

# Seminararbeit: Lorawan

Tobias Sigmann

23. Mai 2019

# Inhaltsverzeichnis

<b>1</b>	<b>Einführung in Lora</b>	<b>3</b>
<b>2</b>	<b>Aufbau eines Lora-Netzwerk</b>	<b>4</b>
2.1	Gateway . . . . .	4
2.2	Netzwerkserver . . . . .	5
2.3	Join-Server . . . . .	5
2.4	End-Gerät . . . . .	6
<b>3</b>	<b>LoraWan Funktionsweise</b>	<b>6</b>
3.1	Schichtenmodell . . . . .	7
3.2	Netzwerkbeitritt . . . . .	8
3.2.1	OTAA . . . . .	8
3.2.2	ABP . . . . .	9
3.3	Protokoll . . . . .	10
3.4	Übertragungsart . . . . .	13
3.4.1	Adaptive Data Rate . . . . .	15
<b>4</b>	<b>Lora Geräte Klassen</b>	<b>16</b>
4.1	Klasse A . . . . .	17
4.2	Klasse B . . . . .	19
4.2.1	Klassenwechsel A nach B . . . . .	20
4.2.2	Betrieb . . . . .	21
4.2.3	Singel / Multicast . . . . .	22
4.2.4	Beacon . . . . .	23
4.3	Klasse C . . . . .	24
4.3.1	Wechsel von A nach C . . . . .	25
<b>5</b>	<b>Sicherheit</b>	<b>25</b>
<b>6</b>	<b>Live-Beispiel</b>	<b>27</b>
<b>7</b>	<b>Ausblick</b>	<b>27</b>

# 1 Einführung in Lora

Lora ist ein Low Power, Wide Area (LPWA) Netzwerkprotokoll und somit sehr gut für batteriebetriebene kabellose Geräte geeignet. Deswegen wird Lora auch oft im Internet of Things (IoT) Bereich verwendet. Mittels der bidirektionalen Kommunikation ist es möglich Daten und Befehle über weite Strecken zu übertragen. Leider leidet darunter die Geschwindigkeit, sodass sich Lora nicht als WLAN Ersatz eignet. Trotzdem können zwischen 0.3 und 50 kbps erreicht werden. In Europa werden 863 MHz bis 870 MHz verwendet. Allerdings variiert der Frequenzbereich für andere Kontinente. Je nach Bedingungen können so bis zu 20km entfernte Endgeräte erkannt und mit diesen kommuniziert werden. Es ist sogar möglich den Standort des Gerätes zu bestimmen.

Eine Alternative zu Lora ist Sigfox, hierauf werde ich nicht weiter eingehen. LoRaWAN 1.1

[Tec15](Optimiert für Batterie Kapazität(Teilnehmer) Reichweite, Kosten mehrjährige Batterielaufzeit, kleine Datenmengen, große Reichweite, LPWAN (Low Power WAN)

Kriterien für Lora: Netzwerk Architektur, Reichweite, Batterielaufzeit, Interferenzrobustheit, Anzahl Knoten, Sicherheit, bidirektionale Kommunikation, verschiedene Anwendungsunterstützung

Orientiert für Mobile Adressierbare Endgeräte)

[AVTP<sup>+</sup>17](alternativen: Sigfox, Ingenu, Dash7

Klassen Kompromiss zwischen Reichweite, Performance(Latenz/ Durchsatz) und Energiebedarf

Energiesparend durch ADR (Adaptive Daten Rate) )

Es wird folgen: Was ist lora, wo und wofür wird es benutzt, wie weit kann man senden und wie schnell...

## 2 Aufbau eines Lora-Netzwerk

Lora wird auch Deswegen gerne für IoT-Geräte verwendet, weil der Netzwerkaufbau ermöglicht die über Lora verwendeten Daten im Internet abzurufen und so ohne weiteres das Gerät mit dem Internet zu verbinden. Um die von den End-Geräten gesendeten LoRa-Pakete auf IP/TCP Pakete umzusetzen wird ein Gateway benötigt, das auf der einen Seite LoRa-Pakete empfängt/sendet und auf der anderen Seite TCP/IP Pakete verwendet. Das Gateway implementiert aber keinerlei Logik. Hierzu ist ein Netzwerkservers zuständig der durch die Gateways das Netzwerk kontrolliert und steuert. Gleichzeitig stellt er die Verbindung zu einem Applikationsserver her, in dem er die vom Gateway empfangenen Daten Weiterleitet.

Der Applikationsserver ist zuständig den die gesendete Nachrichten zu verarbeiten und gegebenenfalls selbst welche an die Endgeräte zu senden.

Diese Architektur wurde gewählt um die Laufzeit der Akku betriebenen Endgeräte, Anzahl der Endgeräte, Qualität Signals und Sicherheit des Netzwerkes möglichst hoch zu halten. [Tec15, S. 8 ff.]

### 2.1 Gateway

Das Teilnetz das aus dem Gateway und mehreren LoRa-Endgeräten besteht ist Sternförmig aufbau. Jedes Endgeräten kommuniziert direkt mit dem Gateway. Diese Art der Kommunikation wird auch “Single-Hop-Connection” zu Deutsch (Einfacher-Sprung-Verbindung) genannt, da die gesendeten Daten ohne Umwege an das Gateway gesendet werden. Jedes Gateway ist mit mindestens einem Netzwerkservers verbunden.

Ein Endgerät kann gleichzeitig an mehreren Gateways senden. Der Netzwerkservers ist zuständig die Pakete auf Duplikate zu überprüfen und nur einmalig an die Applikationsservers zu senden. Ein weiterer Vorteil ist das kein Übergabe der Endgeräte bei Standortwechsel zu andern Gateways nötig ist. Dadurch müssen die Gateways mit vielen Endgeräten kommuniziert. Um diese hohe Endgeräteanzahl zu ermöglichen wurde darauf verzichtet mit jedem Endgerät einzelne zu kommunizieren und stattdessen auf eine Parallele Kom-

munikation gesetzt. Hierzu werden adaptive Datenraten und Mehrkanal-Multi-Modem-Transceiver verwendet.

Durch die genannten Eigenschaften der Gateways wird eine gute Skalierbarkeit erzielt. Dadurch können neue Gateways die Anzahl der Endgeräte um das 6 bis 8-fach erhöhen.vgl. [Tec15, S.10]

## 2.2 Netzwerkserver

Der NetzwerkServer ist das “Herzstück“ eines jeden Lora-Netzwerkes. Er kann mit mehreren Gateways und mehreren Applikationsserver verbunden sein.

Die wichtigste Aufgabe des Netzwerksserver ist das Steuern des LoRa-Teils des Netzwerkes. Der Server verwaltet jedes Endgerät separat indem es mit ihm den zu verwendenden Funkkanal Aushandelt und die Datenrate kontrolliert wenn ADR(Adaptiv Data Rate) verwendet wird. Außerdem ist er bei dem Netzwerkbeitritt eines Endgerätes .beteiligt.

Weiterhin überprüft er die empfangen Pakete auf ihre Korrektheit, Integrität und filtert Duplikate, die durch das Empfangen der gleichen Übertragung von einem Endgerät an verschiedenen Gateways, verursacht wurden. Dabei ermittelt er auch die Gateways, die den besten empfang zu den jeweiligen Endgeräten hat und nutzt dieses um Daten an die Endgeräte zu senden.

Es ist nicht immer möglich Daten direkt zu senden, da die Endgeräte nur manchmal empfangsbereit sind. Um die Applikationsserver zu entlasten, puffert der Netzwerkserver die Daten und sendet diese zum nächst möglichem Zeitpunkt.

Eine weitere sehr Wichtige Ausgabe ist es eine API für den Applikationsserver bereitzustellen um eine einfache und schnelle Kommunikation zu ermöglichen. [SOR17b]

## 2.3 Join-Server

Der Server kann mit mehreren Netzwerkservern verbunden werden und jeder Netzwerkserver kann mehrere Join-Server haben.

