

MeshCom/MeshCom Anwendungen

Versionsgeschichte interaktiv durchsuchen

VisuellWikitext

Version vom 2. Februar 2022, 13:25 Uhr (

Quelltext anzeigen)

Oe6rke ([Diskussion](#) | [Beiträge](#))

(→Sammlung diverser Anwendungsideen)

Markierung: Visuelle Bearbeitung

← Zum vorherigen Versionsunterschied

Version vom 2. Februar 2022, 17:00 Uhr (

Quelltext anzeigen)

Oe1kbc ([Diskussion](#) | [Beiträge](#))

K

Markierung: Visuelle Bearbeitung

Zum nächsten Versionsunterschied →

Zeile 29:

*Aussenden von Daten aus Citizen Science Projekten (Radioaktivität, <https://safecast.org/devices/>, <https://www.gmcmap.com/index.asp>)

*Verbindung mit SMS-Funktion in DMR-Netzen (IPSC2)

– ***meshCom** als **afu** relevanten Newsticker verwenden (**zbsp** Aussenden von Eventankündigung) analog **eine Afu pagers**

– *Fernwirken via **meshCom** (Ein /Ausschalten von exponierten Relais)

– *HAMNET **last mile** Ersatz (langsame Datenspeed) als Store & Forward Routing Device

– *Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne **mqtt**) aka Cellbroadcast

– *Failover **mqtt** Server mit **lokal** (Bundesland) **Erst-Mqtt** zur Vermittlung trotz ausgefallenen **Haupt Mqtt**

*Merkmal der klaren Identifikation des OM (cert, echolink login) um Identität feststellen zu können

*REST API App lokale zum Device für eigene Erweiterungen via PortIO, Arduino oder .Net Dev Umgebung

Zeile 29:

*Aussenden von Daten aus Citizen Science Projekten (Radioaktivität, <https://safecast.org/devices/>, <https://www.gmcmap.com/index.asp>)

*Verbindung mit SMS-Funktion in DMR-Netzen (IPSC2)

+ ***MeshCom** als **AFU** relevanten Newsticker verwenden (**zb.** Aussenden von Eventankündigung) analog **eines AFU-Pagers**

+ *Fernwirken via **MeshCom** (Ein /Ausschalten von exponierten Relais)

+ *HAMNET **lastmile** Ersatz (langsame Datenspeed) als Store & Forward Routing Device

+ *Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne **MQTT**) aka Cellbroadcast

+ *Failover-**MQTT**-Server mit **lokaler** Bundesland-**Master-MQTT** zur Vermittlung trotz ausgefallenen **Master-MQTT**

*Merkmal der klaren Identifikation des OM (cert, echolink login) um Identität feststellen zu können

*REST API App lokale zum Device für eigene Erweiterungen via PortIO, Arduino oder .Net Dev Umgebung

		*Kopplung des zentralen MQTT mit Azure PowerApps und AWS Cloud zur Abbildung eigener Apps		*Kopplung des zentralen MQTT mit Azure PowerApps und AWS Cloud zur Abbildung eigener Apps
–		*Asynchrones Datenmodem mit meshCom (RX zum User via ATV/DATV/Baken Subcarrier, TX request via MeshCom), ähnlich den ersten SAT via Internet Anwendungen	+	*Asynchrones Datenmodem mit MeshCom (RX zum User via ATV/DATV/Baken Subcarrier, TX request via MeshCom), ähnlich den ersten SAT via Internet Anwendungen
–		*einfacher Entfernungspeiler (je weniger Hops umso näher drann bei ausgeschalt enen GPS)	+	*einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschalte nem GPS)
		*Multi MeshCom Channel Bounding für größere Geschwindigkeiten		*Multi MeshCom Channel Bounding für größere Geschwindigkeiten
–		*Multifunktions Device (via Button Sequenz von MeshCom zu APRS Tracker zu Radiosonden Module zu SSTV Device umschalten), Apps dabei von SD Karte ziehen	+	*Multifunktions-Device (via Button Sequenz von MeshCom zu APRS Tracker zu Radiosonden Module zu SSTV Device umschalten), Apps dabei von SD Karte ziehen
–		* Konfigurationsrepository online oder l ocal auf SD Karte legen für schnelles Cloning oder Wiederherstellen von Hardware	+	* Konfigurations-Repository online oder lokal auf SD Karte legen für schnelles Cloning oder Wiederherstellen von Hardware
		*Arduino Library und Sketch vorsehen, damit MeshCom in eigene Schaltungen leicht integriert werden kann (wie universeller Steckbaustein) für OMs die dort programmieren		*Arduino Library und Sketch vorsehen, damit MeshCom in eigene Schaltungen leicht integriert werden kann (wie universeller Steckbaustein) für OMs die dort programmieren
		*wer hat weitere Ideen?		*wer hat weitere Ideen?
		__HIDETITLE__		__HIDETITLE__
		__KEIN_INHALTSVERZEICHNIS__		__KEIN_INHALTSVERZEICHNIS__

