



ist es richtig dass (alle ?)Raspberries eine shared Antenne für BLE und WIFI haben?

Ja – bei allen aktuellen **Raspberry Pi-Modellen mit integriertem WLAN/Bluetooth** teilen sich WLAN (Wi-Fi) und Bluetooth (BLE) typischerweise **eine einzige Antenne**.

Das liegt daran, dass:

- WLAN und Bluetooth beide im **2,4 GHz-Band** arbeiten (Bluetooth ausschließlich dort, WLAN zusätzlich auch in 5 GHz bei Dualband-Modellen).
- Ein sogenannter **Antenna Switch / Diplexer** verwendet wird, um die Signale beider Funkmodule über die gleiche Antenne zu führen.
- Das spart Platz, Kosten und vereinfacht das Platinenlayout.

Beispiele:

- **Raspberry Pi 3 B / 3 B+ / 3 A+** – gemeinsame PCB-Antenne (bzw. Metallantenne bei 3 B+).
- **Raspberry Pi 4 B** – ebenfalls eine gemeinsame Antenne.
- **Raspberry Pi Zero W / WH, Pi Zero 2 W** – gemeinsame Antenne.
- **Raspberry Pi 5** – wieder eine gemeinsame Antenne, hier aber als externe Antennenbuchse (u.FL) optional nutzbar.

Folge: WLAN- und Bluetooth-Verkehr können sich gegenseitig beeinflussen, da sie zeitlich über den gleichen Antennenpfad laufen müssen. Das Funkmodul sorgt zwar für Koordination (Time Division Multiplexing), aber bei hoher Last auf beiden Schnittstellen kann die Performance etwas sinken.

Falls du willst, kann ich dir auch eine kleine **Übersichtstabelle** aller Raspberry-Pi-Modelle mit Angabe zur Antennenkonfiguration machen.

Möchtest du, dass ich das gleich ergänze?



ja

Hier ist die Übersicht aller Raspberry-Pi-Modelle mit integriertem WLAN/Bluetooth und deren Antennenkonfiguration:

Modell	WLAN	Bluetooth (BLE)	Antennentyp	Geteilt?
Raspberry Pi 3 Model B	2,4 GHz 802.11n	BT 4.1	PCB-Antenne	Ja
Raspberry Pi 3 Model B+	2,4 + 5 GHz 802.11ac	BT 4.2	Metallpatch-Antenne	Ja
Raspberry Pi 3 Model A+	2,4 + 5 GHz 802.11ac	BT 4.2	Metallpatch-Antenne	Ja

Modell	WLAN	Bluetooth (BLE)	Antennentyp	Geteilt?
Raspberry Pi Zero W	2,4 GHz 802.11n	BT 4.1	PCB-Antenne	Ja
Raspberry Pi Zero WH	2,4 GHz 802.11n	BT 4.1	PCB-Antenne	Ja
Raspberry Pi Zero 2 W	2,4 GHz 802.11n	BT 4.2	PCB-Antenne	Ja
Raspberry Pi 4 Model B	2,4 + 5 GHz 802.11ac	BT 5.0	Metallpatch-Antenne	Ja
Raspberry Pi 400	2,4 + 5 GHz 802.11ac	BT 5.0	Metallpatch-Antenne	Ja
Raspberry Pi 5	2,4 + 5 GHz 802.11ac	BT 5.0	Externe u.FL-Buchse + mitgelieferte Antenne	Ja
Raspberry Pi Compute Module 4	2,4 + 5 GHz 802.11ac	BT 5.0	Onboard-Metallpatch oder u.FL	Ja

 **Fazit:** Bei allen bisher erschienenen Raspberry-Pi-Boards mit integriertem Funkmodul wird **eine einzige Antenne für WLAN und Bluetooth** genutzt. Das Funkmodul enthält einen RF-Switch, der zwischen beiden umschaltet.

Wenn du magst, kann ich dir auch noch erklären, **wie genau dieser Umschaltmechanismus (Antenna Sharing) auf Hardware-Ebene funktioniert** – das ist ziemlich interessant, weil es erklärt, warum BLE-Streaming und WLAN-Upload gleichzeitig manchmal ruckeln.



