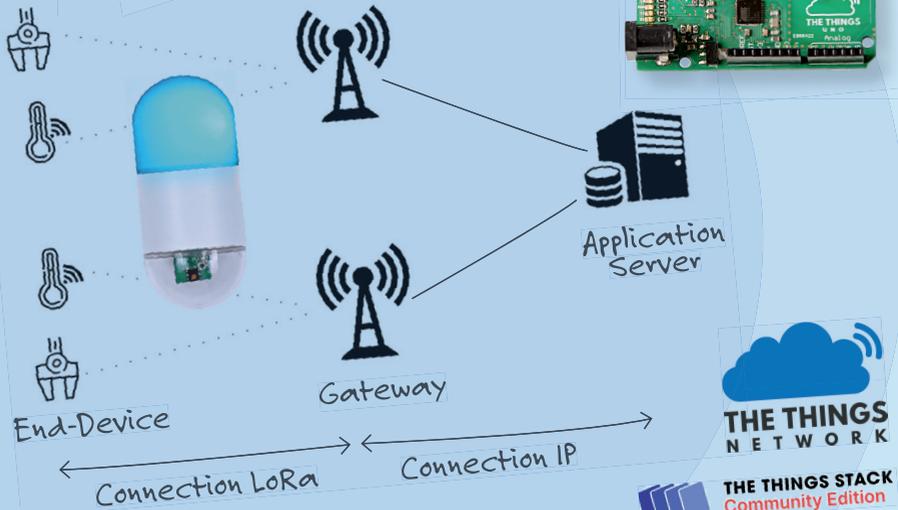


LoRaWAN-Knoten im IoT

Fertige und selbst aufgebaute Arduino-Knoten im TTN



Claus Kühnel

LoRaWAN-Knoten im IoT

Fertige und selbst aufgebaute Arduino-Knoten im TTN



Dr. Claus Kühnel

● © 2021: Elektor Verlag GmbH, Aachen.

1. Auflage 2021

● Alle Rechte vorbehalten.

Die in diesem Buch veröffentlichten Beiträge, insbesondere alle Aufsätze und Artikel sowie alle Entwürfe, Pläne, Zeichnungen und Illustrationen sind urheberrechtlich geschützt. Ihre auch auszugsweise Vervielfältigung und Verbreitung ist grundsätzlich nur mit vorheriger schriftlicher Zustimmung des Herausgebers gestattet.

Die Informationen im vorliegenden Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Die in diesem Buch erwähnten Soft- und Hardwarebezeichnungen können auch dann eingetragene Warenzeichen sein, wenn darauf nicht besonders hingewiesen wird. Sie gehören dem jeweiligen Warenzeicheninhaber und unterliegen gesetzlichen Bestimmungen.

Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autor können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Für die Mitteilung eventueller Fehler sind Verlag und Autor dankbar.

● Erklärung

Autor, Übersetzer und Herausgeber haben sich nach besten Kräften bemüht, die Richtigkeit der in diesem Buch enthaltenen Informationen sicherzustellen. Sie übernehmen keine Haftung für Verluste oder Schäden, die durch Fehler oder Auslassungen in diesem Buch verursacht werden, unabhängig davon, ob diese Fehler oder Auslassungen auf Fahrlässigkeit, ein Versehen oder eine andere Ursache zurückzuführen sind und lehnen hiermit jegliche Haftung gegenüber Dritten ab.

Umschlaggestaltung: Elektor, Aachen

Übersetzung: Andreas Riedenauer

Satz und Aufmachung: D-Vision, Julian van den Berg | Oss (NL)

Druck: Ipskamp Printing, Enschede, Niederlande

● **ISBN 978-3-89576-467-7** Print
ISBN 978-3-89576-468-4 eBook

Elektor-Verlag GmbH, Aachen

www.elektor.de

Elektor ist Teil der Unternehmensgruppe Elektor International Media (EIM), der weltweit wichtigsten Quelle für technische Informationen und Elektronik-Produkte für Ingenieure und Elektronik-Entwickler und für Firmen, die diese Fachleute beschäftigen. Das internationale Team von Elektor entwickelt Tag für Tag hochwertige Inhalte für Entwickler und DIY-Elektroniker, die über verschiedene Medien (Magazine, Videos, digitale Medien sowie Social Media) in zahlreichen Sprachen verbreitet werden. www.elektor.de

Kapitel 1 • Vorwort	9
Kapitel 2 • LoRa	13
2.1 LoRa-Grundlagen	13
2.2 LoRa-Reichweite	19
2.3 LoRa-Kommunikation	22
Kapitel 3 • LoRaWAN	24
3.1 LoRaWAN-Grundlagen	24
3.2 LoRaWAN-Geräteklassen	25
3.2.1 Class A	25
3.2.2 Class B	26
3.2.3 Class C	26
3.3 LoRaWAN-Kommunikation	26
3.4 LoRaWAN-Service Provider	27
3.5 LoRaWAN-Gateway	28
3.6 LoRaWAN-Endgeräte	28
Kapitel 4 • The Things Network (TTN)	30
4.1 TTN-Infrastruktur	30
4.2 The Things Stack Community Edition TTS (CE)	32
4.3 Eigenes LoRaWAN-Gateway	33
4.3.1 Heltec HT-M00 Dual Channel LoRa Gateway	34
4.3.2 Dragino LPS8	38
4.4 LoRaWAN-Endgeräte anmelden	41
4.5 Duty Cycle & TTN Fair Use Policy	46
4.6 Payload	48
4.7 Integrations	52
4.7.1 MQTT	53
4.7.2 Node-Red	55
4.7.2.1 Node-Red Installation unter Windows	56
4.7.2.2 Node-Red Installation auf Raspberry Pi	58
4.7.2.3 Node-Red Anwendung	61
4.7.3 Cayenne	65
4.7.4 ThingSpeak	68

