

# Bluetooth shifts up a gear: Intelligent tools solve in-vehicle RF range and interference issues

In-vehicle broadband services offer a huge potential market for Bluetooth technology. But interference from existing in-vehicle RF devices can be detrimental to Bluetooth broadband signals and existing interference defenses may not provide adequate protection. Channel quality driven data rate (CQDDR) is a Bluetooth option that promises to ensure optimum data throughput.

By Ken Noblitt

**H**ands-free mobile phone operation in vehicles offers a huge potential market for Bluetooth, and kits are already finding their way into both after-market solutions and embedded designs. Alongside hands-free mobile phone operation, future Bluetooth applications in-vehicle will include Internet access, wireless additions to the infotainment system, vehicle personalization and even vehicle diagnostics.

To make all of this possible, there are problems that must be overcome.

Other RF devices, such as car stereos, global positioning system (GPS) navigation equipment, satellite digital audio radio services (SDARSes), GSM transceivers and other electrical devices, that are already in-vehicle can cause interference or be susceptible to interference.

Also, think of a car as a reflective *tin-can*, where radio waves are reflected within the vehicle cabin, which results in a phase shift that, with superposition, can effectively cancel out or corrupt the wanted signal.

All this RF activity can be detrimental to the data throughput of an in-vehicle wireless system. As the applications for Bluetooth expand, Bluetooth modules are likely to become more widespread around the car, further compounding the potential interference risk.

## Existing interference defenses

Bluetooth already has an existing arsenal of defenses to combat interference designed into the standard. But the unique and difficult conditions within a car have forced designers to sharpen their swords to lessen the considerable potential effects of interference.

One of Bluetooth's standard weapons is frequency hopping, which requires both the receiver and transmitter to tune/hop to one of its 79 different channels 1,600 times per second in a predetermined pattern.

This provides a good level of immunity to interference. If a data packet is lost or received incorrectly, no acknowledgement will be returned, and a retransmission will be

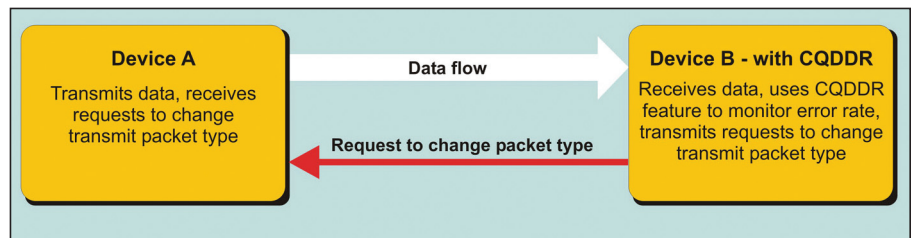


Figure 1. Effective bandwidth plotted against increasing BER.

sent at the next available time slot.

By utilizing frequency hopping technology, and with care taken to separate the Bluetooth antennas from other sensitive in-cabin receivers or transmitters, interference can be minimized. But even with these measures in place, a high amount of in-vehicle RF activity can be detrimental to data throughput and link reliability.

less equipment operating in the same band (such as Wi-Fi).

The Bluetooth standard offers a range of data packet types, known as *DH* and *DM*. *DH* types have higher payloads; *DM* types support only medium payloads, but include forward error correction for integrity in noisy operating conditions.

All have cyclic redundancy checks (CRCs)

*To ensure a good user experience, packet types must be chosen not just on the basis of how much data is waiting to be sent, but also on the ambient error conditions, which can greatly vary in a vehicle cabin.*

## Channel quality driven data rate (CQDDR)

Channel quality driven data rate (CQDDR) is a Bluetooth option that ensures that products will achieve the optimum data throughput.

Most Bluetooth products support all data packet types. However, if the IC's firmware decides to employ a type that doesn't match the noise environment, a designer can end up with a very inefficient communications link. This is because of noise, caused for example, by users pushing products beyond their limits for range and noise immunity, and other wire-

for error detection, but this cannot correct errors. The payloads of *DM* packets are divided into blocks of 10 bits, with an additional five bits for forward error correction. These can be used to detect and correct all one-bit errors, and to detect and reject all two-bit errors in each 10-bit block.

Although *DH* packets, particularly *DH5*, seem to give the best performance, in the real world things aren't so simple. To ensure a good user experience, packet types must be chosen not just on the basis of how much data is waiting to be sent, but also on the ambient error conditions, which can greatly