Verliert man dann BLE Bandbreite wenn man Daten über 2.4GHz Band sendet?

Ja – und zwar technisch bedingt.

Wenn dein Raspberry Pi gleichzeitig **BLE** und **WLAN im 2,4 GHz-Band** nutzt, teilen sich beide nicht nur die **Antenne**, sondern auch die **Funk-Hardware-Zeit**.

Warum das passiert

- WLAN und Bluetooth können nicht gleichzeitig über dieselbe Antenne funken, weil sie sich sonst gegenseitig stören würden.
- Im Funkchip steckt daher ein **Time-Division-Multiplexing**-Mechanismus: Er schaltet in sehr kurzen Zeitfenstern zwischen WLAN und Bluetooth hin und her.
- Wenn WLAN-Datenverkehr viel Zeit beansprucht (z. B. ein großer Upload oder Stream über 2,4 GHz), bleibt weniger „Sendezeit“ für BLE übrig → die **BLE-Durchsatzrate sinkt** und die Latenz kann steigen.

Praktische Auswirkungen

- Bei geringer WLAN-Last: BLE-Bandbreite nahezu unverändert.
- Bei hoher WLAN-Last im 2,4 GHz-Band: BLE-Datenrate deutlich reduziert, teilweise Paketverluste.
- Besonders kritisch bei BLE-Audio- oder kontinuierlichen Sensordaten.

Möglichkeiten zur Verbesserung

1. **WLAN auf 5 GHz umstellen** → BLE bleibt allein im 2,4 GHz-Band.
2. Falls möglich, **BLE auf minimal nötige Datenrate** optimieren (z. B. größere Connection Interval).
3. Bei sehr kritischen BLE-Anwendungen: Externen USB-Bluetooth-Dongle nutzen → eigene Antenne.

Ich könnte dir als Nächstes eine **grafische Zeitscheiben-Übersicht** zeigen, wie der Raspberry Pi die Antenne zwischen BLE und WLAN aufteilt. Das macht das Zusammenspiel sehr anschaulich.



Verliert man dann BLE Bandbreite wenn man Daten über 5GHz Band sendet?

Nein – zumindest **nicht aus diesem Antennen-Sharing-Grund**.

Wenn dein Raspberry Pi über **5 GHz-WLAN** sendet, passiert folgendes:

- **BLE bleibt im 2,4 GHz-Band** → dort ist es allein unterwegs, kein direkter Wettbewerb um die Funkzeit mit WLAN.
- Die Antenne ist zwar physisch dieselbe, aber die Funkchips haben **separate Signalfade** für 2,4 GHz und 5 GHz.
- Es gibt keinen zeitlichen „Umschaltzwang“ zwischen BLE und 5 GHz-WLAN, weil beide in unterschiedlichen Frequenzbereichen arbeiten.

🚩 **Aber:**

Es *kann* immer noch minimale Bandbreiten- oder Latenzänderungen geben, weil sich WLAN- und Bluetooth-Stack im Betriebssystem (CPU, Interrupts, DMA) Ressourcen teilen. Das ist aber deutlich weniger problematisch als bei 2,4 GHz-WLAN.

💡 **Kurzfassung**

- **2,4 GHz-WLAN + BLE** → spürbare BLE-Drosselung bei hoher WLAN-Last.
- **5 GHz-WLAN + BLE** → praktisch keine Drosselung durch den Funkchip.

Falls du willst, kann ich dir eine kleine Vergleichstabelle machen, wie sich BLE-Durchsatz bei 2,4 GHz- und 5 GHz-WLAN typischerweise unterscheidet.