4.7.5 Datacake	69
Kapitel 5 • LoRa-Transceiver	75
5.1 Semtech LoRa-Transceiver	75
5.2 LoRa-Transceiver Module	75
5.2.1 HopeRF RFM95/96/97/98(W)	76
5.2.2 Olimex LoRa868	76
5.2.3 Microchip RN2483.	77
5.2.4 Murata CMWX1ZZABZ	78
Kapitel 6 • Reduzierung des Strombedarfs	80
Kapitel 7 • LoRaWAN-Knoten	84
7.1 Fertiggeräte	84
7.1.1.Elsys ERS Lite - LoRaWAN Raumsensor	84
7.1.2 Dragino LHT65 Temperatur- & Feuchtigkeitssensor	92
7.1.3 Dragino LDS01 Tür- & Fenstersensor.	100
7.1.4 Dragino LDS02 Tür- & Fenstersensor.	106
7.1.5 Dragino GPS Tracker LGT-92	106
7.1.6 tabs Healthy Home Sensor IAQ	111
7.1.7 Big Clown LoRaWAN Monitor	120
7.1.8 CleverCity Greenbox	124
7.2 Boards	128
7.2.1 M5Stack ENV.II Unit	129
7.2.2 The Things Uno	135
7.2.3 Seeeduno LoRaWAN	143
7.2.4 Arduino MKR WAN 1300	149
7.2.5 Heltec WiFi LoRa 32	158
7.2.6 Heltec CubeCell	169
7.2.6.1 Heltec CubeCell DevBoards	170
7.2.6.2 Heltec CubeCell Capsule Sensors	176
7.2.7 M5Stack COM.LoRaWAN	184
7.2.8 M5Stack LoRaWAN868 Unit	191
7.2.9 PaperiNode	198
7.2.10 ATmega1284 TTN/LoRa Board V3.1	202

Kapitel 8 • Glossar	215
8.1 Allgemeine Begriffe	215
8.2 LoRaWAN-spezifische Begriffe	216
Kapitel 9 • Index	218
Kapitel 10 • Referenzen	223

Kapitel 1 • Vorwort

Das Internet der Dinge (Internet of Things, IoT) ist in aller Munde und verheißt massive Änderungen in unserem gesamten Umfeld. Die Auseinandersetzung mit jeder neuen Technologie ist insofern wichtig, dass man später nicht unvorbereitet mit diesen Dingen konfrontiert wird. Im Nachhinein lässt sich ein solches Gebiet möglicherweise nur schwierig erschließen.

Das IoT definiert sich über die Vernetzung von Dingen (Sensoren, Embedded Systems), der damit verbundenen Schaffung von Mehrwert und schließlich darauf aufbauend neuen Geschäftsmodellen. Das verbindende Element sind erfasste und transportierte Daten.

Mit den bislang zur Verfügung stehenden Technologien war das prinzipiell bereits möglich, allerdings nicht zu akzeptablen Kosten oder nur mit eingeschränkten Eigenschaften. Wenn beim oft beschworenen intelligenten Kühlschrank die IoT-Anbindung teurer als der Kühlschrank selbst wird, dann ist kaum Akzeptanz für ein solches Produkt zu erwarten.

Die Grundeigenschaften eines IoT-Knotens, wie wir fortan unseren intelligenten Kühlschrank und ähnliche zu vernetzende Komponenten, also die Dinge an sich, bezeichnen, sind

- große Reichweite,
- kabellose Verbindung zum Netzwerk,
- Batteriebetrieb mit langer Lebensdauer der Batterie und
- tiefer Preis.

Als IoT-Knoten werden vielfältige Objekte, Sensoren, Messgeräte, Überwachungsanlagen, Sicherheitssysteme u.v.a.m. bezeichnet.

Das Low Power Wide Area Network (LPWAN) steht hier als Oberbegriff für viele unterschiedliche Kommunikations-Protokolle. Es gibt weitere, wie Ihnen Abbildung 1 anhand der weltweiten LPWAN-Verbindungen zeigt. Neben dem hier noch zu betrachtenden LoRa bzw. LoRaWAN stehen NB-IoT, Sigfox, LTE-M, Weightless, Symphony Link und einige andere im Wettbewerb.

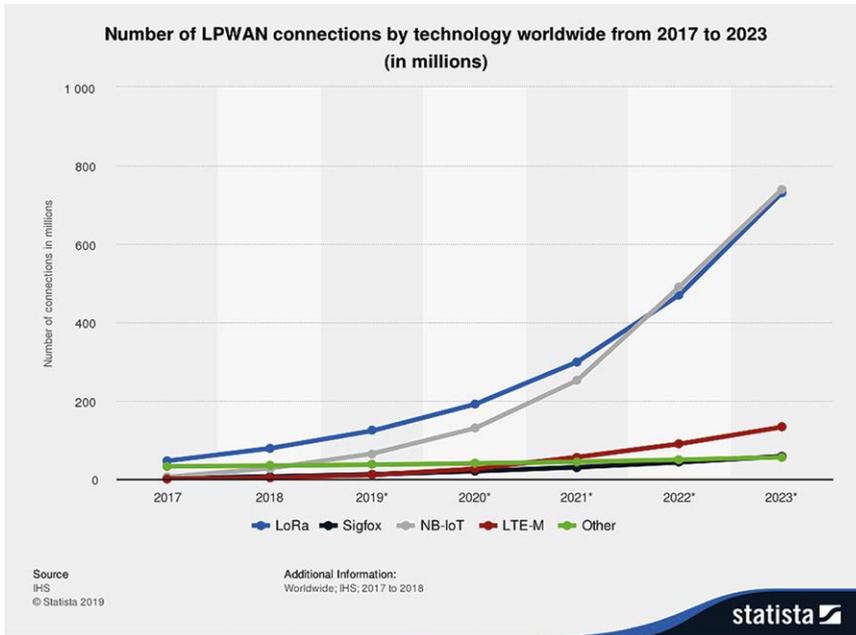


Abbildung 1: Anzahl von LPWAN-Verbindungen weltweit

In Abbildung 1 interessant ist die Dominanz von LoRaWAN und NB-IoT gegenüber den anderen LPWAN-Technologien.