Ein Join-Server wird benötigt um den Beitritt mittels OTAA zu ermöglichen. Mehr zu OTAA kann in dem Kapitel OTAA gelesen werden. Wenn ein Endgerät dem Netzwerk beitreten möchte, leitet der Netzwerkserver die Anfragen an den Join-Server weiter. Dieser führt dann die nötigen Schritte des Beitritts aus wie z.B. ableiten von Schlüsseln oder Senden der nötigen Einstellungen. Um dies zu tun muss ihm der NwkKey und der AppKey bekannt sein, da diese zum Verschlüsseln der Nachrichten verwendet werden aber aus Sicherheitsgründen nie über das Netz übertragen werden dürfen. [SOR17a, S. 9 f.]

## 2.4 End-Gerät

Endgeräte sind Geräte die Informationen mittels LoRa empfangen oder senden. Jedes Endgerät ist mit einem bestimmten Applikationsserver verbunden.

Jedes Endgerät muss zur korrekten Funktion mehrere wichtige Informationen speichern.

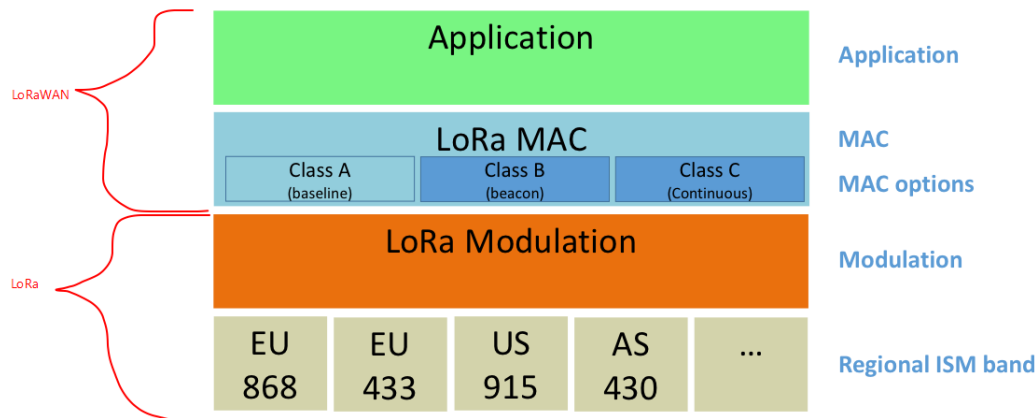
- DevEUI: Globale Endgeräte\_ID die eindeutig für jedes Endgerät definiert ist. Vergleichbar mit der MAC-Adresse eines TCP/IP Gerätes.
- JoinEUI: Globale Adresse des Join-Servers an den die Anfrage gehen soll. Wird nur für OTAA Geräte benötigt.
- NwkKey und AppKey: Werden verwendet um spätere Schlüssel abzuleiten und die Kommunikation während der Beitrittsprozedur in ein Netzwerk abzusichern. Dafür müssen sie sowohl dem Join-Server als auch dem Endgerät bekannt sein da sie nie übertragen werden.

[SOR17b, S.47 ff.]

## 3 LoraWan Funktionsweise

Im folgenden Kapitel wird näher auf die Funktionsweise von LoRaWAN eingegangen. Speziell, liegt der Fokus auf dem Netzwerkeitritt, das verwendete Protokoll und wie die Daten physikalisch Übertragen werden.

### 3.1 Schichtenmodell



Das Schichtenmodell lässt sich in zwei Teile unterteilen. Der LoRa Teil ist der unterste und kümmert sich um die Physikalische Übertragung der Pakete und der LoRaWAN Teil des Modells ist für die Steuerung des Netzwerkes, Implementierung der LoRaWAN-Klassen und das Überprüfen und Verschlüsseln der Daten zuständig.

Die unterste Schicht des LoRa Teils moduliert die verwendeten Frequenzen. In Europa muss das ISM-Band 868 verwendet werden, in den Vereinigten Staaten wird das Band 915 verwendet.

Die darüberliegende Schicht heißt LoRa Modulation und kümmert sich darum, dass die Pakete so in die Frequenz "moduliert" werden, dass der Empfänger diese korrekt und effizient empfangen kann. Mehr dazu im Kapitel Übertragungsart.

Über der LoRa Modulationschicht liegt die erste LoRaWAN-Schicht, LoRa MAC. Diese Schicht ist für die Implementierung der einzelnen Endgeräteklassen und für das Übertragen der Steuerkommandos zuständig. Mehr zu den Klassen kann im Kapitel LoRa Geräte Klassen und im Kapitel Protokoll gelesen werden.

Die oberste Schicht nennt sich Applikationsschicht und ist dafür zuständig, die Nutzdaten einer Nachricht passend zu verpacken, zu verschlüsseln und zu authentifizieren.

[Tec15](Application, LoRa MAC, MacOptions(Classes), LoRaModulation(Regionales ISB))

## 3.2 Netzwerkbeitritt

End-Geräte sind immer bestimmten Netzwerken zugeordnet. Es gibt zwei wege um ein neue End-geräte zu einem bestehenden Netzwerk hinzuzufügen.

[SOR17b](zwei arten OTAA(Over the air activation), ABP(Activation by Personalization) Jedes gerät hat eine vorgegebene DevEUI (wie MAC adresse eines PCs), JoinEUI muss angegeben werden und adressiert den Join server )

ABP steht für Activation by Personalization und bedeutet Wörtlich übersetzt Aktivierung durch Personalisierung.

### 3.2.1 OTAA

Die sicherste aber auch aufwendigste methode um ein End-Gerät mit einem Netzwerk zu verbinden heißt OTAA (Over-the-Air Activation). Hierbei muss jedes mal wenn einem Netzwerk beigetreten werden soll die Join-Prozedur ausgeführt werden. Hierfür müssen folgende 4 Werte Vorgegeben werden. DevEUI, JoinEUI , NwkKey, AppKey(Verschlüsselung des join requests).

Als erstes muss das End-Gerät eine join- oder rejoin-Nachricht senden. Die Nachricht besteht aus der JoinEUI, dem DevEUI und einer DevNonce. Mit der DevNonce sollen replayattacks verhindert werden. Sie ist das beim ersten Join request 0 und sollte sich bei jedem Join-Request erhöhen. Außerdem muss sie auch dann noch gespeichert werden wenn kein Strom zur verfügung steht. Falls von dem gleichen endgerät eine Join-Request mit einer zu kleinen DevNonce empfangen wird, wird die Nachricht ignoriert und es ist nicht möglich dem Netzwerk beizutreten.

Die Accept Nachricht besteht aus einer JoinNonce, einem NetzwerkID Net\_ID, einer Geräteadresse DevAddr, einer einstellungsfeld DLSettings , einer angabe wie lange auf eine antwort nach dem senden gewartet werden muss RxDelay und einer optionalen liste an Netzwerkparametern CFList. Die JoinNonce wird außerdem benutzt im schlüssel wie AppSKey, ... herzuleiten. Für jedes Endgerät wird eine eigene JoinAccept nonce geführt, sie sollte sich nicht wiederholen. Jedes Endgerät merkt sich die letzte JoinNonce und tritt auch nur bei wenn

Was macht der? wohin da- mit? umschreiben !!!!! welche noch, wo beschriebe ich wie?



diese größer ist als die letzte empfangene.

Der NWKSEKEY ist für die Verschlüsselung der Datenpakete bis zu Gateway zuständig. Auch dieser Key wird vom Netzwerkservers erzeugt und muss manuell in den Code eingetragen werden.

