



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

Thomas Randl  
Fakultät für Informatik

WS 2020/21

ToDo

# Inhaltsverzeichnis

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

## **Abkürzungsverzeichnis**

<b>BLE</b>	Bluetooth Low Energy
<b>GAP</b>	Generic Access Profile
<b>GATT</b>	Generic Attribute Profile
<b>HCI</b>	Host Controller Interface

# 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 anhand des Protokollstacks und zum anderen die Funktionsweise erläutert.

### 3.1 Protokollstack

Die Architektur von BLE lässt sich anhand des Protokollstacks erkennen. in Abbildung 1 ist dieser dargestellt. Besonders auffällig ist, dass er in drei Ebenen unterteilt ist. Der „Controller“ ist dabei der Bereich, welcher am nächsten an der Hardware sitzt. Hier befinden sich die zwei Lagen „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 beginnend bei 2,4GHz bis 2,4835GHz [Ran20, Seite 7]. Diesen Bereich teilt sich das Protokoll mit anderen wie zum Beispiel 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 17].

Das Link Layer unterscheidet sich in seiner Funktionsweise nicht großartig von dem anderer Kommunikationsprotokolle. Hier werden die Nachrichten die für das Versenden

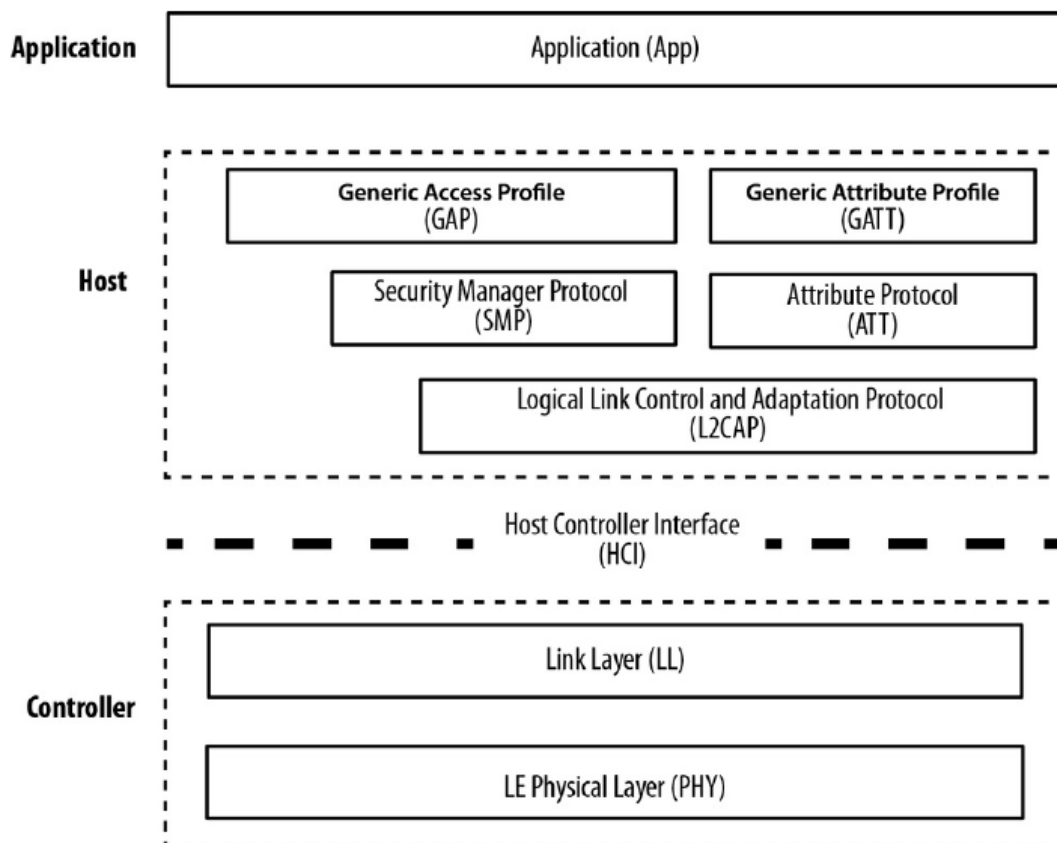


Abbildung 1: BLE Protokollstack [Tow14, Seite 16]

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].