Version vom 2. Februar 2022, 17:00 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXTINFO)**
 - Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POSINFO)**
 - Positionen welche mittels eingebautem GPS-Empfänger empfangen wurden können regelmäßig via MeshCom gesendet werden. Die Übertragung zu aprs.fi Cloud ist eingebaut.
 - Positionen können auch fix gespeichert werden und werden ebenfalls via MeshCom zu aprs.fi übertragen.
- **Knotenmeldungen (NODEINFO)**
 - Es werden diverse, im Knoten durch Konfiguration oder Betrieb festgelegte Parameter, übertragen:
 - Rufzeichen, auch mit 1-2 stelliger SSID
 - Kurzrufzeichen (AKA) verwendete Hardware
 - Knoten HEX-ID
 - Empfangene Signalstärke

Sammlung diverser Anwendungsideen

- Verknüpfung mit [MORSERINO](#)
- MeshCom via QO-100 WB-Transponder, einen der obersten Kanäle (uplink 2409,25 MHz)
- Notruf-Taster in Gemeinden (3 Tasten für Polizei, Rettung, Feuerwehr)
- Verbindung mit Chat-Betriebsart auf Kurzwelle (JS8Call, RPR-BPQ-Chat, usw.)
- Verbindung mit CONVERS-Chat Kanal auf Packet-Radio
- Aussenden von Daten aus Citizen Science Projekten (Radioaktivität, <https://safecast.org/devices/>, <https://www.gmcmap.com/index.asp>)
- Verbindung mit SMS-Funktion in DMR-Netzen (IPSC2)
- MeshCom als AFU relevanten Newsticker verwenden (zb. Aussenden von Eventankündigung) analog eines AFU-Pagers
- Fernwirken via MeshCom (Ein/Ausschalten von exponierten Relais)
- HAMNET lastmile Ersatz (langsame Datenspeed) als Store & Forward Routing Device

-
- Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
 - Failover-MQTT-Server mit lokaler Bundesland-Master-MQTT zur Vermittlung trotz ausgefallenen Master-MQTT
 - Merkmal der klaren Identifikation des OM (cert, echolink login) um Identität feststellen zu können
 - REST API App lokale zum Device für eigene Erweiterungen via PortIO, Arduino oder .Net Dev Umgebung
 - Kopplung des zentralen MQTT mit Azure PowerApps und AWS Cloud zur Abbildung eigener Apps
 - Asynchrones Datenmodem mit MeshCom (RX zum User via ATV/DATV/Baken Subcarrier, TX request via MeshCom), ähnlich den ersten SAT via Internet Anwendungen
 - einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem GPS)
 - Multi MeshCom Channel Bounding für größere Geschwindigkeiten
 - Multifunktions-Device (via Button Sequenz von MeshCom zu APRS Tracker zu Radiosonden Module zu SSTV Device umschalten), Apps dabei von SD Karte ziehen
 - Konfigurations-Repository online oder lokal auf SD Karte legen für schnelles Cloning oder Wiederherstellen von Hardware
 - Arduino Library und Sketch vorsehen, damit MeshCom in eigene Schaltungen leicht integriert werden kann (wie universeller Steckbaustein) für OMs die dort programmieren
 - wer hat weitere Ideen?