Der letzte Wert heißt APPSEKEY und sichert die Kommunikation vom Endgerät zu dem Applikationsserver ab. Der Schlüssel wird genau wie der NWKSEKEY vom Netzwerkservers erzeugt und verwaltet.

Mehr Informationen zu den verschiedenen Schlüsseln finden Sie in dem Kapitel Sicherheit.

Wenn der Netzwerkservers den Beitritt des Endgerätes erlaubt sendet er eine Join-Accept Nachricht zurück. Das Endgerät erwartet die Nachricht nach JOIN\_ACCEPT\_DELAY1 oder JOIN\_ACCEPT\_DELAY2 nach dem Senden des Request. Sollte die Join-Accept Nachricht zu einem anderen Zeitpunkt gesendet werden, wird diese nicht empfangen, weil das Endgerät nicht empfangsbereit ist. Die Nachricht enthält Einstellungen für das Endgerät sowie die ID des Netzwerkes und die neue Adresse für das Endgerät. Um Replayatacken zu verhindern enthält die Nachricht zusätzlich eine JoinNonce. Diese wird für jedes Endgerät separat geführt und muss größer sein als die zuletzt gesendete.

### 3.2.2 ABP

Die einfachste Art des Beitritts heißt ABP was für Activation by Personalization bzw. deutsch Aktivierung durch Personalisierung steht. Hierbei muss lediglich vor Inbetriebnahme des End-Gerätes 3 Konstanten definiert werden. Manche Hersteller "brennen" diese drei Werte fest in den Chip ein, sodass er nicht geändert werden kann. Falls es nicht möglich ist dem Hersteller die gewünschten Werte zukommen zu lassen, sind solche End-Geräte nicht für den Beitritt mittels ABP geeignet.

Als erstes muss die DeviceAddress angegeben werden. Diese Adresse existiert nur einmal im Netzwerk und wird verwendet, um das Endgerät zu identifizieren. Die Adresse wird vom Netzwerkservers erzeugt und muss manuell von dort kopiert werden. Die verwendete Frequenz entspricht der RX1 bzw. RX2

aus dem kapitel Klasse A.

Mit hilfer dieser 3 Werte kann die Join-Request - Join-Accept prozedur übersprungen werden. Daher kann das Gerät direkt einen Loranetz beitreten wenn es angeschalten wird und muss nicht erst alle schlüssel neu ableiten und aushandeln. Allerdings ist diese methode deswegen weniger sicher, da immer die selben schlüssel verwendet werden.

Nach der verwbindung muss das ResetInd Mac command im FOpt feld gesendet werden gesendet werden solange bis ein ResetConf erhalten wird. hab NUn ist das Gerät im netzwerk und kann unter der eingestellten Adresse und noch mit den eigestellten schlüssel arbeiten. nci

Sobal dem Netzwerk erfolgreich beigetreten wurde werden ein die benötigten von schlüssel aus dem vorher gesetzten Werten abgeleitet. genauers dazu in kapitel mac Sicherheit. ge-spro-chen

### 3.3 Protokoll

Das LoRaWAN Protokoll ist optimiert für Batteriebetriebenen Endgeräte die drahtlos kommunizieren möchten. Um energieeffizient zu sein setzt LoRa hauptsächlich auf zwei Punkte. Die Modulationstechnik und eine Adaptive Datenrate (ADR). Auch die Öne-Hoparchitektur trägt zur energieeffizienz bei. Die Art wie LoRa signale Moduliert wird in kapittel Übertragungsart besprochen. Um die gannannten Eigenschaften und das Lora Netzwes zu steuern werden sogenannte MAC commands verwendet. Diese werden von Dem Netzwerkserver oder von einem Endgerät gesendet. MAC steht hierbei für "Media Access Protokoll und bietet die möglichkeit die kommunikation mit den endgeräten, frequenzen, kanäle und vieles mehr zu stuern. Da die kommandos nur für den doppeltgemoppelt Netzwerkserver und die Endgeräte von bedeutung sind, werden diese nicht an den Applikationsserver gesendet sonder om Netzwerkserver herausgefiltert. Im folgenen wird näher auf die MAC komannndos und die Paketstruktur eingegangen.

Jedes Pakete besteht aus grundlegend aus 2 Felder (Preamble, PHYPayload). Falls es sich um einen Uplinkpaket handelt wird noch ein CRC code erkläre up und down-link

hinzugefügt (Preamble, PHYPayload, CRC). In diesem fall spricht man von einem implizit Paket oder von dem implezitem modus. Implizit modus bedeute das das es kein payload, codirungsrate und der CRC längenangebe gibt und diese somit eine feste zuvor definierte länge haben. Im expliziten modus werden noch 2 Felder hinzugefügt, PHDR und PHDR\_CRC. SOmit sihet ein explizitest paket folgernermaßen aus( Preamble,PHDR, PHDR\_CRC, PHYPayload). AUch hier gilt, im alle eines Uplinkpaketes wird am ende ein CRC felad angefügt => ( Preamble,PHDR, PHDR\_CRC, PHYPayload, CRC).

Die Preamble ist dafür gedacht dem Empfänger mitzuteilen das gleich Datengesendet werden. Deswegen wird hier nur ein Signal gesendet das ohne Informationen ist, aber von dem Empfänger wargenommen wird.

Da Tiele des LoraWAN protokols geschützt sind, finden sich über die PHDR und PHDR\_CRC felder kaum informationan . ALlerdings geht herfor das der PHDR die länge des PHYPayloads und die Zieladresse beinhalten sollte. Das PHDR\_CRC Feld wird benutzt um sicherzustellen dass bie empfangenen Werte korekt sind mittels ddes CRC verfahrens .

Wie schon merhfach erwähnt wird in Uplinknachrichten ein zusätzliches CRC feld verwendet. CRC steht für Cyclic Redundancy Check und wird verwendet um die korrektheit der NACHricht zu bestätigen. PHDR, PHDR\_CRC und das CRC Fled werden automatisch vom dem Funktransceiver (modul asu empfänger und sender) hinzugefügt.

Die bis jetzt ebhandelten Felder des LoRa Paketes wurden alle von der LoRa Modulationsebene erstellt.

Die Darüberliegende Ebende "LoRa Mac"fügt nun das PHYPayload Fled ein. PHYPayload setht für Physikalische Payload. Es gibt 3 Mögliche PHY Payloads . Entweder wird ein MACPayload eingefügt oder es werden join-rejoin request oder aber es wird die join accept nachricht darin transportiert. Um die Daten bzw die MAC kommandos richtig auswerten zu können und um die korrektheit überprüfen zu können werden einige Headders und zusätzliche felder benötigt. Deswegen lässt sich das Feld weiter unterteilen in (MHDR, MACPayload). Für den Fall das der MACPayload keine join-rejoin oder MacPayload nachricht ist, wir noch ein MIC feld hinzugefügt (MHDR, MACPayload, MIC).

MIC steht für Message Integrity Code und wird verwendet um die korrektheit der Unterfelder MHDR | FHDR | FPort | FRMPayload festzustellen. Diese unbekannten felder werde im laufe des kapitells noch behandelt.

Das MHDR Feld beschribt wie die Daten im MACPayload Feld zu deuten sind. Wieder wird dieses Feld in Unterfelder Unterteilt. MTType, RFU und Major heisen die Unterfelder. Das MType feld beschreibt die Art der Nachricht. z.B: kann hier angegebn werden ob es sich um Datennachrichten, Join-Nachrichten, ... handelt. RFU steht für "Reserved for Future Usage" Deutsch "für zukünftige verwendung reserviert". Daher kann dieses Feld in der version 1.1 und niedriger ignoriert werden. Im Major Unterfeld wird verwendet um das Format der Nachricht zu definieren. Momentan ist nur der wert 0 Definiert. 0 Steht für loRaWan R1. Die restlichen werde sind für zukünftige updates reserviert.