LoRaWAN ist eine offene LPWAN-Systemarchitektur, die von der LoRa Alliance™, einer gemeinnützigen Vereinigung von mehr als 500 Mitgliedsunternehmen, entwickelt und standardisiert wurde. LoRa baut auf der *Chirp Spread Spectrum (CSS)* Modulationstechnik auf, die die Long-Range-Low-Power-Kommunikation ermöglicht. Darauf werde ich noch im Detail eingehen.

NB-IoT hingegen arbeitet im lizenzierten Spektrum (also nicht kostenfrei) und nutzt, wie *LTE*, *Frequency Division Multiple Access (FDMA)* im Uplink, *Orthogonal FDMA (OFDMA)* im Downlink und *Quadrature Phase Shift Keying (QPSK)*-Modulation.

Beide Technologien können bei Quality of Service (QoS) konkurrieren. IoT-Anwendungen, die mehr Kommunikation erfordern, werden durch NB-IoT besser bedient, da NB-IoT keine Einschränkungen des Arbeitszyklus im lizenzierten Spektrum hat. NB-IoT weist allerdings höhere *Total Cost of Ownership (TCO)* im Vergleich zu LoRaWAN auf.

Keine Angst vor den hier verwendeten Begriffen. Im Glossar am Ende des Buchs sind hierzu Erläuterungen zu finden. So viel zur Abgrenzung von LoRaWAN und NB-IoT.

Wollen Sie den Vergleich der beiden LPWAN-Technologien weiter vertiefen, dann möchte ich Sie auf das Whitepaper mit dem Titel „LoRaWAN and NB-IoT: Competitors or Complementary?“ verweisen [5].

Es bleibt also festzuhalten, dass LoRaWAN eine verbreitete und unter bestimmten Bedingungen auch für jeden zugängliche und kostenfreie Möglichkeit zur Übermittlung von Daten eines IoT-Knotens ist und deshalb hier im Mittelpunkt der Betrachtungen steht.

Im Gegensatz zu einigen anderen Protokollen ist der LoRa-Standard Open Source und nicht proprietär. Das ist einer der Gründe für das rasante Wachstum von LoRaWAN-Netzwerken über ganze Länder hinweg, beginnend in den Ballungszentren.

In diesem Buch werden kommerziell angebotene Sensorknoten sowie mit einfachen Mitteln und zu niedrigen Kosten entwickelte LoRaWAN-Sensorknoten eingesetzt. Diese LoRaWAN-Knoten senden ihre Daten an einen LoRaWAN-Server. Von diesem sind sie dann abrufbar und in eine beliebige Anwendung integrierbar.

Wenn Sie bislang mit einem Arduino erste Erfahrungen sammeln konnten, dann sind Sie bestens auf diese zukunftssträchtige Aufgabenstellung vorbereitet. Die Arduino IDE deckt alle hier betrachteten Mikrocontroller unterschiedlicher Architektur ab, so dass Sie hier nicht mit unterschiedlichen Entwicklungsumgebungen arbeiten müssen.

Auf der Basis eines ESP8266-Mikrocontrollers von Espressif hatte ich zu einem früheren Zeitpunkt bereits gezeigt, dass man einen WiFi-tauglichen IoT-Knoten zu sehr geringen Kosten (es waren 15 US\$) aufbauen kann [1].

Ich hatte damals mit einem direkt aus China bezogenen *ESP8266 Node MCU* gearbeitet. Mittlerweile gibt es auch hier genügend günstige Angebote, allerdings nicht zu vergleichbaren Preisen.

Dass WiFi auf Grund der geringen Reichweite und des doch recht hohen Strombedarfs für einen batteriebetriebenen IoT-Knoten allerdings nur bedingt geeignet ist, war auch durch eigene Untersuchungen gezeigt worden [2][3].

Mit den hier vorzustellenden LoRaWAN-Knoten eröffnen sich vollkommen neue Möglichkeiten, wobei die Kosten nicht wesentlich höher liegen.

Für das *Internet of Things* (IoT) ist eine weltweite Infrastruktur erforderlich, die die IoT-Knoten zum Informationsaustausch verbindet. Schon heute begegnen uns diese Entwicklungen täglich.

Denken Sie an Ihren Fitness-Tracker, der Sie an zu absolvierende Übungen erinnert oder Ihre Waage, die vor weiterer Gewichtszunahme warnt. Ihre Armbanduhr deckt gnadenlos ein mögliches Schlafdefizit auf und Ihr Mobilphone wird mit einer Corona Warn-App vielleicht einmal helfen, Covid-19-Infektionsketten zu unterbrechen.

Schon mit diesen wenigen, willkürlich ausgewählten Beispielen zeigt sich die Komplexität der Anwendungen, die über den technischen Ebenen von Verbindungsaufbau und Datenaustausch anzuordnen ist. Um die letzteren Themen soll es in diesem Beitrag zu LoRa & LoRaWAN gehen.

LoRaWAN hat sich als Kommunikationslösung im IoT hervorragend entwickelt. Für den Maker wie für den kommerziellen Einsatz hat *The Things Network* (TTN) viel dazu beigetragen.

TTN ist ein globales kollaboratives IoT-Ecosystem, das Netzwerke, Geräte und Lösungen mit LoRaWAN erstellt. Mit der *The Things Stack Community Edition* (TTS (CE)) betreibt TTN jetzt ein offenes und dezentrales LoRaWAN-Netzwerk. Dieses Netzwerk ist eine großartige Möglichkeit, um mit dem Testen von Geräten, Anwendungen und Integrationen zu beginnen und sich mit LoRaWAN vertraut zu machen.

Mit den hier vermittelten Erkenntnissen werden Sie in die Lage versetzt, Geräte und Gateways im TTS (CE) anzumelden oder Ihre bereits im TTN V2 befindlichen Geräte auf TTS (CE) zu migrieren.

