

Inhaltsverzeichnis

1. MeshCom/MeshCom Anwendungen	22
2. Benutzer:Oe1kbc	7
3. Benutzer:Oe6rke	12
4. MeshCom	17

MeshCom/MeshCom Anwendungen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 2. Februar 2022, 13:17 Uhr (

Quelltext anzeigen)

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

(→[Sammlung diverser Anwendungsideen](#))

Markierung: [Visuelle Bearbeitung](#)

← [Zum vorherigen Versionsunterschied](#)

Aktuelle Version vom 18. März 2024, 15:

59 Uhr (Quelltext anzeigen)

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

Markierung: [Visuelle Bearbeitung](#)

(8 dazwischenliegende Versionen von 4 Benutzern werden nicht angezeigt)

Zeile 2:	Zeile 2:
<div></div>	<div></div>
<div>==Anwendungen==</div>	<div>==Anwendungen==</div>
<div>– <div>[[Datei:MeshCom.jpg 200x200px]]</div></div>	<div>+ <div>[[Datei:MESHCOM 40 LOGO SCHMAL.png alternativtext= 400x400px]]</div></div>
<div></div>	<div></div>
<div>====Bereits im MeshCom Grundmodul vorhanden:====</div>	<div>====Bereits im MeshCom Grundmodul vorhanden:====</div>
<div></div>	<div></div>
<div>– <div>'''Meldungstexte (TEXTINFO)'''</div></div>	<div>+ <div>'''Meldungstexte (TEXT)'''</div></div>
<div>– <div>**Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden</div></div>	<div>+ <div>**Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden</div></div>
<div></div>	<div></div>
<div>– <div>'''Positionsmeldungen (POSINFO)'''</div></div>	<div>+ <div>'''Positionsmeldungen (POS)'''</div></div>
<div><div>**Positionen welche mittels eingebautem GPS-Empfänger empfangen wurden können regelmäßig via MeshCom gesendet werden. Die Übertragung zu aprs.fi Cloud ist eingebaut.</div></div>	<div><div>**Positionen welche mittels eingebautem GPS-Empfänger empfangen wurden können regelmäßig via MeshCom gesendet werden. Die Übertragung zu aprs.fi Cloud ist eingebaut.</div></div>
<div>– <div>**Positionen können auch fix gespeichert werden und werden ebenfalls via MeshCom zu aprs.fi übertragen.

</div></div>	<div>+ <div>**Positionen können auch fix gespeichert werden und werden ebenfalls via MeshCom zu aprs.fi übertragen.</div></div>
<div></div>	
<div>– <div></div></div>	
<div>– <div>'''Knotenmeldungen (NODEINFO)'''</div></div>	

– ****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=====

=====Sammlung diverser
Anwendungsideen=====

Zeile 27:

*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 (**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

Zeile 20:

*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>, **Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)**

*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	+	*Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
-	*Failover mqtt Server mit localer (Bundesland) Erst-Mqtt zur Vermittlung trotz ausgefallenen Haupt Mqtt	+	*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		*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		*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 ausgeschalteten GPS)	+	*einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem 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 lokal 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

				*Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
				*SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
	*wer hat weitere Ideen?			*wer hat weitere Ideen?
-	__HIDETITLE__	+		
	__KEIN_INHALTSVERZEICHNIS__			__KEIN_INHALTSVERZEICHNIS__

Aktuelle Version vom 18. März 2024, 15:59 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXT)**
 - Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POS)**
 - 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.

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>, Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)
 - 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
 - Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
 - SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
 - wer hat weitere Ideen?

MeshCom/MeshCom Anwendungen: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 2. Februar 2022, 13:17 Uhr (
Quelltext anzeigen)

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

(→[Sammlung diverser Anwendungsideen](#))

Markierung: **Visuelle Bearbeitung**

← [Zum vorherigen Versionsunterschied](#)

Aktuelle Version vom 18. März 2024, 15:
59 Uhr (Quelltext anzeigen)

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

Markierung: **Visuelle Bearbeitung**

(8 dazwischenliegende Versionen von 4 Benutzern werden nicht angezeigt)