Mit der Unterteilung des MACPayload springen wir in den LoRaStack noch eine ebene höher in die Applikationsschicht. Enthalten im MacPayload feld sind der Frameheader (FHDR), der Frame Port (FPort) und der Frame payload (FRMPayload). Daten die gesendet werden sollen befinden sich in dem FRMPayload Feld. Wenn keine Daten gesendet werden, kann das FRMPayload Feld auch MAC kommandos enthalten. In dem Feld FPorts wird angegeben an welchen port und somit an welche teilapplikation die Daten geleidet werden. Es gib einige feste Ports. z.B. Port 0 Zeit an das das FRMPayload Feld MAC commands enthält, 0x01 bis 0xDF sind Anwendungsspezifische Ports und Port 244 ist für das LoRaWan Test Layer protokoll reserviert. Falls ein andere Port als die geraden genannten angegeben wird, wird die nachricht verweorfen. Erneut kann der FHDR "Frame Header" in einzelne Felder unterteilt werden (DevAddr, FCtrl, FCnt, Fopts).

In dem feld DevAddr wird die Zieladresse der Nachricht vermerkt. Im feld FCnt (Frame counter) wird der jeweilige counterwert für die bisher gezählten Nachrichten übermittelt. Damit schützt man sich vor replay Attacken. Im FOpt feld können bis zu 5 MAC kommandos parallel zu Daten übermittelt werden. Die Anzahl kommt auf die menge der mitgelieferten variablen an. Das Letzte feld das in Unterfelder unterteilt wird ist das FCTRL feld. Hier wird das

verhalten des Gerätes gesteuert sowie nachrichten acknoleged. Es gibt leichte unterschiede für ein Up-Link und für Down-link Nachrichten. Beide Nachrichtenentypen haben ein ADR, ein ACK und ein FOptsLen feld. Im ADR wird definiert ob der sendende bereit ist im Modus "Adaptive Data Rate" Daten zu senden, siehe Adaptive Data Rate. Mit dem Ack Feld können empfangene nachrichten markiert werden. Ob Nachricht bestätigt werden müssen steht im MType feld. In dem FOptsLen feld wird die länge des FOpts feldes mitsamt des Headers eingetragen .

da

Ein Downlinkpaket hat zusätzlich ein RFU feld das nicht verwendet wird und ein FPending feld. In diesem feld kann das Gateway bzw der Netzwerks server dem Endgerät mitteilen, dass noch mehr Daten zu senden sind und mehr empfangsfenster geöffnet werden müssen.

steht

noch

was

von

Dahingegen hat ein Uplinkpaket ein ClassB feld indem das endgerät dem Gateway mitteilt, dass es gerne auf Funktionsklasse B wechseln würde und ein ADRACKReq feld. Dieses feld wird verwendet um zu überprüfen ob das Netzwerk noch antwortet. .

port

und

so!!

erklären

erklären!

[SOR17b] ( Mac commands werden benutzt um geräte zu steuern => frequenzen zu ändern, ... Application wird diese nie erhalten, läuft zwischen netzwerkserver und lora gerät ab. Verschlüsselt hier oder da. aufbau: 1byte command, x byte extra data. müssen vom empfanher acknoleged werden. Reihenfolge ist zu beachten. Alle nachrichten in einem frame müssen auch in einem frame ack werden. => Macbuffer ermöglicht dies. Wenn buffer überleuft werden die ältesten ack.

(Was

CRC usw wurden von sender erstellt und eingefügt. )

pas-

[CB<sup>+</sup>17] ( Um energieeffizient zu sein setzt LoRa hauptsächlich auf zwei Punkte. Die Modulationstechnik und eine Adaptive Datenrate (ADR) )

siert

mit

dem

rest?

### 3.4 Übertragungsart

Um die Enstandenen Pakete in Signale umzusetzen und gleichzeitig übertragen zu können nutzt LoRa Chirp Spread Spectrum (CSS). Hierbei werden die Frequenz über eine gewisse Zeit hinweg verändert. Durch erkennen

in welche richtig, ansetigen oder Abfalleb, die frequenz verändert wird, können 1 und 0 Codiert werden. Man spricht bei einem Bit von einem Chrip-Impuls. Durch aneinanderreihe der verschiedenen impulse ist es möglich mehere Bits nacheinander zu übertragen. Das entstandene Signal wird auch al Sub-Chrip bezeichnet. Durch verwenden von Unterschiedlichen anstiegs und abfalszeiten ist es möglich mehere Signale auf der Selben frequenz zu übertragen ohne das die Signale sich gegenseitig stören. Dies nennt man Spread Factor. Auserdem kann die Paralelität durch verschiedene Frequenzbereiche verbessert werden. CSS ist besonders für große reichweiten geeignet und somit auch bestens für Lo- nachprüfen ra. Am besten ist das Signal wenn das endgerät nahe am Gateway ist. Je weiter es entfernt desto schlechter wird das signal. Um die kommunikation trotzdem zu ermöglichen wird der Spredding Factor erhöht. Dies hat auch den Vorteil dass der Energieaufwand gering gehalten wird. Analog wie Menschen auf einer Party nicht immer versuchen lauter zu sprechen, sonder auch versuchen besonders langsam und deutlich zu sprecehn.

Mann sichbrcht auch von Channels. Channels können beliebeig benutzt werde. es gibt alerdings zwei regeln zu beachtn.

1. Channels werden per Pseudozufallszahl geändert
2. Sendezeit erfüllt die Regionalen Bestimmungen

Das Aloha Protokoll wird verwendet um festzustellen wann gesendet werden soll. Dabei wird einfach gesendet wenn Daten zum senden vorhanden sind. Wenn nun zwei Sender gleichzeitig auf der selbern Frequenz senden möchten kommt es zu einer kollision. Dadurch kann das Gateway die empfangenen Dta- ne nicht mehr auswerden. Deswegen warten beide Endgeräte eine zufällige, unterschiedliche Zeit abe bist sie erneut senden.

[SOR17b]( Knoten können zu jeder Zeit, auf beliebigen Kanälen, beliebig schnell, beliebig lange senden, solange folgende regeln befolgt werden.

- Channels werden per Pseudozufallszahl geändert
- Sendezeit erfüllt die Regionalen Bestimmungen

) [SOR17b]( Geschwindigkeit ist kompromiss zwischen abstand/geschw. die untersch freuenzen bze. geschwindigkeiten beeinflussen sich nicht gegeseitig => keine inteferenz Die Datenrate ist einstellbar, jedoch wird die Reichweite bei höherer Datenrate gemindert. Ein Vorteil von Lora ist, das die einzelnen Datenraten nicht interferieren und so jedes Endgerät seine eigene Datenrate unabhängig von den anderen Verwenden kann. Außerdem wird die Datenrate und die Sendeleistung für jedes Gerät separat gesteuert (ADR, Adaprive Data Rate) )

[AVTP<sup>+</sup>17]( Chrip Signal => Zeitliche Änderung in Trägerfrequenz(höhere Frequenz als Datenrate)(positiv chirp/negativ chirp)

Datensignal wird in Chrip Signal moduliert. Resultierende Signal ist breitbandiger als Datensignal. Maximale Datenrate auch mit Rauschen erreichbar.

Durch orthogonale SSpread Factor"mehrere Signale auf einem Chanel ) [Tec15](normal FSK, schon sehr efficent. Lora "chirp spread spectrum odulation". Ist wie FSk aber größere Rechiweite, robuster. Stammt aus dem Militär/raumfahrt.Lora als erstes für kommerziellen billigen Einsatz.

