legacy SX127x LoRa transceivers with spreading factors and modulation specs

Titel:	SX1262 Productpagina
URL:	https://www.semtech.com/products/wireless-rf/lora-connect/sx1262
Bron:	Semtech
Beschrijving:	Product information for the newer SX1262 LoRa transceiver

19.2 Application Notes

Titel:	AN1200.22 LoRa Modulation Basics
URL:	https://www.frugalprototype.com/wp-content/uploads/2016/08/an1200.22.pdf
Bron:	Semtech
Beschrijving:	Fundamentele application note over Chirp Spread Spectrum modulatie en processing gain

Titel:	AN1200.13 LoRa Modem Designer's Guide
URL:	https://www.openhacks.com/uploadsproductos/loradesignguide_std.pdf
Bron:	Semtech
Beschrijving:	Technical manual on SNR requirements and spread spectrum processing gain calculations

Titel:	RSSI and SNR for LoRa Modulation
URL:	https://www.st.com/resource/en/application_note/an5664-rssi-and-snr-for-lora-modulation-on-stm32wl-series-stmicroelectronics.pdf
Bron:	STMicroelectronics
Beschrijving:	Application note over processing gain berekening en detectie van signalen onder de ruisvloer

19.3 Academic Papers

Titel:	From Demodulation to Decoding: Complete LoRa PHY
URL:	https://dl.acm.org/doi/10.1145/3546869
Bron:	ACM TOSN
Beschrijving:	Diepgaande analyse van LoRa encoding pipeline, dechirping en FFT piekdetectie

Titel:	Complete Reverse Engineering of LoRa PHY
URL:	https://www.epfl.ch/labs/tcl/wp-content/uploads/2020/02/Reverse_Eng_Report.pdf
Bron:	EPFL
Beschrijving:	Gedetailleerd reverse engineering rapport van de LoRa fysieke laag

Titel:	A Tutorial on CSS for LoRaWAN
URL:	https://arxiv.org/abs/2310.10503
Bron:	arXiv
Beschrijving:	Comprehensive academic tutorial on CSS modulation in LoRaWAN systems

Titel:	LoRaWAN Mesh Networks: A Review
URL:	https://pmc.ncbi.nlm.nih.gov/articles/PMC7435450/
Bron:	PMC/MDPI
Beschrijving:	Comprehensive overview of multihop mechanisms and mesh topologies for LoRaWAN

Titel:	NELoRa: Ultra-low SNR LoRa Communication
URL:	https://cse.msu.edu/~caozc/papers/sensys21-li.pdf
Bron:	SenSys'21
Beschrijving:	Research on dechirp signal concentration and SNR threshold analysis

Titel:	FTrack: Parallel Decoding for LoRa
URL:	https://www4.comp.polyu.edu.hk/~csyqzheng/papers/FTrack_Sensys19.pdf
Bron:	SenSys'19
Beschrijving:	Paper on FFT peak detection in dechirped signals

Titel:	Design of a Baseband LoRa Demodulator
URL:	https://ieeexplore.ieee.org/document/8836176/
Bron:	IEEE
Beschrijving:	Paper on digital LoRa demodulator design with de-chirp method and FFT

19.4 Educational Resources

Titel:	LoRa/CSS: Overview and Decoding
URL:	https://gyulab.github.io/lora/
Bron:	Gyujun Jeong
Beschrijving:	Comprehensive tutorial on CSS modulation, dechirping and FFT-based symbol detection

Titel:	Understanding LoRa PHY
URL:	https://wirelesspi.com/understanding-lora-phy-long-range-physical-layer/
Bron:	Wireless Pi
Beschrijving:	Accessible explanation of LoRa physical layer concepts and de-chirping

Titel:	LoRa modem with LimeSDR
URL:	https://myriadrf.org/news/lora-modem-limesdr/
Bron:	MyriadRF
Beschrijving:	Practical guide on dechirping through conjugate chirp multiplication

