



SAP und IoT - Die Anbindung von Bluetooth Low Energy (BLE) an SAP

Thomas Randl
Fakultät für Informatik

WS 2020/21

ToDo

Inhaltsverzeichnis

1	Einleitung	4
2	Vorgestellte Technologien	4
2.1	IoT	4
2.2	Industrie 4.0	4
2.3	SAP	4
3	Allgemeine Funktionsweise Bluetooth Low Energy	4
3.1	Protokollstack	4
3.2	Kommunikation	7
3.3	Anwendungsszenarien	8
4	Schnittstellenbeschreibung	9
4.1	SAP	9
4.2	BLE	9
4.3	Anbindungsmöglichkeiten	9
4.3.1	Bewertung der Möglichkeiten	9
4.3.2	Administration	9
5	Vergleich mit anderen gängigen IoT Kommunikationsprotokollen	9
5.1	Vorteile	9
5.2	Nachteile	9
6	Fazit	9

Abkürzungsverzeichnis

BLE	Bluetooth Low Energy
GAP	Generic Access Profile
GATT	Generic Attribute Profile
HCI	Host Controller Interface
IOT	Internet of Things
SIG	Special Interest Group
UUID	Universally Unique Identifier

1 Einleitung

2 Vorgestellte Technologien

Im folgenden Kapitel wird einleitend eine kurze Erläuterung über die drei zentralen Begriffe dieser Arbeit gegeben. Besonderer Schwerpunkt ist dabei die Aufschlüsselung der Begriffe und die Erklärung warum Diese in dieser Arbeit einen derartigen Stellenwert besitzen.

2.1 IoT

2.2 Industrie 4.0

2.3 SAP

3 Allgemeine Funktionsweise Bluetooth Low Energy

Im nachfolgenden Kapitel wird nun ein kurzer Überblick über die Technologie Bluetooth Low Energy (BLE) gegeben. Dabei wird zum einen die Architektur unter Erläuterung des Protokollstacks und zum anderen die Funktionsweise erläutert.

3.1 Protokollstack

Die Architektur von BLE geht aus dem Protokollstack hervor. Dieser ist in Abbildung 1 dargestellt. Besonders auffällig ist die Untergliederung in drei Ebenen. Der „Controller“

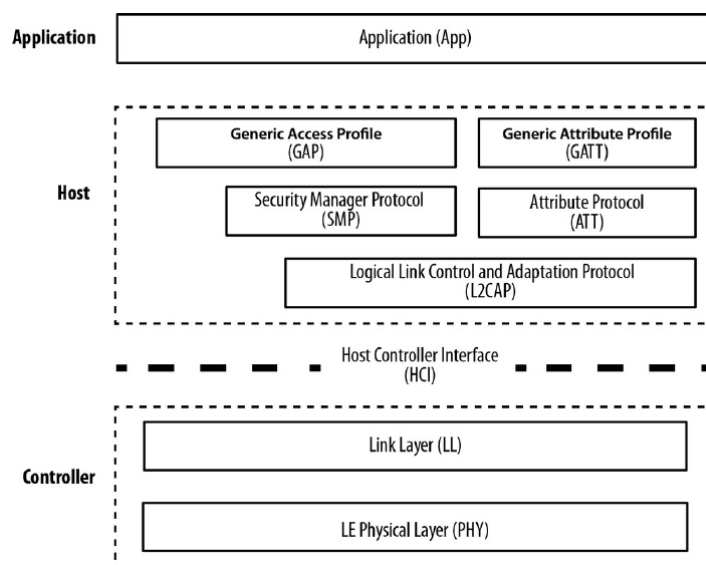


Abbildung 1: BLE Protokollstack [Tow14, Seite 16]

stellt dabei den hardwarenächsten Bereich dar. Hier befinden sich die zwei Layer „LE Physical“ und „Link“.

Diese beiden Protokolle sind in einer Großzahl von Gerätearchitekturen beheimatet. Das Physical Layer ist dafür vorgesehen, digitale Signale (Bitfolgen) in analoge umzuwandeln. Dieser Schritt wird benötigt, um die BLE Signale für etwaige Empfänger zugänglich zu machen. Natürlich werden im Physical Layer auch empfangene analoge Signale in digitale umgewandelt. Dieser werden dann im Protokollstack nach oben ins Link Layer weitergereicht. Im Fall von BLE ist die Schnittstelle für das Senden der analogen Signale die Luft. BLE sendet daher im Frequenzbereich 2,4GHz bis 2,4835GHz. Diesen Bereich teilt sich das Protokoll mit anderen wie beispielsweise Wifi. Um Kollisionen zu vermeiden teilt BLE den Bereich in 40 Kanäle auf und wechselt während der Verbindung in regelmäßigen Abständen oder bei Übertragungsproblemen den Kanal. Dieser Ansatz nennt sich Frequency Hopping Spread Spectrum [Tow14, Seite 16f].