Spread spectrum => signale sind Ortohonal für versch. spreizraten, fakto koreliert mit datenrate => verschiedene Datenraten auf einem Kanal

Nähere Geräte sind schneller => höhere Datenrate => kürzee übertragungsdauer und lassen somit merh zeit für andere, => bessere Batterielaufzeit. Deswegen sidn symetrische up/downlinks nötig.) Frequenzhopping, spread spectrum, code-channels

### 3.4.1 Adaptive Data Rate

Adaprive Data Rate oder kurz ADR wird verwedent um immer die optimalste senderate und die optimale sendepower für das Endgerät zu finden und so schnellstmöglich die Daten zu senden. ADR kann nur verwendet werden wenn im FHDR feld des LoraPaketes das ADR Bit gesetzt ist, siehe Protokoll. Die Steuerung duch ADR findet durch den Netzwerksrerver statt. Sobald der NNetzwerksrerver bereit ist, stzt er das bit im downlink. IST das endgerät ebenfalls bereit setzt es ebenfalls das bit und ADR kann verwendet werden. Flass es

soll ich  
ver-  
hält-  
niss  
zwi-  
schen  
data-  
rat sf  
und  
ener-  
gie  
rein-  
ma-  
chen?

nicht möglich sein sollte ADR zu verwenden sollte es durch das Applikationslayer gesteuert werden. warum?

Die Steuerung findet durch spezielle MAC kommandos statt. Standardgemäß wird die höchst übertragungsstärke verwendet allerdings auch die geringste Übertragungsrate. Falls diese gedrosselt werden soll wird vom Netzwerkservers das **LinkADRReq** MAC command benutzt. Mit diesem wird das Endgerät informiert, dass es die data rate, transmit power, repetition rate or channel ändern soll. Was auf welchen wert geändert werden soll wird in die Parameter codiert. Sobald die Werte geändert wurden, muss periodisch überprüft werden ob das Netzwerk die Nachrichten noch bekommt. Deswegen wird jedes mal wenn der uplinkframecount erhöht wird, wird der ADR\_ACK\_CNT counter verwendet. Wenn dieser counter ein gewissen schwellenwert (ADR\_ACK\_Limit) überschreitet, wird das ADRACKReq bit gesetzt. Dieses signalisiert den Netzwerkservers das er mit einem Uplink ein Downlink senden muss um die Verbindung zu bestätigen. Falls dieser Downlink nicht in ADR\_ACK\_Delay frames empfangen wird, wird zuerst die übertragungsstärke auf max gesetzt. Falls möglich wird ausserdem die Datenrate verringert um die Reichweite zu erhöhen. Die Datenrate wird solange weiter jede ADR\_ACK\_Delay frames verringert bis diese minimal ist. Falls diese schon minimal ist müssen alle channels wiederverwendet werden. Dies wird solange probiert bis eine verbindung hergestellt werden kann.

[AVTP<sup>+</sup>17]( Datenrate und Funkfrequenz(RF) werde passend zum Abstand angepasst

nahe Knoten => hohe Datenrate => kurze Sendezeit => weniger RF-Power kann nach Bedarf geändert werden

=> immer möglichst schnelle senden => weniger Energie

) [CB<sup>+</sup>17](crip signal)

## 4 Lora Geräte Klassen

Um maximal energie zu sparen aber trotzdem die möglichkeit dass die endgeräte agiel Daten empfangen können wurden die Geräteklassen eingeführt.



Das Hauptmerkmal der Klassen sind die unterschiedlichen empfangsmodien. Es gibt 3 Klassen, A, B und C. Die Klasse A muss standartgemäß von jedem Endgerät implementiert werden. B und C sind Optional und müssen nicht vorhanden sein . Alle Geräte die mehr als A können werden als "high class End-Devices" genannt. Joinen nur in

[SOR17b](Geräte müssen mindestens A können, alle die mehr können werden auch "high class End-Devices" genannt) Vielleicht zu klein => in anderes Kapitel stopfen. Bei mehrfacher übertrageung wird nicht erhöht A be-schrie-ben?

Die Endgeräte sind je nach Kommunikationsart/Protokoll Art in drei Klassen (A, B und C) unterteilt. wohin mit

Jede Klasse hat 3 counter FCntUP(Pro uplink ++), FCNTDown(pro downlink auser port 0 => mach), AFCntDown(port ungleich 0 dann ++) (nur beschreiben wie diese grob funktionieren) Zähler sollen nicht flüchtig sein(Batteriewechseln kein reset) bei neuverbinden müssen alle counter auf uf 0 gesetzt werden. counter müssen auf beiden seiten glich gehalten werden(Synchron geführt) Wenn nachricht empfangen ist muss der darin enthaltenene counter größer sein als der eigene. coun-ter?

die Counter Werte sollen so weit wie möglich nur einam verwendet werden.

) [Tec15](Asynchrone Knoten wegen Batterie => Event/Scheduler gesteuert verwendet ALOHA

Normal Netze müssen sich synchronisieren und Nachrichten abrufen. Lora partiell nicht => laut GSMA 3 bis 5 fach effizienter)

zur besseren Anpassung/ Anpassung an Batterie

EU: 10 Kanäle (8: 250bps bis 5.5kbps) (1: FSK 50kbps) (high rate Lora 114kbps)  
)

## 4.1 Klasse A

Klasse A wird auch (All end-Devicec) gennnt undzeichnet sich durch sehr geringer Stromverbrauch aus. Die Kommunikation kann bidirektionalen Stadtfinden, allerdings muss die Kommunikation von dem Endgeräte gestartet werden.

Das bietet die Möglichkeit das das Endgerät, wenn keine Daten gesendet werden müssen, in einen sehr sparsamen Schlafmodus wechselt. Um das Endgeräte nicht zum aufwachen zwingen zu müssen, wurde auf einen Hardbeat oder ähnliches verzichtet. Dadurch kann das Endgerät so lange "schlafen" wie es möchte. Somit ist die Klasse A auch die potenziell Stromsparende Endgeräteklasse. Die Klasse A erlaubt außerdem das das Endgerät andere Protokolle schickt solange es keine LoRa Daten sendet oder empfängt.

Das Endgerät startet die Kommunikation in dem es Daten an das Gateway sendet (uplink). Daraufhin hat das Gateway die Möglichkeit 2 mal Daten zum Endgeräte senden (downlink). Die Downlinkfenster werden RX1 und RX2 genannt. Da die Kommunikation asynchron stattfindet, muss das Endgerät warten bis die uplinkphase abgeschlossen ist.

Die empfangsfenster RX1 und RX2 müssen mindestens solange geöffnet bleiben das sie eine beginnende Übertragung feststellen können. Falls keine Übertragung empfangen wird, wird das fenster wieder geschlossen. Anderenfalls werden die Daten empfangen. Das Empfangsfenster RX1 wird nach RECIEV\_DELAY1 zeiteinheiten  $\pm 20\text{msec}$  nach beendigung des Uplinks geöffnet. Es wird die selbe frequenz und Datenrate verwendet die auch bei den uplink verwendet wurde. Wenn festgestellt in RX1 festgestellt wurde das keine weiteren Daten mehr empfangen werden müssen kann auf das öffnen des RX2 fensters auch verzichtet werden. RX2 wird nach RECIEV\_DELAY2 zeiteinheiten  $\pm 20\text{msec}$  nach beendigung des Uplinks geöffnet. Allerdings ist die Datenrate und frequenz fest. Nur Mittels spezieller MAC commands kann dies verändert werden.