Zur besseren Lesbarkeit beachte ich bei der textlichen Darstellung die folgenden Konventionen:

- Befehle und Ausgaben an die Konsole werden in *Courier New* dargestellt.
- Eingaben über die Konsole erfolgen in **Courier New fett**.
- Programm- und Dateinamen erscheinen *kursiv*.

Lange Webadressen (URLs) werden oft durch den URL Shortener Bitly (<https://bitly.com>) verkürzt.

Alle bestehenden Links wurden im Sommer 2021 überprüft. Da sich das Internet ständig verändert, kann nicht sichergestellt werden, dass diese Links funktionieren oder zum selben Inhalt führen wie zum Zeitpunkt der Aufnahme. Bitte informieren Sie mich über defekte Links.

Die hier vorgestellten Programmbeispiele sind unter folgender URL auf GitHub abgelegt und können von dort heruntergeladen werden:

<https://github.com/ckuehnel/LoRaWAN-Node/tree/master/---%20Elektor%20--->

Altendorf, im Juli 2021

Claus Kühnel

Kapitel 2 • LoRa

LoRa wurde speziell für die Anforderungen des IoT entwickelt und ist vor allem durch die folgenden Merkmale gekennzeichnet [4]:

- LoRa kann kleine Datenmengen per Funk mehrere Kilometer weit übertragen.
- Die große Übertragungsdistanz wird mit minimaler Sendeleistung erreicht.
- Durch den geringen Energiebedarf können Sensorknoten mit Batterien bestückt über mehrere Jahre autonom funktionieren.
- LoRa ermöglicht eine bi-direktionale Kommunikation mit Datenraten von 0,3 bis 50 kbps. Datenübertragungen sind in einer städtischen Umgebung über Entfernungen von 2 bis 5 km möglich. In einer Vorort-Umgebung sind Übertragungen bis zu 15 km realisierbar.
- Die Modulationsart von LoRa ist eine Ableitung der Chirp Spread Spectrum (CSS) Encoding-Technologie. Die Technologie nutzt die gesamte zugewiesene Bandbreite für das Senden eines Signals, um es über ein breiteres Band des Spektrums zu verteilen. Dies bedeutet, dass ein solches System auch bei einem niedrigen Signal-zu-Rausch-Verhältnis (SNR) betrieben werden kann. LoRa weist damit auch eine hohe Robustheit gegenüber Rauschen auf.
- LoRa nutzt in Europa die lizenzfreien ISM-Bänder bei 433 MHz und 868 MHz (ISM = Industrial Scientific & Medical)
- Dank integrierter Verschlüsselungstechnologie werden die Daten sicher übertragen und vor unerlaubtem Zugriff optimal geschützt.

Wie diese Eigenschaften erreicht werden und welche Anpassungsmöglichkeiten LoRa bietet, werde ich Ihnen in den folgenden Abschnitten zeigen.

2.1 LoRa-Grundlagen

LoRa ist eine proprietäre und patentierte Technologie der Semtech Corp., die aufbauend auf der Direct Sequence Spread Spectrum Modulation (DSSS) mit der Chirp Spread Spectrum (CSS) Modulation arbeitet.

Durch die Modulation des Datenstroms (Bitfolge) mit einer Codesequenz (pre-defined bit pattern) wesentlich höherer Bitrate wird ein Ausgangssignal mit stark vergrößerter Bandbreite erzeugt. Die Bits der Codesequenz werden zur Unterscheidung von den Bits des digitalen Datenstroms Chirps genannt. Abbildung 2 zeigt die DSSS-Modulation und die damit verbundene Spreizung (Spreading) des zu übertragenden Frequenzbandes.

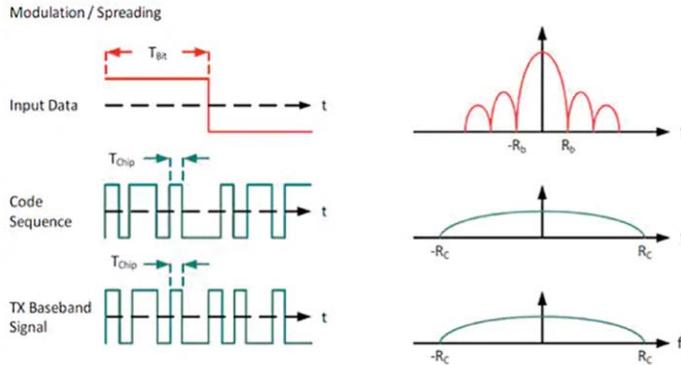


Abbildung 2: Direct Sequence Spread Spectrum Modulation (Bild: Semtech)

Aus Abbildung 2 wird ersichtlich, dass die Bandbreite durch die Chip Rate bestimmt wird. Je größer die Chip Rate (d.h. je kleiner die Chip Time T_{Chip}), desto größer wird die resultierende Bandbreite:

$$BW = R_c = \text{chip rate} \left(\frac{\text{chirps}}{s} \right)$$

Der Nachteil eines solchen DSSS-Systems ist, dass es einen hochgenauen Referenztakt benötigt.

Die LoRa Chirp Spread Spectrum (CSS)-Technologie von Semtech bietet eine kostengünstige und stromsparende, aber dennoch robuste DSSS-Alternative, die keinen hochgenauen Referenztakt erfordert. Bei der LoRa-Modulation wird die Spreizung des Signalspektrums durch die Erzeugung eines Chirp-Signals erreicht, das sich kontinuierlich in der Frequenz ändert. Abbildung 3 zeigt die verwendeten Chirp-Signale.