Das Link Layer unterscheidet sich in seiner Funktionsweise nicht großartig von dem anderer Kommunikationsprotokolle. Hier werden die Nachrichten die für das Versenden aus den oberen Schichten ankommen in Pakete gepackt und an das Physical Layer weitergereicht. Dieser Prozess ist für ankommende Pakete natürlich vice versa [Tan14, Seit 194]. Besonders wichtig ist die Festlegung der Paketgröße von BLE durch das Link Layer. Seit Version 4.2 ist eine Payload von bis zu 251 Byte pro Paket möglich [Gup]. Zum Vergleich bei WLAN (IEEE 802.11) kann ein Paket bis zu 2312 Byte groß sein [Ges15, Seite 233]. Daraus lässt sich schließen, dass BLE einen weitaus geringeren Datendurchsatz als WLAN hat. Allerdings liefert BLE wiederum andere Vorteile. Auf diese wird in Kapitel 5 näher eingegangen.

Das Herzstück des BLE Protokollstacks bilden die beiden Profile Generic Access Profile (GAP) und Generic Attribute Profile (GATT). Diese befinden sich wie in Abbildung 1 zu erkennen im Host Bereich der Architektur. Die beiden Protokolle bilden die Schnittstelle zur tatsächlichen Anwendung mit welcher der Nutzer interagiert.

Das GAP ist dafür vorgesehen sämtliche Parameter der Verbindung zwischen den Geräten zu verwalten. Vom Verbindungsaufbau bis hin zur Kommunikation werden sämtliche Funktionen von diesem Profil bereitgestellt und abgehandelt.

Im Zuge der jeweiligen Konfiguration kann ein Gerät in BLE eine der folgenden vier Rollen annehmen:

- Broadcaster (Keine Verbindung)
- Observer (Keine Verbindung)
- Central (Verbindung)
- Peripheral (Verbindung)

In BLE ist es nicht festgeschrieben, dass Geräte eine Verbindung eingehen müssen um Informationen zu erhalten. Die Rollen des Broadcasters und Observers sind sogar ausschließlich ohne feste Verbindung zwischen den Geräten vorgesehen. Diese Funktion wird im allgemeinen gerne von BLE Beacons verwendet.

Ein Gerät welches als Broadcaster definiert ist sendet dauerhaft einen bestimmten Datensatz. Dabei ist zu keinem Zeitpunkt klar, ob Geräte in Reichweite sind, welche den Datensatz empfangen. Ein Gerät welches diese Daten lesen kann muss als Observer konfiguriert sein. Ein solcher scannt die drei Advertisement Kanäle von BLE dauerhaft nach Broadcastnachrichten. Falls er eine erhält liest er diese und verwendet sie. Wichtig ist hierbei, dass der Observer keine Antwort auf eine Nachricht sendet.

Sollte ein Gerät allerdings eine Verbindung eingehen, dann muss dieses als Central konfiguriert sein. Dies ist die gängigste Form des BLE Gerätes. So ist beispielsweise jedes Smartphone in der Regel als Central konfiguriert und kann Verbindungen zu Peripherals wie zum Beispiel BLE Kopfhörern aufnehmen. Dabei ist ein Central in der Regel sogar in der Lage mehrere Verbindungen zu selben Zeit einzugehen.

Ein Peripheral wiederum ist das Gegenstück zum Central, welches seine Verbindungsbereitschaft an sämtliche Geräte in Reichweite signalisiert. Im Fall einer aktiven Verbindung übernimmt das Central die Steuerung des Gerätes unter Berücksichtigung des Funktionsumfangs des Peripherals [Usa17, Seite 34].

Das GAP ermöglicht es einem BLE Gerät zusätzlich seine Sichtbarkeit und Verbindungsbereitschaft gegenüber anderen Geräten über die Advertisement Kanäle mitzuteilen. Dafür wird auf dem Gerät ein Modus eingestellt welcher anschließend an alle Geräte in Reichweite mitgeteilt wird. Ein Modus ist dabei eng mit der Geräterolle verbunden. Ein Gerät kann folgende Modi annehmen [Tow14, Seite 35]:

- Broadcast (Rolle: Broadcaster)
- Nicht zu entdecken (Rolle: Peripheral)
- Eingeschränkt zu entdecken (Rolle: Peripheral)
- Normal zu entdecken (Rolle: Peripheral)
- Nicht verbindbar (Rolle: Alle)
- Verbindbar (Rolle: Central, Peripheral)

Im GATT wird definiert, ob es sich bei dem Gerät um einen Client oder Server handelt. Zweiterer verarbeitet die Kommunikationsanfragen des Clients und liefert die gewünschten Antworten oder führt entsprechende Aktionen aus. Der Server ist in der Regel ein Peripheral auf dem Services hinterlegt sind [Usa17, Seite 30]. Welche das sind und was für eine Aktion mit diesen verbunden ist wird in der Regel durch die Nutzeranwendung festgelegt. Der Client ist dementsprechend ein Central, welches die gesamte Verbindung

steuert. Jeder Service verfügt über einen Universally Unique Identifier (UUID). Mit diesem kann der Client eine gezielte Anfrage auf den entsprechenden Service tätigen.