Für alle join / rejoin aktivitäten wird immer die Klasse A verwendet . Schon

[SOR17b] (radio packet explicit mode, vom Gateway(1) zum Knoten(1), erklärt ausgelöst vom Netzwerkservers, auch multikasts möglich, (Preamble, PHDR, oder PHDR\_CRC, PHYPayload) Um Nachricht kurz zu halten kein CRC am ende, woan- nach Reciever\_Delay1 / Reciever\_Delay2 kann empfangen werden (rx1, rx2) ders

Fenster müssen lange genug für Preamble auf bleiben=> wenn erkannt wird empfangen wenn nicht fenster wieder zu. Es darf nur gesendet werden wenn beide fenster zu sind. ==> Es ist auch erlaubt andere protokolle zu

sprechen wenn nicht gesendet oder gehört wird.  $\leq$  ) [SOR17b]( Frequenz abhängig von Uplinkfrequenz, Datenrate abhängig on Uplinkdatenrate, wird nach Receiver\_Delay 1 +/- 20 msec erwartet, Datenrate auch abhängig von Regionalen regeln, Standart: Datenrate = Uplinkdatenrate ) [SOR17b]( feste Frequenz/Datenrate, nach Delay2 +/- 20 msec, Frequenz/Datenrate mittels MAC änderbar ) [SOR17b]( Öffnungslänge muss für Preamble ausreichen, nach RX1 + MIC und autenthigitätscheck muss nicht zwingen RX2 geöffnet werden, Sender muss in einem der beiden Fenster stattfinden, Falls Downlink über beide Fenster => frames müssen gleich sein. Knoten dürfen nicht während empfangen/ zwischen RX1 und RX2 senden, ender Protokolle dürfen gesprochen werden wenn gesendet werden darf )

## 4.2 Klasse B

Die Klasse B (B für BEACON) bietet bidirektionale Kommunikation mit einer deterministischem downlink Latenz. Um diese latenz zu gewährleisten, muss die Kommunikation Synchron ablaufen. Außerdem muss festgestellt werden, ob das Endgerät bzw das Gateway noch in Reichweite ist. Dies wird mittels eines periodischem "beacon" die zu festgelegten. Dieser Beacon wird regelmäßig vom Gateway gesendet und dient der synchronisation der Endgeräte. Zeitpunkten gesendet werden realisiert. Die Latenz ist einstellbar und kann bis zu 128 Sekunden. Die Endgeräte öffnen in regelmäßigen Abständen ein empfangsfenster das pingslot genannt wird. Ein Downlink der in einem Pingslot gesendet wird wird ping genannt. Da immer Das gateway mit dem besten empfang die Daten an das Gateway sendet, muss das Endgerät selbständig feststellen wenn es einen Beacon mit einer unbekannten ID bekommt und durch einen uplink dem server mitteilen das es in einer neuen Umgebung ist. Dadurch lernt der server wo sich das Endgerät befindet und kann das Gateway mit dem besten empfang wählen.

Obwohl das Endgerät durch die periodischen "beacons" nicht "schlafen" kann, ist die Klasse B für den Batteriebetrieb gedacht.

[SOR17b](wird verwendet wenn mehr bedarf für empfangsfenster ist. Hier-

zu ist ein synchronsignal nötig=> zu bestimmten zeiten kann damit empfangen werden Gateway sendet Beacon für synchronisation. Um daten empfangen zu werden werden empfangsslots => pingslots verwendet, werden periodisch geöffnet und mittels beacon synchronisiert. Normalerweise werden diese schnell geschlossen außer es wird etwas empfangen. Gateway dessen beacon benutzt wird, wird nach empfangsqualität ausgewählt. Wenn neuer/unbekannter Beacon von einem anderen Gateway empfangen wird, wird der netzwerkserver benachrichtet und dieser entscheidet welcher verwendet wird (passt rote an).

Das Netzwerk muss die standard ping-slot periode Datenrate und kanal kennen.

Um ein gerät auf klasse B zu kommen muss erst von Klasse A gewechselt werden.

Entgeräte müssen Netzwerkserver über position informieren. Dies kann über eine leere nachricht passieren oder eine normale (uplink).

Das beacon und die enthaltenen daten werden an die applikation geschickt. Der server kann den beacon auswerten. zwischen beacon und uplink wird random time verwendet um kolisionen zu verhindern. änderungen an pingslot-periode .. muss mitgeteilt werden. Hierzu ist klasse A nötig => wechsel zu A, wechsel zu B.

Nachschuen  
wie ge-  
nau  
das  
funk-  
tio-  
niert

Beacon wird genutzt um clockdrift auszugleichen. Wenn kein beacon empfangen wird => Beaconless mode. Dieser wird bis zu 2 stunden beibehalten. Reines verlassen auf interne Uhr. Wenn beacon empfangen wird, wird zeit zurückgesetzt. )

#### 4.2.1 Klassenwechsel A nach B

Um einen Wechsel überhaupt zu ermöglichen muss der Netzwerkserver die default ping-slot perioden die pingslot datenrate und den Pingslot channel kennen.

Alle endgeräte treten in Klasse A dem Netzwerk bei. Das wechseln in die klasse B wird durch folgenden Prozess realisiert.

Als erstes muss das Programm des Endgerätes beim LoRaWAN layer an-

frohen ob es möglich ist in klasse B zu wechseln. Der Layer sucht nun nach einem beacon. Wird ein beacon entdeckt, wird die BEACON\_LOCKED Serviceprimitive zurückgeliefert. Wenn kein Beacon empfangen wurde wird die BEACON\_NOT\_FOUND primitive zurückgegeben. Um diesen Prozess zu beschleunigen kann das DeviceTimeReq MAC kommando verwendet werden. Damit wird das Gateway aufgefordert ein beacon zu senden. Nun kann das endgerät in den modus B wechseln.

Als Nächstes setzt der MAC Layer des endgerätes das Class B Bit im FCtrl feld des Uplinks auf 1. Dadurch ist er auch verantwortlich die Ping slots und für die Beacons zu öffnen. Dabei muss mit der größt möglichen abweichung der internen Uhr gerechnet werden und dementsprechend die Empfangsfenster angepasst werden. Diese darf pro Beacon nicht mehr als +/- 1.3msec liegen. Der Inhalt der empfangenen Beacons wird mit der Signalstärke und das Programm des Endgerätes zur weiteren Verarbeitung gesendet. Damit kann z.B. dem LoRaWAN layer angewiesen werden die Uhr nachzustellen.

[SOR17b] (Endgerät fordert LoRaWAN layer an. Layer sucht beacon. Mac command DeviceTimeReq um schneller beacon zu bekommen nutzen. Danach wird das ClassB feld auf 1 gesetzt. Bei den geöffneten fenstern wird der maximal mögliche clockdrift berücksichtigt. Downlink läuft wie bei A ab.

)

#### 4.2.2 Betrieb

Damit der Netzwerkserver dem Endgerät mitteilen kann dass die pingslots frequenz und/oder die Datenrate geändert werden soll gibt es den PingSlotChannelReq Mac kommando. Die werden sind in den argumenten enthalten.

Das Endgerät kann die Periode der Pingslots zu einer beliebigen Zeit ändern. Ist dies der Fall, so muss das Endgerät in Klasse A wechseln mit mittels dem MAC kommando PingSlotChannelReq die geänderte periode mitteilen. Danach kann zurück in Klasse B gewechselt werden.

Falls einige länger als 2 Stunden kein Beacon empfangen wird, kann die synchronisation mit dem Netzwerk verloren gehen. Dadurch funktioniert die Kom-

1/s  
=> Hz

munikation in Klasse B nicht mehr und es wird in Klasse A gewechselt. Da sich nun die Kommunikationsstrategie verändert muss mit einem Uplink in dem das ClassB Fled 0 ist, der Netzwerkservers informiert werden. Nun kann versucht werden eine Verbindung mit der Klasse A aufzubauen. Das Programm des Endgerätes kann versuchen wieder in Klasse B zu wechseln. Dieser Prozess kann sich immer wieder wiederholen.