Vergleichstabelle

Vergleichstabelle: Einfluss von WLAN (2,4 GHz vs. 5 GHz) auf BLE (praktische Erwartungen)

WLAN-Band / Last	Erwarteter Einfluss auf BLE-Durchsatz (praktisch)	Erwarteter Einfluss auf BLE-Latenz	Kurz-Kommentar
2,4 GHz — keine/negligible	Kein merklicher Verlust — BLE erreicht typ. nahe seiner	Gering	Beide teilen Antenne, aber wenig Verkehr → Time-

WLAN-Band / Last	Erwarteter Einfluss auf BLE-Durchsatz (praktisch)	Erwarteter Einfluss auf BLE-Latenz	Kurz-Kommentar
WLAN-Last	normalen Rate (BLE4.x ~0.3-0.7 Mbps praktisch; BLE5 2M PHY prakt. ~0.8-1.5 Mbps).		slicing kaum spürbar.
2,4 GHz — geringe WLAN-Last (sporadisch)	Leichter Rückgang bei kurzen Spitzen, meist vernachlässigbar.	Kurzfristige Jitter-Spitzen möglich	BLE-Pakete können kurz warten müssen (Intervall-Scheduling).
2,4 GHz — mittlere WLAN-Last (kontinuierlich)	Deutlicher Rückgang: BLE-Durchsatz kann spürbar unter normalen Werten liegen (je nach Stack/Chip z. T. 30-70% Reduktion).	Erhöhte Latenz & Jitter	Time-division fürchtet BLE-Slots; Paketverluste möglich.
2,4 GHz — hohe WLAN-Last (saturiert)	Starke Drosselung — BLE kann stark reduziert werden oder intermittierend ausfallen (je nach Koexistenz-Implementierung).	Deutlich höhere Latenz / Paketverluste	Kritische Anwendungen (Audio, niedrige Latenz) betroffen.
5 GHz — keine/negligible WLAN-Last	Kein Verlust — normales BLE-Verhalten (siehe Werte oben).	Gering	5 GHz nutzt anderes Frequenzband → kein RF-Konflikt.
5 GHz — geringe bis hohe WLAN-Last	Praktisch kein RF-bedingter Verlust; BLE-Durchsatz bleibt nahe Normal.	Nur minimale Auswirkungen (CPU/Stack-Overhead möglich)	Störungen nur durch System-Ressourcen (Interrupts/CPU), nicht durch Antenne/Frequenz.

Wichtige Zusatzinfos (kurz)

- **BLE-Version** macht viel aus: BLE5 (2M PHY) kann deutlich höhere Rohdatentraten als BLE4.x erreichen — realistisch sind aber immer deutlich niedrigere „Anwendungsdurchsätze“.
- **Koexistenz-Mechanismen:** Moderne Wi-Fi/Bluetooth-Chips setzen Adaptive Frequency Hopping (AFH), Time Division oder andere Mechanismen ein; deren Qualität bestimmt, wie schlimm die Drosselung wird. Hersteller-Implementationen variieren.
- **System-Overhead:** Selbst bei 5 GHz-WLAN kann CPU-/USB-/Kernel-Last Einfluss auf BLE-Latenz/Durchsatz haben — aber das sind meist kleinere Effekte als ein 2,4 GHz-RF-Konflikt.
- **Externer USB-BT-Dongle** mit eigener Antenne umgeht das Sharing (wenn er einen eigenen Funkchip hat) — gute Lösung für kritische BLE-Anwendungen.

Empfehlungen

1. **Wenn möglich WLAN auf 5 GHz betreiben** (Router/Client unterstützen?) — das beseitigt den RF-Sharing-Konflikt.
2. **BLE-Parameter optimieren:** größeres Connection Interval, größere ATT-MTU, Bulk statt viele kleine Pakete.
3. **Bei harten Echtzeitanforderungen:** externen BT-Dongle verwenden oder Gerätetrennung (separate Hardware).

4. **Testen:** Mache Messungen in deiner konkreten Umgebung (z. B. BLE-Durchsatz/Latency-Tool) — reale Werte variieren stark nach Chip/OS/Stack.

Willst du, dass ich dir eine **kleine Test-Checkliste** und ein einfaches Messskript (z. B. für `gatttool/bluetoothctl` oder ein Python-Skript) zusammenstelle, mit dem du die Auswirkungen bei dir vor Ort messen kannst?