WHITEPAPER: Reliable Wireless Connectivity for IoT and Sensor devices

RELIABLE WIRELESS CONNECTIVITY FOR IOT AND SENSOR DEVICES

Abstract:
When designing products relying on wireless communication, reliability of the communication link is important. There are many factors influencing the quality of the link, and thereby the likelihood for successful delivery of payload data. Multiple wireless technologies exist which are marketed for many different applications – IoT, Industrial IoT (Industry 4.0), Wireless Sensors etc.. These technologies are all claiming to be the optimal solution for almost any application out there. However, not many are highlighting the actual performance when it comes to payload data delivery reliability.
This whitepaper provides details into the parameters to consider when selecting wireless technologies from a reliability point of view, and how to achieve cable-like reliability in a wireless connectivity solution.

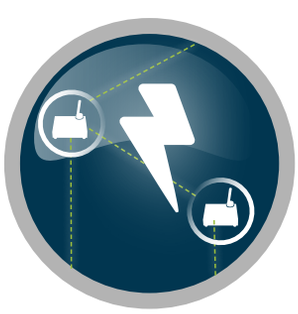
ARE CABLES **RELIABLE?**

A lot of industrial sensor networks, or in general sensor and control networks are based on wired communication. The actual implementation can vary, but often some type of FieldBus is being applied.

The reliability of these systems is very high, and is often considered to be "100%" reliable – but in reality even wired systems are not 100 % reliable. Just consider the case where the cable is damaged in some way, either completely broken so signal is lost, or maybe just slightly damaged resulting in intermittent connection. In addition to damaged cables, wired systems are also susceptible to noise which leads to bit errors in the payload data received. Overall, in most cases, a wired solution is fairly easy to shield off from noise, and as such the most critical part to consider would be the cable-damage scenario. Depending on the actual installation, the cable may be more or less exposed to damage. Cables are often hidden away – in installation shafts, or even under ground. So when a cable is damaged in some way, the detection of the damage and the repair can be troublesome. These type of systems are supposed to work for many years - which exacerbates the situation further as longer service life increases the likelelyhood of cable damage.
In systems where high communication reliability is required, redundant communication channels may be applied. Typically this is achieved by running multiple cables. So the reliability of cabled solutions comes at a cost.

WIRELESS COMMUNICATION **– ASKING FOR TROUBLE?**

Wireless communication using radio waves has been used for ages and is attractive as it alleviates the need for running cables. That in many cases reduces the cost of installation and even allows for communication links in situations where not otherwise possible.

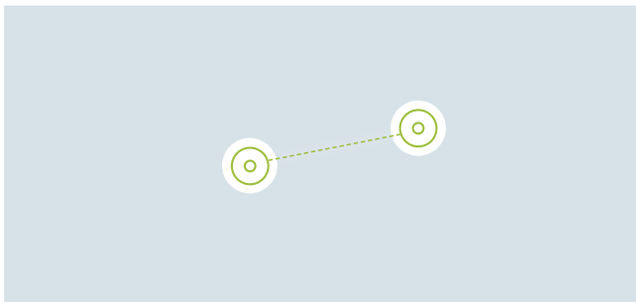


Also, if the network is dynamic and subject to changes over time, wireless is the easiest and cheapest solution. The caveat obviously is that the communication channel is not as well protected as with cables.
In general, wireless communication is either done in licensed or unlicensed spectrum. In licensed spectrum there is a fairly high control of how the particular part of the spectrum is used, and as such the unwanted radio signals – or noise – is limited. In unlicensed spectrum (ISM and SRD bands) however, there is no control over how much unwanted radio energy is present. This is often seen in the 2.4GHz spectrum where WiFi network performance can be highly degraded in areas where many networks are installed. Does this mean that wireless communication in unlicensed spectrum is not a good solution in applications requiring a high degree of reliability? Not necessarily. Let us consider some typical wireless communication methodologies, and how they differ from a reliability point of view:

WIRELESS COMMUNICATION **METHODOLOGIES**

The simplest form of wireless (digital) communication, is a setup where one device is trying to send some data to another.