Um auch innerhalb der maximal 2 Stunden in denen kein Beacon empfangen wurde eine Kommunikation zu ermöglichen wird jedes mal wenn ein Beacon verloren geht in den beacon-less Modus gewechselt. Dieser Modus orientiert sich ausschließlich an der internen Uhr. Um den Drift auszugleichen werden die Empfangsfenster immer früher begonnen und immer später beendet. Das bedeutet einen höheren Energieverbrauch aber auch eine höhere Wahrscheinlichkeit noch Daten zu empfangen obwohl die Uhren des Gateways und des Endgerätes auseinanderlaufen.

drift?

### 4.2.3 Singel / Multicast

Die Downlink der Klasse B unterscheidet sich nicht von denen der Klasse A. Allerdings kann sich der Frequenzplan unterscheiden.

In Klasse B können die Nachrichten als Singelcast oder als Multicast Nachrichten verwendet werden. Eine Singelcast Nachricht wird an das Gerät das im DevAddr Feld der Nachricht codiert ist gesendet. Im Multicastmodus wird das Paket an alle Endgeräte gesendet. Damit die möglich ist müssen sich die Geräte die selbe Multicast Adresse und die dazugehörigen Schlüssel teilen. Durch verschiedene Multicastadressen ist es möglich sogenannte Multicastgruppen zu erzeugen die nicht alle sondern nur ein Teil aller Endgeräte beinhalten. LoRaWAN gibt allerdings keine Methode vor wie die Adressen und Schlüssel verteilt werden. Diese Aufgabe muss also in der Applikationsebene sprich im Programm der Endgeräte oder Direkt bei der Personalisierung (Programmierung) erledigt werden.

In Multicastadressen sind keine MAC Kommandos erlaubt. Nur Daten dürfen als Multicastnachricht übertragen werden. Dies wurde eingeführt da Multi-

castnachrichten nicht die selbe robutheit wie Singelcastnachrichten haben. Die Nachrichten dürfen nicht acknologed werden . Das Fpending zeigt an das mehr Multicasnachrichten zu senden sind. [SOR17b]( sepearate Adresse für Multi- cast Festgelegt durch layer oder manuell für gruppenmulticast Nicht führ MAC geeignet, )

#### 4.2.4 Beacon

Wie schon erwähnt wird der Beackon verwendet um das ENdgerät mit dem Netzwerck zu synchronisierern. Deswegen wird dieser Periodisch gesendet. Die Zeit zwischen zwei Beackons wir BEACKON\_Period genannt. Die Endgeräte öffnen Enmpfangsfenster um diese Beacons zu empfangen. Ein BAcok zu übertragen dauert BEACON\_RESERVED lange. Das beaon wird Beakon\_GUARD früher geöffnet um sicher zu stellen das BEac auch wirklich zu empfangen. Während versucht wird ein Beacon zu empfangen kann kein pingslot geöffnet werden. Auserdem wird die Beakon\_GUARD benutzt um sicherzustellen das kein Ping slott mehr geöffnet ist. Deswegen muss diese Beakon\_GUARD mindestens so lang sein wie ein maximaler pingslot. Ein weiterer vorteil ist, dass nicht daraf geachth werden muss wann ein pingslot geöffnet wird, da er sowieso im zweifelsfall fertig ist befor ein beakon empfangen wird.

Um snychronsisierungen druch die beacons zu vermeiden, wie alle entgeräde wollen sofort nach den beakon senden wollen, wird mittels zufälliger warte, pingslot zeiten und zufälliger pingslotanzahlen verhindert.

Beacons haben ihr eigenes Paketformat. Diese PAKete sind immer gleich lang. Dadurch kann auf header verzichtet werden was auch der Geschiwndigkeit der verarbetug zu gute kommt. Wie auch ein Normales LoRaPaket, so besteht auch das erste Feld des Beakonpaketes aus der Preamble nur das die des Beakonpaketes länger dauert was ein bemerken der übertragung warscheinlicher macht. Danach folgt nur noch der BCNPayload. Der BCNPayload lässt sich unterteilen in RFU, Time, CRC, GWSpecific, RFU, CRC. Die zwei CRC Felder weisen schon auf die logische unterteilung in zwei hälten hin. Der erste Teil enthält beacon speziische informationen (time und CRC). In dem Timefeld

ist die zeit seit 00:00:00, Sunday 6th of January 1980 (start of the GPS epoch) modulo 2<sup>32</sup> enthalten. das CRC feld wird verwendet um die korrekttheut des Zeit und des RFU Fledes zu versichern. Die andere hälfte ist GAtewayspezifisch. Sie enthält das GwSpecific fled und ein RFU fled das auch dirch ein zweites CRC feld abgesichert ist. Das GwSpezific feld lässt sich unterteilen in InfoDesc und Info felder. Das InfoDesc gibt an auf was sich das Infofeld bezieht.0 GPS coordinate of the gateway first antenna 1 GPS coordinate of the gateway second antenna 2 GPS coordinate of the gateway third antenna 3:127 RFU 128:255 Reserved for custom network specific broadcasts. Sonlage sich im infofeld koordinaten enthalten kann dieses unterteilt werden in Längen und Breitengrad.

Auch Klasse A kann den beacon somit nutzen um herauszufinden von welchem gateway es gerade Datenempfängt und um somit eventuelle Standortwechsel festzustellen.

In Europa werden die Beacons auf einer festen Frequenz übertragen die sich nicht ändert außer über das MAC Kommando PingSlotChannelReq. Auf anderen Kontinenten kann es sein dass Frequenzhopping angewendet wird.

regionale  
para-  
merter  
er-  
wähnt?

### 4.3 Klasse C

C steht für CONTINUOUSLY Listening. Wie der Name schon sagt wird hier unaufhörlich ein empfängssender geöffnet. Dadurch wird es ermöglicht fast Latenzfrei zu übertragen. Dies bedeutet aber auch dass der Stromverbrauch am höchsten ist und somit nicht für den Batteriebetrieb geeignet. Das Gateway kann immer Daten senden außer wenn das Endgerät gerade Daten sendet. Hier sind Geschwindigkeit von bis zu 50mb möglich.

Geräte die Klasse C implementieren sollen aus nicht die Klasse B implementieren das es sonst zu Fehlern kommen kann.

Diese Klasse verwendet die gleichen Empfangsfenster mit den gleichen Funktionen wie in Klasse A. Der große Unterschied besteht allerdings darin dass RX2 immer dann geöffnet ist wenn nicht gerade Daten an das Gateway gesendet werden oder RX1 geeignet ist. Also auch während . Außerdem stehen

suche



die gleichen MAC kommandos und zwei zusätzliche zu verfügbung . normal

Auch in Klasse C ist es, wie in B, möglich Multicastnachrichten zu senden. net?  
hierbei gelten die gleichen Regeln wie bei B.

[SOR17b]( öffnet RX1 und RX2 fenster wie in Klasse A. Immer wenn nicht  
gesendet wird oder RX1 offen ist, ist RX2 offen. Multicast ist auch möglich. )

#### 4.3.1 Wechsel von A nach C

Da es kein ClassC Fled in einem LoRapaket gibt, wurde für das umschalten  
in ClassC mode MAC kommandos eingeführt. Das endgerät sendet das Device-  
ModeInd commando. ALs parameter kann es 0 für Klasse A und 2 Für klasse  
C angeben. Der Netzwerkservers kann mit DeviceModeConf welches den wert der  
klasse enthält in das gewechselt wurde.

## 5 Sicherheit

Sicherheit in netzwerkfähigen Systemen ist ein sehr wichtiges und heiß disku-  
tiertes thema. Da LoRa daten Über die Lpft überträgt, ist es extrem wichtig  
sich und die Daten zu schützen. Da Luft als Medium benutzt wird könnten  
alle in der Nähe befindlichen geräte die gesendeten Daten mithören. Aber ge-  
nauso kann ein Endgerät sich als ein anderes ausgeben und in seinem Namen  
Daten an ein Fremden Server send. Um zu verhindern das Gesendete Daten  
mitgelsen werden müssen diese verschlüsselt werden. Um zu verhindern das je-  
mand anders so tut als wäre er das endgerät müssen die Daten Autentifiziert  
werden. Slest jetzt könnten z.B. Join-request mitgeschnitten werden und von  
dem (bösen) endgerät wiederholt werden um das (gut) endgerät daran zu hin-  
dern aktiv dem Netzwerk beizutreten. Deswegen wurden zähler eingebaut. Im  
folgenden wird sich näher damit beschäftigt welche mechanismen es gibt die  
genannten probleme zu umgehen.