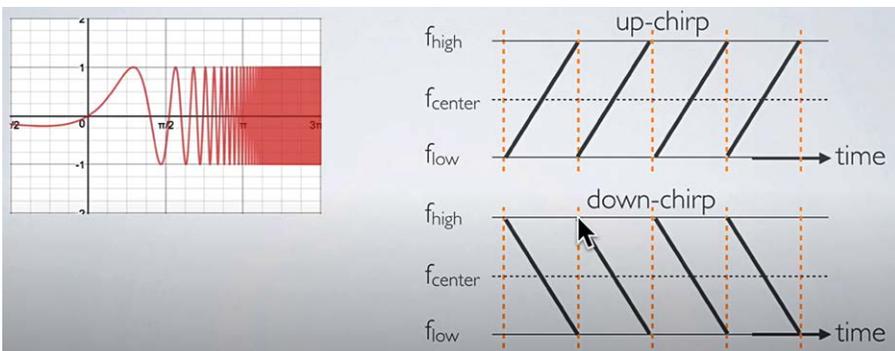


Abbildung 3: Chirp Signal for LoRa-Modulation (Bild: mobilefish.com)

Abbildung 4 zeigt ein unmoduliertes und ein moduliertes Signal im Vergleich.

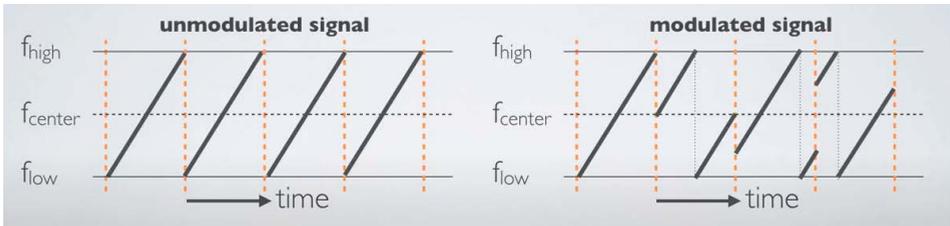


Abbildung 4: LoRa-Modulation (Bild: mobilefish.com)

Beim unmodulierten Signal geht die Frequenzänderung stetig von f_{low} nach f_{high} . Beim modulierten Signal hingegen kennzeichnet der Frequenzsprung das zu übertragende Symbol. Ein Symbol repräsentiert dabei ein oder mehrere Datenbits.

Der Spreading Factor (SF) gibt an, wie viele Chirps verwendet werden, um ein Symbol zu codieren. Für die Symbolrate R_s und die Bitrate R_b gelten damit folgende Beziehungen:

$$R_s = \frac{BW}{2^{SF}} \quad \text{und} \quad R_b = SF \frac{BW}{2^{FS}}$$

Eine komplette LoRa-Massage ist in Abbildung 5 dargestellt. Die LoRa-Massage beginnt mit einer Preamble (Einleitung) mit acht Up-Chirps, gefolgt von zwei Down-Chirps zur Synchronisation (durch Pfeil markiert). Danach schließen sich die modulierten Daten an.

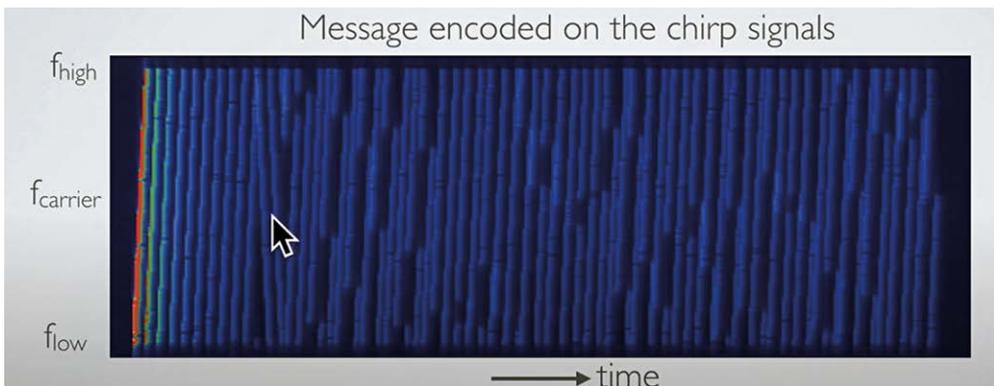


Abbildung 5: LoRa-Massage (Bild: mobilefish.com)

Das Format einer kompletten LoRa-Massage zeigt schließlich Abbildung 6.

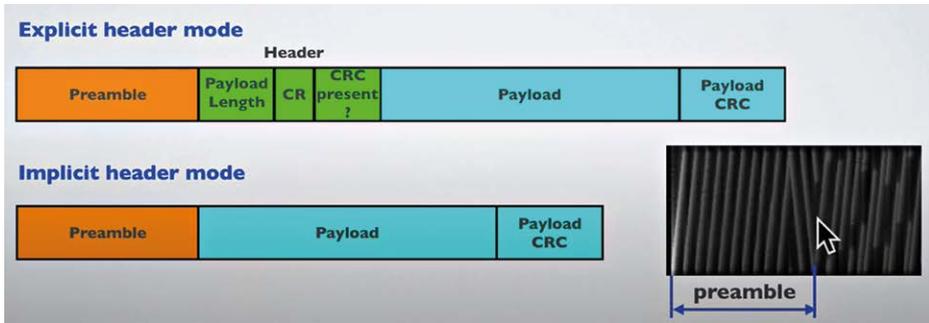


Abbildung 6: LoRa-Message-Format (Bild: mobilefish.com)

Interessant ist, wie lange die Übertragung einer solchen LoRa-Message dauert. Die sogenannten Time-on-Air (ToA) oder Air Time berechnet sich nach folgenden Beziehungen:

$$ToA = T_{Packet} = T_{Preamble} + T_{Payload}$$

$$T_{Preamble} = 12.25 T_s \text{ (for EU 868) mit } T_s = \frac{2^{FS}}{BW}$$

Die Berechnung der Payload-Time ist etwas komplexer und kann am einfachsten mit einem entsprechenden Tool vorgenommen werden. Unter der URL <https://www.loratools.nl/#/airtime> finden Sie einen *Air Time Calculator*.