Titel:	The Hidden Side of LoRa
URL:	https://www.disk91.com/2024/technology/lora/the-hidden-side-of-lora/
Bron:	disk91
Beschrijving:	Explanation of the internal LoRa packet structure

Titel:	LoRa Link Budget Calculations
URL:	https://www.techplayon.com/lora-link-budget-sensitivity-calculations-example-explained/
Bron:	Techplayon
Beschrijving:	Practical explanation of spread spectrum processing gain and SNR requirements

19.5 Standards and Specifications

Titel:	Spreading Factors
URL:	https://www.thethingsnetwork.org/docs/lorawan/spreading-factors/
Bron:	TTN
Beschrijving:	Official documentation on spreading factors and receiver sensitivity

Titel:	ETSI EN 300 220
URL:	https://www.etsi.org/
Bron:	ETSI
Beschrijving:	Europese standaard voor korteafstandsapparatuur in de 868 MHz ISM-band

Titel:	LoRa Alliance Regional Parameters
URL:	https://www.lora-alliance.org/
Bron:	LoRa Alliance
Beschrijving:	Official specifications for regional frequency plans (EU868, US915)

Titel:	LoRa (Wikipedia)
URL:	https://en.wikipedia.org/wiki/LoRa
Bron:	Wikipedia
Beschrijving:	Overview of cyclically shifted chirps, spreading factor selection (SF5-SF12)

19.6 MeshCore Documentation

Titel:	MeshCore Officiële Website
URL:	https://meshcore.co.uk/
Bron:	MeshCore
Beschrijving:	Official website with product information, downloads and documentation

Titel:	MeshCore GitHub Repository
URL:	https://github.com/meshcore-dev/MeshCore
Bron:	GitHub
Beschrijving:	Open-source repository with MeshCore firmware and example code

Titel:	MeshCore FAQ
URL:	https://github.com/meshcore-dev/MeshCore/wiki/FAQ
Bron:	GitHub Wiki
Beschrijving:	Frequently asked questions about advertising, frequencies, firmware and licenses

Titel:	MeshCore Web Flasher
URL:	https://flasher.meshcore.co.uk/
Bron:	MeshCore
Beschrijving:	Web-based firmware flash tool for supported devices

Titel:	MeshCore Companion Apps
URL:	https://meshcore.co.uk/apps.html
Bron:	MeshCore
Beschrijving:	Overview of available client applications for Android, iOS and web

Titel:	MeshCore Discord Server
URL:	https://discord.gg/ZVH2ujy9ex
Bron:	Discord
Beschrijving:	Official community server for support and development discussion

Titel:	MeshCore Kaart
URL:	https://map.meshcore.dev/
Bron:	MeshCore
Beschrijving:	Live kaart met actieve MeshCore nodes, repeaters en room servers

Titel:	MeshCore Nederlandstalig Forum
URL:	https://forum.meshcore-net.nl/
Bron:	Community
Beschrijving:	Dutch-language forum for MeshCore users

19.7 Hardware Documentation

Titel:	Heltec WiFi LoRa 32 V3
URL:	https://heltec.org/project/wifi-lora-32-v3/
Bron:	Heltec
Beschrijving:	Product page for Heltec V3 development board with ESP32-S3 and SX1262

Titel:	Heltec WiFi LoRa 32 V4
URL:	https://heltec.org/project/wifi-lora-32-v4/
Bron:	Heltec
Beschrijving:	Product page for Heltec V4 with 28dBm transmit power

Titel:	**RAKwireless WisBlock**
URL:	https://store.rakwireless.com/
Bron:	RAKwireless
Beschrijving:	Modular IoT hardware system with RAK4631 (nRF52840 + SX1262)

Titel:	**LilyGO T-Deck**
URL:	https://www.lilygo.cc/
Bron:	LilyGO
Beschrijving:	Standalone MeshCore device with screen and keyboard

Titel:	**Seeed Studio T1000-E**
URL:	https://www.seeedstudio.com/
Bron:	Seeed Studio
Beschrijving:	Compact GPS tracker with nRF52840 and LoRa

— End of Document —