Zeile 2:	Zeile 2:
<div></div>	<div></div>
<div>==Anwendungen==</div>	<div>==Anwendungen==</div>
<div>- [[Datei:MeshCom.jpg 200x200px]]</div>	<div>+ [[Datei:MESHCOM 40 LOGO SCHMAL.png alternativtext= 400x400px]]</div>
<div></div>	<div></div>
<div>====Bereits im MeshCom Grundmodul vorhanden:====</div>	<div>====Bereits im MeshCom Grundmodul vorhanden:====</div>
<div></div>	<div></div>
<div>- *'''Meldungstexte (TEXTINFO)'''</div>	<div>+ *'''Meldungstexte (TEXT)'''</div>
<div>- **Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden</div>	<div>+ **Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden</div>
<div></div>	<div></div>
<div>- *'''Positionsmeldungen (POSINFO)'''</div>	<div>+ *'''Positionsmeldungen (POS)'''</div>
<div>**Positionen welche mittels eigebautem GPS-Empfänger empfangen wurden können regelmäßig via MeshCom gesendet werden. Die Übertragung zu aprs.fi Cloud ist eingebaut.</div>	<div>**Positionen welche mittels eigebautem GPS-Empfänger empfangen wurden können regelmäßig via MeshCom gesendet werden. Die Übertragung zu aprs.fi Cloud ist eingebaut.</div>
<div>- **Positionen können auch fix gespeichert werden und werden ebenfalls via MeshCom zu aprs.fi übertragen.

</div>	<div>+ **Positionen können auch fix gespeichert werden und werden ebenfalls via MeshCom zu aprs.fi übertragen.</div>
<div>- </div>	

– *****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=====

=====Sammlung diverser
Anwendungsideen=====

Zeile 27:

*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 (**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

Zeile 20:

*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>, **Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)**

*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	+	*Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
-	*Failover mqtt Server mit localer (Bundesland) Erst-Mqtt zur Vermittlung trotz ausgefallenen Haupt Mqtt	+	*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		*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		*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 ausgeschalteten GPS)	+	*einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem 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 lokal 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

				*Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
				*SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
	*wer hat weitere Ideen?			*wer hat weitere Ideen?
-	__HIDETITLE__	+		
	__KEIN_INHALTSVERZEICHNIS__			__KEIN_INHALTSVERZEICHNIS__

Aktuelle Version vom 18. März 2024, 15:59 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXT)**
 - Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POS)**
 - 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.

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>, Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)
- 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
- Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
- SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
- wer hat weitere Ideen?

MeshCom/MeshCom Anwendungen: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 2. Februar 2022, 13:17 Uhr (
Quelltext anzeigen)

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

(→[Sammlung diverser Anwendungsideen](#))

Markierung: **Visuelle Bearbeitung**

← [Zum vorherigen Versionsunterschied](#)

Aktuelle Version vom 18. März 2024, 15:
59 Uhr (Quelltext anzeigen)

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

Markierung: **Visuelle Bearbeitung**

(8 dazwischenliegende Versionen von 4 Benutzern werden nicht angezeigt)

Zeile 2:

-	[[Datei: MeshCom.jpg 200x200px]]
	====Bereits im MeshCom Grundmodul vorhanden:====
-	***Meldungstexte (TEXTINFO)***
-	**Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden
-	***Positionsmeldungen (POSINFO)***
	**Positionen welche mittels eigebautem 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. <br />

-	

Zeile 2:

+	[[Datei: MESHCOM 40 LOGO SCHMAL.png alternativtext= 400x400px]]
	====Bereits im MeshCom Grundmodul vorhanden:====
+	***Meldungstexte (TEXT)***
+	**Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
+	***Positionsmeldungen (POS)***
	**Positionen welche mittels eigebautem 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**
- *****Kurzurufzeichen (AKA) verwendete Hardware**
- *****Knoten HEX-ID**
- *****Empfangene Signalstärke**

=====Sammlung diverser
Anwendungsideen=====

Zeile 27:

*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 (**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

Zeile 20:

*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>, **Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)**