Ich habe mit diesem Air Time Calculator die zu erwartende Air Time für SF7 (Abbildung 7) und SF12 (Abbildung 8) bei einer Payload von 10 Byte berechnet.

Da diese Berechnungen für das noch zu betrachtende LoRaWAN gelten, sind die entsprechenden Defaults auch schon berücksichtigt.

Calculate the air time of your LoRa frame.

Default values are for EU868 band.

LoRa Modem settings

Spreading factor	<input type="text" value="7"/>	7 - 12
Bandwidth	<input type="text" value="125"/> kHz	125 kHz default for LoRaWAN. 250 kHz also supported.
Code rate	<input type="text" value="1"/>	$4 / (CR + 4) = 4/5$. 4/5 default for LoRaWAN

Frame configuration

Payload length	<input type="text" value="10"/> bytes	
Preamble length	<input type="text" value="8"/> symbols	Default for frame = 8, beacon = 10.
Explicit header	<input checked="" type="checkbox"/> Yes	Default on for LoRaWAN
CRC	<input checked="" type="checkbox"/> Yes	Default on for LoRaWAN

Low data rate optimization	No	Enabled for bandwidth 125 kHz and Spreading factor >= 11
Preamble length	12.54 ms	
Symbol length	1.02 ms	
Symbols in frame	28	
Time on air	41.22 ms	
Duty cycle	One message every 00:04 (mm:ss)	

Abbildung 7: Air Time Berechnung für SF7 bei 10 Byte Payload

Calculate the air time of your LoRa frame.
 Default values are for EU868 band.

LoRa Modem settings

Spreading factor: 7 - 12

Bandwidth: kHz 125 kHz default for LoRaWAN. 250 kHz also supported.

Code rate: 4 / (CR + 4) = 4/5. 4/5 default for LoRaWAN

Frame configuration

Payload length: bytes

Preamble length: symbols Default for frame = 8, beacon = 10.

Explicit header: Yes Default on for LoRaWAN

CRC: Yes Default on for LoRaWAN

Low data rate optimization: Yes Enabled for bandwidth 125 kHz and Spreading factor >= 11

Preamble length: 401.41 ms

Symbol length: 32.77 ms

Symbols in frame: 18

Time on air: 991.23 ms

Duty cycle: One message every 01:39 (mm:ss)

Abbildung 8: Air Time Berechnung für SF12 bei 10 Byte Payload

Erwartungsgemäß unterscheiden sich die Zeiten deutlich. In Tabelle 1 habe ich die Ergebnisse vergleichend gegenübergestellt.

Time	SF7	SF12
Preamble Length	12.54 ms	401.41 ms
Symbol Time	1.02 ms	32.77 ms
Time on Air	41.22 ms	991.23 ms
Duty Cycle	0:04 min	1:39 min

Tabelle 1: Ergebnisse Air Time Calculator

Es bleibt festzuhalten, dass bei einem hohen Spreading Factor die Übertragung einer LoRa-Messung entsprechend lange dauert und damit auch einen erhöhten Energiebedarf aufweisen wird.

Durch die Duty Cycle Beschränkung im LoRaWAN können dadurch weniger oft LoRa-Messages verschickt werden. Hierauf werde ich später noch eingehen.

Eine Zusammenfassung der LoRa-Spezifikationen für Europa EU868 zeigt Tabelle 2.

Parameter	Wert
Frequenzband	867 bis 869 MHz
Kanäle	10
Kanalbandbreite Upstream	125/250 kHz
Kanalbandbreite Downstream	125 kHz
TX-Leistung Upstream	+14 dBm
TX-Leistung Downstream	+14 dBm
Spreading Factor Upstream	7 bis 12
Datenrate	250 Bit/s bis 50 Kbit/s
Link-Budget Upstream	155 dB
Link-Budget Downstream	155 dB

Tabelle 2: LoRa-Spezifikationen für Europa EU868

2.2 LoRa-Reichweite

Das Link-Budget gibt die Qualität eines Funk-Übertragungskanals an. Über ein einfaches Modell lässt sich das Link-Budget durch Addition der Sendeleistung (Transmitter Power, Tx), der Empfängerempfindlichkeit (Receiver Power, Rx), des Antennengewinns auf Sender- und Empfängerseite und der Freiraumdämpfung (Free Space Path Loss, FSPL) errechnen (Abbildung 9) [6].

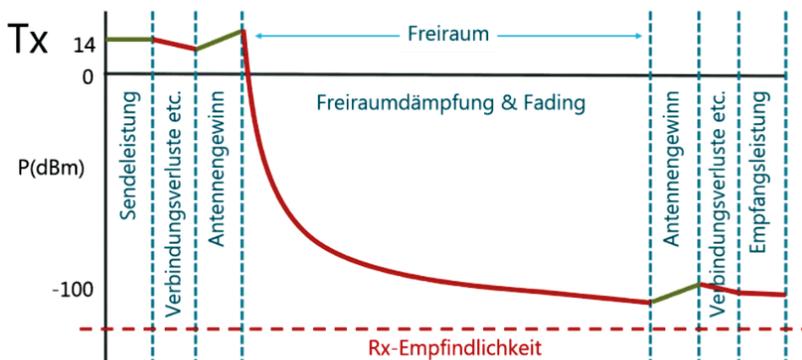


Abbildung 9: LoRa Link Budget (Bild: smartmakers.io)

Die Freiraumdämpfung (FSPL) besagt, wie viel Energie auf einer Entfernung zwischen Sender Tx und Empfänger Rx im freien Raum verloren geht. Je weiter die Entfernung zwischen Tx und Rx, desto niedriger ist die verbleibende Energie.

Mit der Freiraumdämpfung wird versucht, die Stärke eines HF-Signals in einer bestimmten Entfernung vorherzusagen. Dies ist ein theoretischer Wert, da es in der realen Welt viele Hindernisse, Reflexionen und Verluste gibt, die bei der Schätzung des Signals an einem Standort berücksichtigt werden müssen.