Im Link Layer von BLE wird die Paketgröße festgelegt. Diese lässt seit Version 4.2 eine Payload von 251 Byte pro Paket zu [Gup]. Zum Vergleich bei WLAN (IEEE 802.11) kann ein Paket bis zu 2312 Byte groß sein [MTU]. 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 letzten Endes 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. [Ran20, Seite 15]

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 [Ran20, Seite 16]:

- 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)

## 3.2 Kommunikation

Todo: Rollen Verbindung und Kommunikation

Nachdem in Kapitel 3.1 der allgemeine Aufbau von BLE erläutert wurde, wird im nachfolgenden Kapitel auf das Kernelement, die tatsächliche Kommunikation zwischen zwei oder mehreren Geräten von BLE eingegangen. Dabei werden drei Elemente im besonderen betrachtet: die Bekanntmachung, die Verbindung und der Datenaustausch.

### 3.2.1 Advertisement

Das Advertisement, oder zu deutsch die Bekanntmachung, ist die Funktion, mittels welcher sich Geräte, die bereit sind, sich zu koppeln bei suchenden Geräten bekannt machen. Allerdings kann die Bekanntmachung auch für einen Broadcast von Daten ohne explizites Zielgerät verwendet werden.

Für das Advertisement sind drei der 40 Kanäle, in die der Frequenzbereich auf dem **ISM! (ISM!)** Band unterteilt ist, reserviert. Ein Gerät kann diese Kanäle nutzen und ein Paket senden, welches verbindungspezifische Daten bereitstellt. Generell gibt es folgende vier unterschiedliche Pakete, welche somit gesendet werden können:

- ADV\_IND
- ADV\_DIRECT\_IND
- ADV\_SCAN\_IND
- ADV\_NONCONN\_IND

Das „ADV\_IND“ Paket sendet eine Bekanntmachung an alle Geräte, die zuhören und gibt bekannt, dass das Gerät bereit ist, sich mit jeglichem Gerät zu verbinden. Im Gegensatz dazu sendet das „ADV\_DIRECT\_IND“ Paket eine direkte Nachricht an ein bestimmtes Gerät, dass es bereit ist, sich mit genau diesem zu koppeln. Um das zu gewährleisten, enthält die Payload des Paketes die beiden BLE Adressen der betroffenen Geräte. Diese beiden Pakete lassen auch zu, dass sich die Geräte miteinander verbinden. Die restlichen zwei Pakete lassen dies wiederum nicht zu. So ermöglicht das „ADV\_SCAN\_IND“ Paket, lediglich einen Broadcast an alle hörenden Geräte zu senden, dass dieses Gerät in Reichweite ist. Es ist also sichtbar für die anderen Geräte. Um jedoch eine Verbindung einzugehen, müssen weitere Schritte unternommen werden. Das letzte Paket teilt allen Geräten in Reichweite mit, dass dieses Gerät nicht für eine Kopplung zu Verfügung steht. Alle Pakete bis auf das „ADV\_DIRECT\_IND“ Paket können zusätzlich Daten enthalten, die über das Advertisement hinausgehen. Das ermöglicht auch den Einsatz von BLE Beacons, welche, ohne eine Verbindung einzugehen, Daten an alle Geräte in Reichweite senden können [ADV].

### 3.2.2 Verbindung

Nachdem in Kapitel ?? erklärt wurde, wie Pakete aufgebaut sind und übertragen werden, gilt es noch zu erläutern, wie eine Verbindung in BLE abläuft, um diese Pakete tatsächlich transferieren zu können. Nachdem ein Central ein Gerät über das Advertisement gefunden hat, kann es eine Verbindungsanfrage initiieren. In diesem Paket sind drei wichtige Verbindungsparameter angegeben:

- Verbindungsintervall
- Slave Latenz
- Überwachungszeitüberschreitung

Das Verbindungsintervall legt die Zeit fest, die zwischen zwei Verbindungsevents verstreicht. Eine Verbindung in BLE besteht ausschließlich aus derartigen Ereignissen und bleibt solange bestehen, wie diese regelmäßig stattfinden. Bei diesem Intervall gilt es abzuwägen, was wichtiger ist. Je geringer dieses Intervall ist, desto schneller ist die Verbindung. Jedoch verbraucht dieses Verhalten auch mehr Energie. Das Central muss also im Vorhinein festlegen, was für die kommende Verbindung wichtiger ist [CON]. Die Slave Latenz legt die Nummer der Verbindungsevents an, die der Slave erlaubt ist, zu überspringen, bevor die Verbindung als beendet gilt. Ein weiterer Weg, wie eine Verbindung vorzeitig abgebrochen werden kann, wird durch die Überwachungszeitüberschreitung festgelegt. Bei dieser wird definiert, wie viel Zeit zwischen zwei erfolgreichen Übertragungen verstreichen darf. Sollte es also zu dem Fall kommen, dass über einen längeren Zeitraum erfolglos Pakete versendet werden, wird die Verbindung beendet [Tow14, Seite 23].

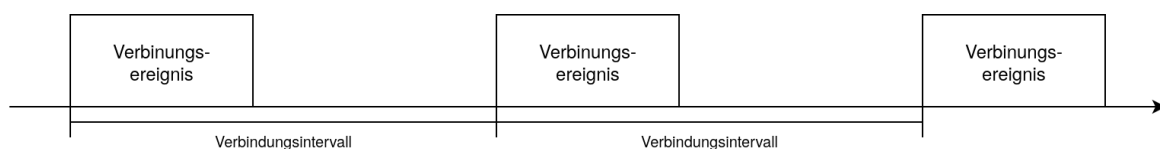


Abbildung 2: Ablauf einer BLE Verbindung [CON]

Wenn das Peripheral die Verbindung annimmt, wird diese, wie in Abbildung 2 dargestellt, aufgenommen. Jedes dieser Ereignisse folgt einem bestimmten Ablauf: zuerst überträgt der Master eine Anfrage an den Slave. Dieser empfängt die Anfrage und verarbeitet diese. Anschließend sendet er eine entsprechende Antwort, die der Master empfängt. Wichtig dabei ist, dass der Master nicht nur die Verbindung initiiert, sondern auch sämtliche Anfragen. Nachdem das definierte Verbindungsintervall abgelaufen ist, wird das nächste Verbindungsevent gestartet [CON].



### **3.3 Anwendungsszenarien**

## **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

- [ADV] Bluetooth Low Energy Scanning and Advertising. [http://dev.ti.com/tirex/content/simplelink\\_academy\\_cc2640r2sdk\\_1\\_12\\_01\\_16/modules/ble\\_scan\\_adv\\_basic/ble\\_scan\\_adv\\_basic.html](http://dev.ti.com/tirex/content/simplelink_academy_cc2640r2sdk_1_12_01_16/modules/ble_scan_adv_basic/ble_scan_adv_basic.html). Last visit: 19 Dez 2019.
- [CON] Bluetooth Low Energy Connections. [http://dev.ti.com/tirex/content/simplelink\\_academy\\_cc2640r2sdk\\_2\\_20\\_03\\_05/modules/blestack/ble\\_connections/ble\\_connections.html](http://dev.ti.com/tirex/content/simplelink_academy_cc2640r2sdk_2_20_03_05/modules/blestack/ble_connections/ble_connections.html). Last visit: 19 Dez 2019.
- [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.
- [MTU] MTU - Maximum Transfer Unit. <https://www.elektronik-kompodium.de/sites/net/0812211.htm>. Last visit: 15 Okt 2020.
- [Ran20] T. Randl. Funktionsweise und Einordnung in den Bereich der IOT Kommunikationsprotokolle. In *Bluetooth Low Energy*, S. 7, 15 – 16. 2020.
- [Tan14] A. Tanenbaum und D. Wetherall. In *Computer Networks*, Bd. 5, S. 95 – 104, 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. O'Reilly Media Inc., Sebastopol (Vereinigte Staaten von Amerika), 2014.