*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

<ul style="list-style-type: none"> - *Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne mqtt) aka Cellbroadcast 	<ul style="list-style-type: none"> + *Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
<ul style="list-style-type: none"> - *Failover mqtt Server mit localer (Bundesland) Erst-Mqtt zur Vermittlung trotz ausgefallenen Haupt Mqtt 	<ul style="list-style-type: none"> + *Failover-MQTT-Server mit lokaler Bundesland-Master-MQTT zur Vermittlung trotz ausgefallenen Master-MQTT
<ul style="list-style-type: none"> - *Merkmal der klaren Identifikation des OM (cert, echolink login) um Identität feststellen zu können 	<ul style="list-style-type: none"> + *Merkmal der klaren Identifikation des OM (cert, echolink login) um Identität feststellen zu können
<ul style="list-style-type: none"> - *REST API App lokale zum Device für eigene Erweiterungen via PortIO, Arduino oder .Net Dev Umgebung 	<ul style="list-style-type: none"> + *REST API App lokale zum Device für eigene Erweiterungen via PortIO, Arduino oder .Net Dev Umgebung
<ul style="list-style-type: none"> - *Kopplung des zentralen MQTT mit Azure PowerApps und AWS Cloud zur Abbildung eigener Apps 	<ul style="list-style-type: none"> + *Kopplung des zentralen MQTT mit Azure PowerApps und AWS Cloud zur Abbildung eigener Apps
<ul style="list-style-type: none"> - *Asynchrones Datenmodem mit meshCom (RX zum User via ATV/DATV/Baken Subcarrier, TX request via MeshCom), ähnlich den ersten SAT via Internet Anwendungen 	<ul style="list-style-type: none"> + *Asynchrones Datenmodem mit MeshCom (RX zum User via ATV/DATV/Baken Subcarrier, TX request via MeshCom), ähnlich den ersten SAT via Internet Anwendungen
<ul style="list-style-type: none"> - *einfacher Entfernungspeiler (je weniger Hops umso näher drann bei ausgeschalteten GPS) 	<ul style="list-style-type: none"> + *einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem GPS)
<ul style="list-style-type: none"> - *Multi MeshCom Channel Bounding für größere Geschwindigkeiten 	<ul style="list-style-type: none"> + *Multi MeshCom Channel Bounding für größere Geschwindigkeiten
<ul style="list-style-type: none"> - *Multifunktions Device (via Button Sequenz von MeshCom zu APRS Tracker zu Radiosonden Module zu SSTV Device umschalten), Apps dabei von SD Karte ziehen 	<ul style="list-style-type: none"> + *Multifunktions-Device (via Button Sequenz von MeshCom zu APRS Tracker zu Radiosonden Module zu SSTV Device umschalten), Apps dabei von SD Karte ziehen
<ul style="list-style-type: none"> - *Konfigurationsrepository online oder lokal auf SD Karte legen für schnelles Cloning oder Wiederherstellen von Hardware 	<ul style="list-style-type: none"> + *Konfigurations-Repository online oder lokal auf SD Karte legen für schnelles Cloning oder Wiederherstellen von Hardware
	<ul style="list-style-type: none"> + *Arduino Library und Sketch vorsehen, damit MeshCom in eigene Schaltungen leicht integriert werden kann (wie universeller Steckbaustein) für OMs die dort programmieren

				*Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
				*SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
	*wer hat weitere Ideen?			*wer hat weitere Ideen?
-	__HIDETITLE__	+		
	__KEIN_INHALTSVERZEICHNIS__			__KEIN_INHALTSVERZEICHNIS__

Aktuelle Version vom 18. März 2024, 15:59 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXT)**
 - Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POS)**
 - 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.

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>, Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)
- 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
- Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
- SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
- wer hat weitere Ideen?

MeshCom/MeshCom Anwendungen: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 2. Februar 2022, 13:17 Uhr (
Quelltext anzeigen)

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

(→[Sammlung diverser Anwendungsideen](#))

Markierung: **Visuelle Bearbeitung**

← [Zum vorherigen Versionsunterschied](#)

Aktuelle Version vom 18. März 2024, 15:
59 Uhr (Quelltext anzeigen)

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

Markierung: **Visuelle Bearbeitung**

(8 dazwischenliegende Versionen von 4 Benutzern werden nicht angezeigt)

Zeile 2:

-	[[Datei: MeshCom.jpg 200x200px]]
	====Bereits im MeshCom Grundmodul vorhanden:====
-	*'''Meldungstexte (TEXTINFO)'''
-	**Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden
-	*'''Positionsmeldungen (POSINFO)'''
	**Positionen welche mittels eigebautem 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. <br />

-	

Zeile 2:

+	[[Datei: MESHCOM 40 LOGO SCHMAL.png alternativtext= 400x400px]]
	====Bereits im MeshCom Grundmodul vorhanden:====
+	*'''Meldungstexte (TEXT)'''
+	**Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
+	*'''Positionsmeldungen (POS)'''
	**Positionen welche mittels eigebautem 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**
- *****Kurzurufzeichen (AKA) verwendete Hardware**
- *****Knoten HEX-ID**
- *****Empfangene Signalstärke**

=====Sammlung diverser
Anwendungsideen=====

Zeile 27:

*Verbindung mit Chat-Betriebsart auf
Kurzwele (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 (**zbsp** Aussenden von
Eventankündigung) analog **eine Afu
pagars**

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

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

Zeile 20:

*Verbindung mit Chat-Betriebsart auf
Kurzwele (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>, **Infraschall
(=Erdbebenerkennung,
Feinstaubsensor, statische
Luftspannung via Feldmühle,
Bakenerkennung Mikrowelle zum
Erkennen regionaler Tropo
Situationen)**

*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	+	*Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
–	*Failover mqtt Server mit localer (Bundesland) Erst-Mqtt zur Vermittlung trotz ausgefallenen Haupt Mqtt	+	*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		*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		*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 ausgeschalteten GPS)	+	*einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem 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 lokal 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

				*Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
				*SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
	*wer hat weitere Ideen?			*wer hat weitere Ideen?
-	__HIDETITLE__	+		
	__KEIN_INHALTSVERZEICHNIS__			__KEIN_INHALTSVERZEICHNIS__

Aktuelle Version vom 18. März 2024, 15:59 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXT)**
 - Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POS)**
 - 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.

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>, Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)
- 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
- Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
- SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
- wer hat weitere Ideen?

MeshCom/MeshCom Anwendungen: Unterschied zwischen den Versionen

[Versionsgeschichte interaktiv durchsuchen](#)

[Visuell Wikitext](#)

Version vom 2. Februar 2022, 13:17 Uhr (
Quelltext anzeigen)

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

(→[Sammlung diverser Anwendungsideen](#))

Markierung: **Visuelle Bearbeitung**

← [Zum vorherigen Versionsunterschied](#)

Aktuelle Version vom 18. März 2024, 15:
59 Uhr (Quelltext anzeigen)

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

Markierung: **Visuelle Bearbeitung**

(8 dazwischenliegende Versionen von 4 Benutzern werden nicht angezeigt)

Zeile 2:

==Anwendungen==
- [[Datei: MeshCom.jpg 200x200px]]
====Bereits im MeshCom Grundmodul vorhanden:====
- *"'Meldungstexte (TEXTINFO)"'
- **Texte bis zu 228 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- *"'Positionsmeldungen (POSINFO)"'
**Positionen welche mittels eigebautem 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.

-

Zeile 2:

==Anwendungen==
+ [[Datei: MESHCOM 40 LOGO SCHMAL.png alternativtext= 400x400px]]
====Bereits im MeshCom Grundmodul vorhanden:====
+ *"'Meldungstexte (TEXT)"'
+ **Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
+ *"'Positionsmeldungen (POS)"'
**Positionen welche mittels eigebautem 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=====

=====Sammlung diverser Anwendungsideen=====

Zeile 27:

*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 (**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

Zeile 20:

*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>, **Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)**

*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	+	*Sferics local Detektor zum Melden von Gewitter via dezentraler Intrastruktur (ohne MQTT) aka Cellbroadcast
-	*Failover mqtt Server mit localer (Bundesland) Erst-Mqtt zur Vermittlung trotz ausgefallenen Haupt Mqtt	+	*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		*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		*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 ausgeschalteten GPS)	+	*einfacher Entfernungspeiler (je weniger Hops umso näher dran bei ausgeschaltetem 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 lokal 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

				*Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
				*SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
	*wer hat weitere Ideen?			*wer hat weitere Ideen?
-	__HIDETITLE__	+		
	__KEIN_INHALTSVERZEICHNIS__			__KEIN_INHALTSVERZEICHNIS__

Aktuelle Version vom 18. März 2024, 15:59 Uhr

[zurück zu Kategorie:MeshCom](#)

Anwendungen



Bereits im MeshCom Grundmodul vorhanden\:

- **Meldungstexte (TEXT)**
 - Texte bis zu 160 Zeichen können vom Smartphone übertragen bzw- empfangen werden
- **Positionsmeldungen (POS)**
 - 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.

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>, Infraschall (=Erdbebenerkennung, Feinstaubsensor, statische Luftspannung via Feldmühle, Bakenerkennung Mikrowelle zum Erkennen regionaler Tropo Situationen)
- 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
- Rückmeldekanal Device um bei ARDF Veranstaltungen das "Abstempeln" eines OM rückzumelden - Damit kann Klassierung in Echtzeit wie bei F1 Rennen erfolgen bei passender Anwendung
- SOTA-CHANNEL um SOTA Aktivierungen zu melden und anzukündigen
- wer hat weitere Ideen?