Der FSPL ist jedoch eine gute Näherung zum Schätzen des Signalverlusts bei der Ausbreitung durch den freien Raum. Es gilt die Beziehung:

$$FSPL = 20 \log_{10}(d) + 20 \log_{10}(f) + 20 \log_{10}\left(\frac{4\pi}{c}\right) - G_{Tx} - G_{Rx}$$

mit d = Entfernung zwischen Sende- und Empfangsantenne, f = Sendefrequenz, G_{Tx} = Antennengewinn der Sendeantenne und G_{Rx} = Antennengewinn der Empfangsantenne.

Mit dem FSPL Calculator (<https://www.everythingrf.com/rf-calculators/free-space-path-loss-calculator>) wird die Berechnung deutlich vereinfacht. So bekommt man beispielsweise bei einer Entfernung von 1 km zwischen Sende- und Empfangsantenne eine theoretische Freiraumdämpfung von ca. 91 dB (Abbildung 10).

Calculate the Signal Lost in Free Space

Distance (d)
 km

Frequency (f)
 MHz

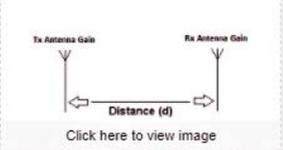
Transmitting Antenna Gain (G_{Tx})
 dB

Receiving Antenna Gain (G_{Rx})
 dB

Result

Free Space Path Loss

dB



Click here to view image

Abbildung 10: FSPL Calculator

Empfängerseitig (Rx) ist die Empfindlichkeit des Empfängers die Größe, die das Link-Budget beeinflusst. Die sog. Rx-Empfindlichkeit bezeichnet die minimal mögliche Empfangsleistung und Toleranz für thermisches Rauschen und wird folgendermaßen berechnet:

$$Sense_{Rx,max} = -174 + 10 \log_{10}(BW) + NF + SNR$$

mit BW = Bandbreite in Hz, NF = Rauschfaktor in dB und SNR = Signal-Rausch-Verhältnis, welches angibt, wie weit das Signal über dem Rauschen liegen muss.

Mit den Werten aus [6]

Tx-Leistung = 14 dBm

$BW = 125\text{KHz} = 10 \log_{10}(125000) = 51$

$NF = 6$ dB (die Gateways in LoRaWAN-Netzwerken haben niedrigere NF-Werte)

$SNR = -20$ (für $SF=12$)

erhalten wir eine Rx-Empfindlichkeit = $-174 + 51 + 6 - 20 = -137$ dBm.

Für das Linkbudget LB gilt

$$LB = Sense_{Rx,max} - Sense_{Tx,max}$$

und mit den genannten Werten dann ein Wert von -151 dB.

Zur Optimierung von Datenraten, Sendezeit und Energieverbrauch im Netzwerk dient der Adaptive Data Rate (ADR) Mechanismus, der die folgenden Übertragungsparameter eines Endgeräts steuert:

- Spreading Factor
- Bandbreite
- Übertragungsleistung

ADR kann den Stromverbrauch von Geräten optimieren und gleichzeitig sicherstellen, dass Nachrichten weiterhin von Gateways empfangen werden.

Wenn ADR verwendet wird, zeigt der Netzwerkservers dem Endgerät an, dass er die Sendeleistung reduzieren oder die Datenrate erhöhen soll. Endgeräte in der Nähe von Gateways sollten einen niedrigeren Spreading Factor und eine höhere Datenrate verwenden, während weiter entfernte Geräte einen hohen Spreading Factor verwenden sollten, da sie ein höheres Link-Budget benötigen.

ADR sollte immer dann aktiviert werden, wenn ein Endgerät ausreichend stabile HF-Bedingungen hat. Dies bedeutet, dass es generell für statische Geräte aktiviert werden kann.

Die Endgeräte entscheiden, ob ADR verwendet werden soll oder nicht, nicht die Anwendung oder das Netzwerk.

Die eingesetzten Antennen können noch einen zusätzlichen Antennengewinn als Beitrag zum Linkbudget bringen. In der Regel werden Stabantennen mit einer Abstrahlcharakteristik eingesetzt, die an einen Donut erinnert (Abbildung 11).

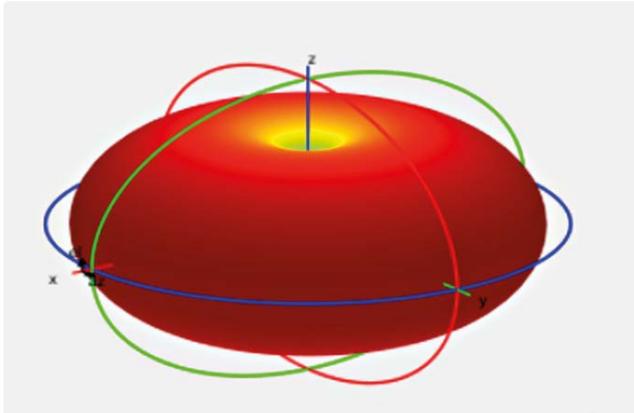


Abbildung 11: Antennencharakteristik Stabantenne

Die Sendeantenne strahlt den Großteil ihrer Leistung seitlich ab, nach oben und unten ist die Abstrahlung geringer. Bei der Empfindlichkeit der Empfangsantenne verhält es sich vergleichbar. Sende- und Empfangsantenne sollten deshalb zumindest in etwa in einer Ebene angeordnet werden.

Zusammenfassend kann folgendes festgehalten werden:

- Die maximale Reichweite einer LoRa-Übertragungsstrecke wird durch das Linkbudget bestimmt.
- Durch die Freiraumdämpfung wird die Reichweite begrenzt.
- Hindernisse, die Reflexionen und Brechungen hervorrufen, verkürzen die Reichweite zusätzlich.
- Antennengewinn hilft die Reichweite zu vergrößern.
- Bei der Ausrichtung der Antennen ist die Antennencharakteristik zu beachten, damit die Abdeckung gesichert ist.