An oberster Stelle des Protokollstacks befindet sich die Nutzeranwendung. Diese ist nach dem entsprechenden Use Case programmiert und variiert von Anwendung zu Anwendung. Ausschließlich der Stack unterhalb ist für alle Applikationen gleich.

3.2 Kommunikation

Nachdem in Kapitel 3.1 der Aufbau von BLE erläutert wurde, wird nun in dem folgenden Kapitel auf die Anwendung der Technologie eingegangen. Dabei wird besonders auf den Verbindungsaufbau durch das Advertisement und den Nachrichtenaustausch der stehenden Verbindung eingegangen.

Die Advertisement Funktion in BLE kann für zwei Szenarien verwendet werden. Zum einen die Signalisierung der Verbindungsbereitschaft. Zum anderen den Broadcast von Daten in der Rolle des Broadcasters (vgl. 3.1).

Von den 40 Kanälen, in die der BLE Frequenzbereich unterteilt ist (siehe Kapitel 3.1), sind drei für das Advertisement reserviert. Diese befinden sich in den Bereichen 2,402 - 2,404GHz, 2,426 - 2,428GHz und 2,48 - 2,482GHz. Dabei sind sie mit den Nummern 37 - 39 belegt [Tow14, Seite 16].

Eine Advertisement Nachricht beinhaltet weiterhin ein Header Feld, welches die Paketadressierung regelt. So kann ein Gerät eine der folgenden vier Advertisement Nachrichten an Geräte in Sendereichweite senden [Tow14, Seite 22]:

- ADV_IND: Allgemeine Mitteilung der Verbindungsbereitschaft
- ADV_DIRECT_IND: Zielgerichtete Mitteilung der Verbindungsbereitschaft
- ADV_SCAN_IND: Allgemeine Mitteilung der Anwesenheit des Gerätes
- ADV_NONCONN_IND: Allgemeine Mitteilung der Verbindungsunverfügbarkeit

Eine Verbindung in BLE besteht aus einer Reihe sogenannter Verbindungsevents. Ein solches wird über drei Parameter definiert welche in den anfänglichen Advertisement Nachrichten vereinbart werden. In Abbildung 2 ist zu erkennen, wie eine BLE Verbindung, in zeitlicher Abfolge, abläuft.

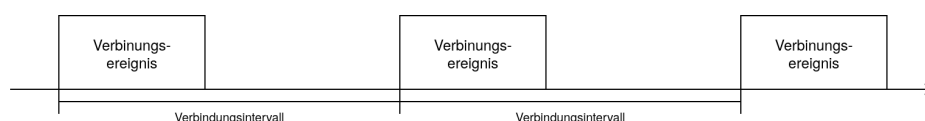


Abbildung 2: Zeitlicher Ablauf einer BLE Verbindung [Tow14, Seite 22]

Der erste Parameter ist das Verbindungsintervall. Wie aus Abbildung 2 hervorgeht ist eine Verbindung in derartige Verbindungsintervalle unterteilt. Diese legen die Zeit zwischen den Verbindungsereignissen fest. Dabei sind Werte zwischen 7,5ms und 4s möglich. Kürzere Intervalle bedeuten allerdings einen höheren Energieverbrauch. Ein Verbindungsereignis liegt im Intervall und kann somit maximal so lange wie dieses dauern. Die anderen beiden Parameter legen fest, wann eine Verbindung abbricht, beziehungsweise beendet wird.

BLE ist in seiner Rolle als energiesparsames Internet of Things (IOT) Protokoll darum bemüht die als Peripheral eingesetzten Server möglichst Energieeffizient zu betreiben. Daher kann ein derartiges Gerät mehrere Verbindungsevents aussetzen, wenn derzeitig keine Daten zu senden oder empfangen sind. Der Parameter „Slave Latency“ legt dabei die Anzahl der überspringbaren Ereignisse fest.

Den Zeitpunkt ab welchem eine Verbindung für Beendet erklärt wird, legt der Parameter „Connection Supervision Timeout“ fest. Dieser legt den Zeitraum fest, der zwischen dem Austausch zweier Datenpakete verstreichen darf. Bei Überschreitung dieses Wertes wird die Verbindung aufgelöst [Tow14, Seite 22f].

