

## MeshCom/MeshCom 2.0

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

**Version vom 10. Juni 2022, 13:18 Uhr (Quelle anzeigen)**

[Oe3mzc](#) ([Diskussion](#) | [Beiträge](#))

[K](#) (→[MeshCom 2.0](#))

Markierung: **Visuelle Bearbeitung**

[← Zum vorherigen Versionsunterschied](#)

**Version vom 10. Juni 2022, 13:20 Uhr (Quelle anzeigen)**

[Oe3mzc](#) ([Diskussion](#) | [Beiträge](#))

(→[Grundlegende Spezifikationen](#))

Markierung: **Visuelle Bearbeitung**

[Zum nächsten Versionsunterschied →](#)

Zeile 4:

\* '''Luftschnittstelle'''

\*\* AFU kompatibel der Source, Node, Gateway, Destination Kennung als Rufzeichen

\*\* Path-Kontrollstruktur (nur für Testzwecke)

Zeile 4:

\* '''Luftschnittstelle'''

+

**\*\* Mesh Netzwerk - selbst bildend und selbstheilend**

\*\* AFU kompatibel der Source, Node, Gateway, Destination Kennung als Rufzeichen

\*\* Path-Kontrollstruktur (nur für Testzwecke)

### Version vom 10. Juni 2022, 13:20 Uhr

## MeshCom 2\0

### Grundlegende Spezifikationen

- **Luftschnittstelle**
  - Mesh Netzwerk - selbst bildend und selbstheilend
  - AFU kompatibel der Source, Node, Gateway, Destination Kennung als Rufzeichen
  - Path-Kontrollstruktur (nur für Testzwecke)
  - Struktur der Payload in die Struktur der Meldung eingebettet
  - Zusätzlich zur Übertragungs-Sicherung durch die Hardware sind CRC und FEC in der Struktur der Meldung einzuplanen
  - Meldung und Payload komprimiert übertragen
  - Node, Digipeater-only, Gateway-only, Point-to-Point (Netzerweiterungen)
  - unverschlüsselt
  - Adress-Header (FromCALL, ToCALL, VIA) komprimiert und mit CRC (kompatibel zu AX25v2)
- **Gateway-Schnittstelle**
  - MQTT-Protokoll mit üblicher Feldstruktur aufbauen
  - UDP-Übertragung

- 
- Hardbeat zur Partner-ONLINE Erkennung
  - Tiefe der Meldung vom und zum Gateway einstellbar (Test- und Entwicklungs-Erleichterung)
  - Nach neustart eines Gateways automatischer Übertragung von Grunddaten wie aktive NODES, Letzter Meldungs-ID Stack, ...
  - **Modul-Schnittstellen**
    - Serial via USB
    - GPIO für externe Hardware und Steuerungen
    - GPS intern, extern, fix
    - WiFi
      - Userschnittstelle
      - Gateway-Schnittstelle
    - Bluetooth
      - APP-Schnittstelle
    - ETH-Schnittstelle optional
  - **Meldungs-Grundtypen**
    - Broadcast
    - Group Call
    - Private Call
    - Store & Forward
    - Entwicklungs- und Debug-Meldungen
  - **Offene Hardware**
    - Die Verwendung der kompatibler MCU sollte eingehalten werden
    - ESP32
    - Fertigmodule MCU, HF, GPS gemeinsam
    - wie TTGO, TLORA, HELTEC, ...
    - Bevorzugterweise Aufbau Basisplatine, Steckmodule
    - wie RAK WisBlock
    - Vorhandene Hardware aus dem LoRa-APRS Projekt
    - Semtech SX1262 LoRa-Transceiver oder kompatibel
    - ETH-Modulblock mit IP-Stack für Gateways
  - **Firmware**
    - Grundstruktur für Entwicklung in der Gruppe vorbereitet
    - Leicht zu erweitern, pflegen
    - Klare Funktionsgliederung
    - Keine direkte Hardware-Bezogenheit in der Logik-Struktur
    - Logik-Struktur mit klaren Schnittstellen aufgebaut um funktionelle Erweiterungen jederzeit einzubauen ohne die getestete Basisfunktionalität zu beeinflussen
  - **Welche Service bietet MeshCom 2.0 an?**
    - Textübertragung
    - Positionsübertragung (Smart Beaconsing)
    - Frei definierbare Payload
  - **Feature-List**
    - Konfiguration über USB-Serial-Schnittstelle
    - Rufzeichen mit APRS-konformen SSID
    - Frequenzeinstellung und Anzeige

- 
- Feldstärkeanzeige (S-Meter, RSSI, MER)
  - LoRa-Modulationsparameter auch detailliert
  - Fix-Position
  - Batterie-Management Stufen
  - Scannen nach verfügbarem MeshCom-Channel
  - **Use Cases**
    - allg. Amateurfunknachrichtendienst
    - Not-Katfunk
    - Infodienste z.B. mit Wetterbericht, SolarFlux, Radioaktivität, Blitzortung, DXCluster, Skeds, SOTA,...

Entwurf: Kurt OE1KBC

Diskussion: TELEGRAM Gruppe MeshCom