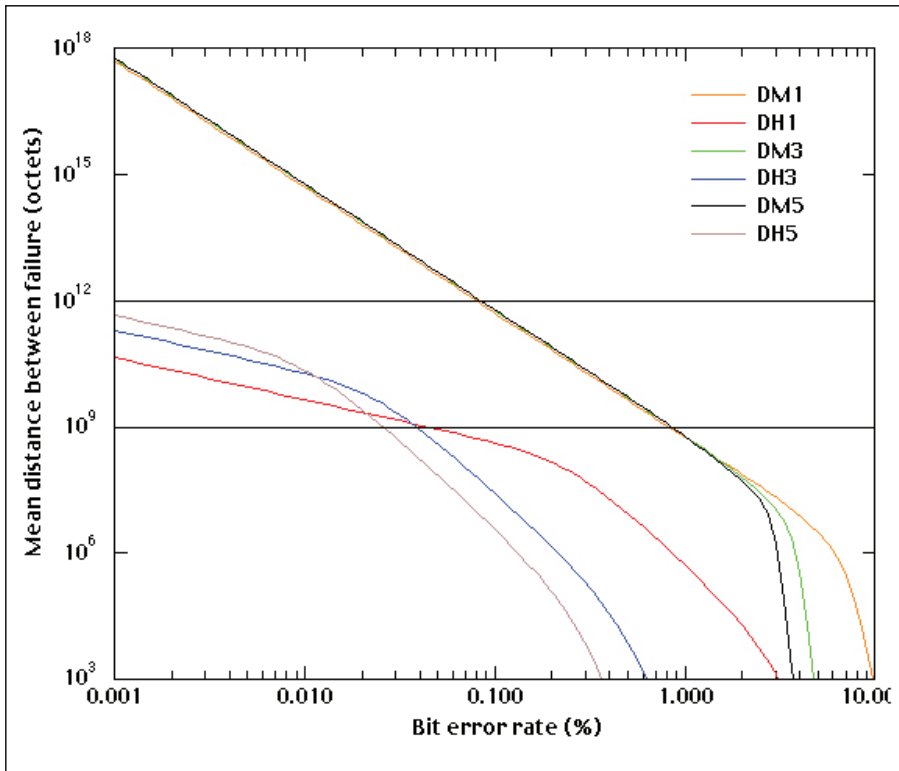


Figure 2. CQDDR lets Bluetooth device dynamically switch packet types according to link conditions.

vary in a vehicle cabin. There are several possible ways this can be achieved within the current specification.

Designers of each layer of the stack can choose either to add an extra layer of error detection, and optional correction, to their layer, or they can pass on the data unmodified.

The former allows them to make a stronger reliability guarantee than the layer below, whereas the latter keeps the layer overhead low.

A simplistic baseband implementation may choose the packet type purely on the number of octets it has waiting to be sent. This means that when there are many octets waiting, it will choose DH5 packets, as these have the largest payload.

In laboratory conditions, this may look like a good solution, but users will ultimately be frustrated. At a bit error rate (BER) of 0.04 percent, for example, DH5 packets have only a 33 percent chance of being received without error.

*CQDDR allows a receiving device to negotiate with the transmitter to change the transmitted packet type according to the conditions experienced at the receiver.*

In particular, the overhead for error correction (as opposed to mere detection) is high. Typically, all the data sent by a layer must be buffered in that layer until it is acknowledged (implicitly or explicitly) by the other side. As flow progresses up the stack, the data transfer latencies get higher, and, hence, the amount of buffering increases.

In other words, it will take an average of three attempts to send a packet — reducing potential maximum bandwidth from 723 kbps to just 241 kbps.

On the other hand, DM5 packets show virtually no degradation from their maximum bandwidth of 477 kbps until 10 times that BER.

There is a mechanism built into the

Bluetooth scheme (version 1.1 of the Bluetooth specification), which, if correctly implemented, solves range and constant interference issues, that being CQDDR.

CQDDR allows a receiving device to negotiate with the transmitter to change the transmitted packet type according to the conditions experienced at the receiver. It requires access to packet and BER statistics to make an intelligent trade-off between bandwidth and data integrity. With CQDDR, if one side finds that it's receiving packets with many errors, it tells the other side to switch to DM packets. If the link clears up, then it can allow the other side to use DH packets again.

Surprisingly few companies have implemented CQDDR in their firmware in spite of the great improvements it brings to data transfer and link reliability in environments, such as a vehicle cabin.

The incredible amount of in-vehicle functionality that Bluetooth could enable requires thought and foresight to implement. By taking care to consider the entire vehicle system, automotive designers can avert many of the problems associated with implementing a Bluetooth system.

Interference control is a vital element for any RF designer, and the vehicle environment only magnifies this factor. With a number of safeguards already put into place in the Bluetooth standard, designers are almost there. CQDDR offers the opportunity for automakers and OEM suppliers to ensure interference is not an issue and to deliver on Bluetooth's promise of a multifaceted wireless connectivity.

ζ

#### ABOUT THE AUTHOR

Ken Noblitt is the technical development director for CSR Inc. ([www.csr.com](http://www.csr.com)). He holds an M.S.E.E. from Southern Methodist University, a B.S.E.E. from The University of Texas at Arlington, and has 22 years of experience in many different spread spectrum systems in the commercial, government and military sectors.

He joined CSR in October 2000 from NEC Corp.'s ([www.nec.com](http://www.nec.com)) semiconductor operations, where he was the field applications manager for system-on-a-chip (SoC) solutions, including wireless, USB, ATM, DSL, and custom high-density ASIC solutions. Noblitt also served as an R&D program manager and CDMA architect at Nokia Corp. ([www.Nokia.com](http://www.Nokia.com)), and a senior systems engineering manager at The Boeing Co. ([www.boeing.com](http://www.boeing.com)) and E-Systems (now Raytheon Co. at [www.Raytheon.com](http://www.Raytheon.com)). He can be reached at [ken.noblitt@csr.com](mailto:ken.noblitt@csr.com).