3.3 Anwendungsszenarien

Die Bluetooth Special Interest Group (SIG) stellt auf ihrer Website eine Vielzahl verschiedener Anwendungsmöglichkeiten für das Protokoll BLE vor. Dabei werden vier Hauptkategorien genannt, auf welche sich BLE konzentriert. Die im Alltag gebräuchlichsten sind dabei das Streamen von Audio und die Datenübertragung. Zusätzlich gibt es allerdings auch noch Positionsservices und das Aufspannen eines Netzwerkes zwischen mehreren BLE Geräten [BLU].

Seit einigen Jahren steigt die Anzahl der Endnutzer, die statt des herkömmlichen Kabelgebundenen Kopfhörer auf die kabellose Alternative wechseln. Mit der Veröffentlichung des Bluetooth Standards 5.0 wurde dieser Use Case mit BLE enorm gefördert. Neben Kabellosen Kopfhörern bilden auch Lautsprecher und Mediensysteme in Fahrzeugen interessante Use Cases seitens Bluetooth.

Die Möglichkeit der Datenübertragung ist die Basis des gesamten BLE Stack. Neben Audio Dateien gibt es eine Vielzahl anderer Anwendungsfälle, wie beispielsweise Kabellosen Tastaturen oder Mäuse. Auch neuere Technologien wie Fitnesstracker oder Smart Watches erfreuen sich in den letzten Jahren immer größerer Beliebtheit. So sind die Umsatzzahlen im Verkauf von Smart Watches laut Heise Online, im Vergleich zum Jahr 2019, dieses Jahr um 20 Prozent gestiegen [HEI]. BLE wird in diesem Fall verwendet um die von den jeweiligen Peripherals erhobenen Fitness oder Gesundheitsdaten an das Smartphone als Central Gerät zu übertragen.

Ein großes Problem der vergangenen Jahre war die Positionsbestimmung in geschlossenen Räumen. Im Freien ist diese durch Satellitenkommunikation möglich. In Gebäuden kann

diese Technologie allerdings auf Grund zu großer Ungenauigkeiten und der fehlenden direkten Verbindung nicht gewährleistet werden. Für dieses Problem liefert BLE mit Hilfe der Broadcaster Rolle eine Lösung. Dafür gibt es ein breites Spektrum an Möglichkeiten. Dieses Reicht von Indoor Navigation über die Positionsüberwachung von Gegenständen oder Personen bis hin zu Informationssystemen. Letztere stellen BLE Broadcaster dar, die bei Annäherung eines Observers nützliche Informationen senden. Dies kann beispielsweise in einem Museum sehr nützlich sein.

4 Schnittstellenbeschreibung

4.1 SAP

4.2 BLE

4.3 Anbindungsmöglichkeiten

4.3.1 Bewertung der Möglichkeiten

4.3.2 Administration

5 Vergleich mit anderen gängigen IoT Kommunikationsprotokollen

5.1 Vorteile

5.2 Nachteile

6 Fazit

Literatur

- [BLU] Learn About Bluetooth Solutions. <https://www.bluetooth.com/learn-about-bluetooth/solutions/>. Last visit: 26 Okt 2020.
- [Ges15] R. Gessler und T. Krause. Wireless-Netzwerke fuer den Nahbereich. In *Eingebettete Funksysteme: Vergleich von standardisierten und proprietären Verfahren*, Bd. 2, S. 233. Springer Vieweg, Kuenzelsau (Deutschland), Widdern (Deutschland), 2015.
- [Gup] S. Gupta und R. Kumar. BLE v4.2: Creating Faster, More Secure, Power-Efficient Designs - Part 1. <https://www.electronicdesign.com/communications/ble-v42-creating-faster-more-secure-power-efficient-designs-part-1>. Last visit: 15 Okt 2020.
- [HEI] Smartwatches: Apple dominiert, Huawei und Garmin stark . <https://www.heise.de/news/Smartwatches-Apple-dominiert-Huawei-und-Garmin-stark-4876304.html>. Last visit: 26 Okt 2020.
- [Tan14] A. Tanenbaum und D. Wetherall. In *Computer Networks*, Bd. 5, S. 194. Pearson, Harlow (Vereinigtes Koenigreich), 2014.
- [Tow14] K. Townsend, C. Cufi, Akiba und R. Davidson. Tools and techniques for lowpower networking. In *Getting Started with Bluetooth Low Energy*, S. 16 – 18, 22– 23, 35. O'Reilly Media Inc., Sebastopol (Vereinigte Staaten von Amerika), 2014.
- [Usa17] M. Usama und B. Aftab. Take your first steps in IoT. In *Building Bluetooth Low Energy Systems*, S. 30, 34. Packt, Birmingham (Vereinigtes Koenigreich), Mumbai (Indien), 2017.