Die hier vorgenommene Darstellung ist rein theoretischer Natur. Eine praktische Vertiefung kann in [7] nachgelesen werden. Dort wird gezeigt, wie sich das Übertragungsverhalten in realer Umgebung verändert und was ggf. von vornherein zu beachten ist.

2.3 LoRa-Kommunikation

Auf Basis von LoRa-Knoten kann ein einfaches Peer-to-Peer Netzwerk aufgebaut werden, in dem alle Knoten gleichberechtigt sind. Abbildung 12 zeigt ein solches Netzwerk, in dem jeder Knoten alle Mitteilungen anderer Knoten empfängt.

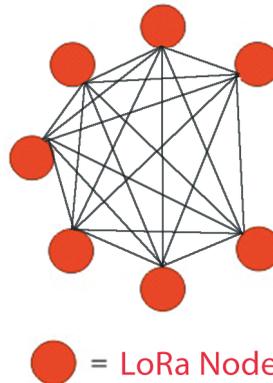


Abbildung 12: Peer-to-Peer Netzwerk

In meinem Arduino-Handbuch [8] hatte ich eine reine LoRa-Datenübertragung beschrieben, die den in Abbildung 13 gezeigten Aufbau hatte.

Für den Test bin ich von einem Sensorknoten ausgegangen, der mit einem DHT11-Sensor Temperatur und Feuchtigkeit erfasst und diese Werte exemplarisch über LoRa an zwei Empfänger sendet. Eine Adressierung der Empfänger ist nicht vorgesehen (Broadcasting).

Grundsätzlich können Sie für die Empfängerknoten die gleiche Hardware wie für den Sensorknoten einsetzen. Ich habe aus praktischen Gründen Wemos-TTGO-Devices eingesetzt.

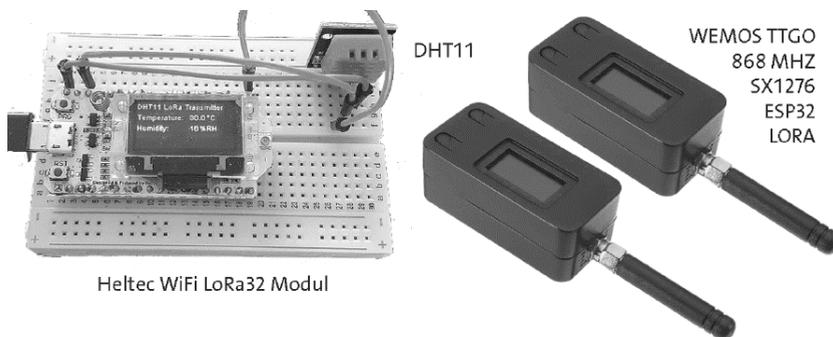


Abbildung 13: LoRa-Funkstrecke

Neben der Einbindung des ESP32 in die Arduino IDE müssen Sie noch die LoRa-Node- und die esp8266-oled-ssd1306-Library einbinden.

Zum Test der LoRa-Verbindung können Sie die Programme *Heltec_LoRa32_Transmit.ino* und *Heltec_LoRa32_Receive.ino* aus dem Repository verwenden. Wollen Sie auf Sensor und OLED verzichten, dann können Sie auch die Programme *LoRaSender.ino* und *LoRaReceiver.ino* verwenden.

Kapitel 3 • LoRaWAN

LoRaWAN bezeichnet eine Low Power Wide Area Network (LPWAN) Spezifikation und gleichzeitig die verwendete Netzwerk-Architektur.

3.1 LoRaWAN-Grundlagen

Abbildung 14 zeigt die unteren Schichten für LoRaWAN gemäß OSI-Modell. LoRa stellt die physikalische (PHY) Schicht dar. Wie im vorigen Abschnitt beschrieben, erfolgt hier die Chirp Spread Spectrum (CSS) Modulation, um eine Kommunikationsverbindung mit großer Reichweite herzustellen.

LoRaWAN bildet die Media Access Schicht (MAC) und ist ein offenes Netzwerkprotokoll, das sichere bidirektionale Kommunikations-, Mobilitäts- und Lokalisierungsdienste bereitstellt.

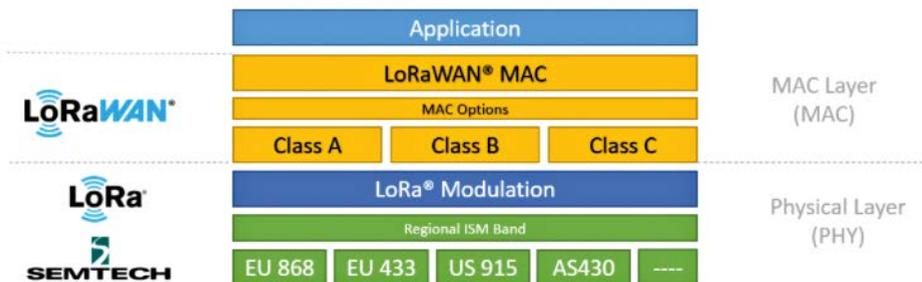


Abbildung 14: LoRaWAN Stack (Bild: Semtech)

LoRaWAN wird von der LoRa Alliance standardisiert und verwaltet. Die erste LoRaWAN-Spezifikation wurde im Januar 2015 veröffentlicht.

Die LoRaWAN Netzwerk-Architektur ist als erweiterte Star-Topologie ausgebildet. Gateways bilden eine Brücke zur Weiterleitung von Mitteilungen von End-Devices (LoRaWAN Nodes) zu einem zentralen Netzwerk-Server (LoRaWAN Server). Gateways sind mit dem Netzwerk-Server über Standard-IP verbunden, während End-Devices drahtlos mit einem oder mehreren Gateways kommunizieren (Abbildung 15).