Oberflächlich gesehen bietet Lora eine end-to-end Sicherheit an, indem es  
die Signale zweimal verschlüsselt. Die erste Verschlüsselung dient dazu die ge-  
sendeten Daten vor eventuellen Mithörern zu verschlüsseln, also pm Endgerät  
bis zu Gateway zu verschlüsseln. Die Verschlüsselung geschieht mit einem 128-

bit Network-Session-Key. Die zweite Verschlüsselung wird bis zur endgültigen Weiterverarbeitung der Daten auf z.B. einen Server verwendet und ist ein 128 bit Application-Session-Key.

Näher betrachtet benutzt LoRa eine ganze Reihe an Schlüssel und Zähler verwendet um die Kommunikation abzusichern. Da die verwendeten Schlüssel bei OTAA-Aktivierung variieren wird hier eine viel höhere Sicherheit geboten als bei ABP-Aktivierung wo alle Schlüssel von Anfang an vorgegeben werden. Infolgedessen wird im folgenden Text auf die Sicherheit unter Verwendung von OTAA bezogen.

Jedes Endgerät hat seine eigenen NwkKey (Netzwerkschlüssel) und AppKey (Applikationsschlüssel). Sobald einem Netzwerk beigetreten wurde wird aus dem NwkKey der FNwkSIntKey, SNwkSIntKey und NwkSEncKey abgeleitet. Aus dem AppKey wird zusätzlich der AppSKey abgeleitet. Die Schlüssel müssen so gespeichert werden, dass es nicht möglich ist diese auf irgendeiner Weise aus dem Speicher zu holen außer für das Endgerät selber. Zusätzlich werden Join-Keys abgeleitet. JSInitKey und JSEncKey.

Der FNwkSIntKey ist einzigartig für ein Endgerät und heißt Forwarding Network session integrity key. Der Schlüssel wird verwendet um ganze oder Teile der MIC felder in den LoRaPaketen zu berechnen. mic

Serving Network session integrity key heißt abgekürzt SNwkSIntKey. Dieser Schlüssel wird verwendet um die Integrität des MIC codes zu überprüfen. Zusätzlich wird er auch verwendet um Teile des MIC codes zu berechnen. Dieser Schlüssel ist spezifisch für ein Endgerät.

NwkSEncKey oder lang Network session encryption key, ist für jede Netzwerksitzung einzigartig und wird verwendet um empfangene oder gesendete Mac kommandos zu ent- oder verschlüsseln.

Der AppSKey wird auch Application session key und wird einem Endgerät zugeordnet. Er wird vom Gateway und vom Endgerät verwendet um Daten die zum Applikationsserver geschickt werden sollen zu verschlüsseln.

pad fügt so viele 0en das die Länge an Vielfaches von 16 ist  $\text{AppSKey} = \text{aes128\_encrypt}(\text{NwkKey}, 0x02 \mid \text{JoinNonce} \mid \text{NetID} \mid \text{DevNonce} \mid \text{pad16})$   
 $\text{FNwkSIntKey} = \text{aes128\_encrypt}(\text{NwkKey}, 0x01 \mid \text{JoinNonce} \mid \text{NetID} \mid \text{Dev-}$

Nonce | pad16) SNwkSIntKey = NwkSEncKey = FNwkSIntKey.

Jedes Gerät hat 3 verschiedene Frame counter um die Anzahl der gesendeten und empfangenen Frames mitzuzählen der FCntUP counter zählt die uplinkframes, der NFCNTDown zählt die MAC-downlinkframes und der AFCntDown welcher alle downlinkframes zählt die Nutzdaten enthalten.

Wenn ein Gerät dem Netzwerk beitrifft, werden zuerst die Counter auf 0 gesetzt. Beide Seiten einer Kommunikation halten die Zähler gleich. Beim Senden wird der aktuelle Counterwert in das FCnt Feld eingetragen. Werden Übertragungen wiederholt so wird der Counter nicht erhöht weiderholung

Durch das Verwerfen von Nachrichten mit zu kleinem Counterwert, wird schon verhindert das Pakete von einem Angreifer aufgenommen und zu einem späteren Zeitpunkt wiederabgespielt werden. drin?

Auch bei den Join oder Accept Nachrichten besteht die Gefahr eine Replayattack. Da hier dem Netzwerk noch nicht beigetreten wurde, können die Zähler nicht verwendet werden. Hier wird eine Nonce in die Join-Pakete codiert. Diese Nonce zählt auf die gleiche Weise hoch wie die Counter. Die gegnerische Seite der Kommunikation muss die Nonce tracken und darf nur Pakete mit einer Nonce akzeptieren die höher ist als die letzte Nonce. wie sieht paket aus.

[GAS17](Netzwerkserver hat AppKey daraus werden AppSKey und NwkS-key erzeugt) [Tec15](Applikationsverschlüsselung(schutz der Daten für mitleesen) Netzwerk(Authentifizierung der Knoten) AFS, Key Exchange IEEE 802.11) [SOR17b](symmetrischer Schlüssel => nur einer benötigt, Sessionkey ist abgeleitet von Knoten-rootkey. JoinServer stellt Verbindung der Keys her. ) frequencyhopping

## 6 Live-Beispiel

wenn vorhanden.

## 7 Ausblick

# Literatur

- [AVTP<sup>+</sup>17] ADELANTADO, FERRAN, XAVIER VILAJOSANA, PERE TUSET-PEIRO, BORJA MARTINEZ, JOAN MELIÀ-SEGUÍ und THOMAS WATTEYNE: *Understanding the Limits of LoRaWAN*. <https://ieeexplore.ieee.org/abstract/document/8030482>, September 2017. Eingesehen am 09.04.2019.
- [CB<sup>+</sup>17] CHEONG, PHUI SAN, JOHAN BERGS, , CHRIS HAWINKEL und JEROEN FAMAHEY: *Comparison of LoRaWAN Classes and their Power Consumption*. <https://ieeexplore.ieee.org/abstract/document/8240313>, November 2017. Eingesehen am 09.04.2019.
- [GAS17] GEMALTO, ACTILITY und SEMTECH: *LoRaWAN<sup>TM</sup> SECURITY WHITE PAPER PREPARED FOR THE LoRa ALLIANCE<sup>TM</sup>*. <https://lora-alliance.org/resource-hub/lora-alliance-security-whitepaper>, Februar 2017. Eingesehen am 09.04.2019.
- [SOR17a] SORNIN, N. (Herausgeber): *LoRaWAN<sup>TM</sup> 1.1 Backend(todo*. Lora-Alliance, <https://tools.ietf.org/pdf/rfc8376.pdf>, 1.1 Auflage, Oktober 2017. Eingesehen am 09.04.2019.
- [SOR17b] SORNIN, N. (Herausgeber): *LoRaWAN<sup>TM</sup> 1.1 Specification*. Lora-Alliance, <https://tools.ietf.org/pdf/rfc8376.pdf>, 1.1 Auflage, Oktober 2017. Eingesehen am 09.04.2019.
- [Tec15] TECHNICALMARKETINGWORKGROUP1: *A technical overview of LoRa® and LoRaWAN<sup>TM</sup>*. <https://lora-alliance.org/resource-hub/what-lorawantm>, November 2015. Eingesehen am 09.04.2019.