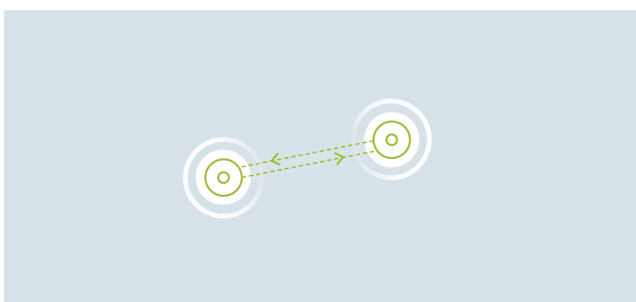
Point to point



The systems consists of a transmitter (TX) and a receiver (RX). When the transmitter sends the payload data, the receiver picks it up. Now, since the wireless communication channel is susceptible to noise, the received payload data may not be identical to the transmitted – a bit error may have occurred.
If the receiver side of the system does nothing, the erroneous data will enter the rest of the system and potentially cause problems. Most systems however, employ some sort of error detection mechanism. Usually CRC (CRC – Cyclic Redundancy Check) is being used which is based on adding additional information to the payload (redundancy) which is calculated by the transmitting device based on the payload data. This information can then be used by the receiving device to check if there is a bit error in the payload. In the setup considered here where the communication happens only one way (from TX to RX), the only thing the receiver can do if it detects a problem in the received data, is to discard it

A more advanced version of the point to point structure, is one where both sides are capable of transmitting and receiving, allowing for bidirectional communication. In such a setup, the devices on both sides are labelled "transceivers".

Bidirectional point to point



The advantage of applying bidirectional communication is that the receiver can signal back to the transmitter if the payload data was received ok (Acknowledge) or not (Non-Acknowledge). This allows for retransmission of the payload should it not have been received correctly at the receiver side.
Different technologies use different approaches. Wireless MBUS for instance, uses unidirectional communication, where no ACK/NACK is possible. Instead, most meters implementing Wireless MBUS relies on sending the same payload package multiple times spread over time, with the idea that at least one of the duplicate transmissions will be received by the receiving device.

No matter what methodology is being applied, the error detection capability (CRC) – and potentially ACK/NACK is an important factor to ensure reliable wireless communication.

IS **CRC** ENSURING CORRECT DATA?

Cyclic Redundancy Checking is optimized at detecting burst errors of short length, which are common in wireless communication channels. The CRC checksum can be of different length. Most embedded SoC's implements a CRC16 hardware accelerated calculation block. CRC16 uses a 16 bit checksum, and as a rule of thumb, the CRC checking will catch all bit errors which are shorter than the checksum size. This means that burst errors shorter than 16 bits will be caught by the CRC16 error checking. Longer burst error (length > 16 bit) will be caught with 1-2-n likelihood, where n is the checksum length.

For CRC16 this means that burst errors larger than 16 bits will be caught with 99,998474% change. This means that payload messages received which have bit errors, may be checked ok by the CRC error check algorithm. In a sensor (and control) network, where thousands of payload packages are delivered each day, at least a few packages will be accepted as "good" although they are actually "bad".
CRC checking with longer checksum is helping to decrease the likelihood for errors. If for instance the CRC checksum is 32 bits, the burst error length will have to be longer to go undetected, and the likelihood of letting through bad packages with burst errors which are longer than 32 bits, will be 99,99999997671%. Significantly better than CRC16 obviously.

No matter what CRC checksum size is being used, there is a risk of letting through packages which contain bit errors. Using CRC16, the likelihood is considerable in typical sensor networks, whereas with CRC32, the likelihood drop to a level which is close 0% - even in application where thousands of packages are transmitted.

SINGLE POINT OF **FAILURE**

In wireless systems based on a point-to-point topology, or even a star network topology, if the link between the sending device and the receiving device is no longer reliable, then the communication breaks down – there is no alternative path for the data to flow. The link between two devices can break for different reasons; it may be due to noise, or because the link is obstructed either temporary or permanently by some object.
The noise level will vary from location to location, and also over time. This means that even though a system installed in one location works flawlessly, it may not work as reliably in another location. Similarly, a system may work with no issues at install time, but sometime later, issues may start to occur.
It could for instance be due to other systems being installed which operates in the same frequency band.

NEOMESH – DESIGNED FOR RELIABILITY

NeoMesh is a connectivity solution based on mesh topology, and includes a range of features specifically targeted at increasing reliability.

First of all, the mesh topology ensures redundant links where the payload data can flow. If one link breaks down, alternative paths can be found. The patented routing protocol, Speed Routing, is optimized for low power mesh network applications, and ensures that data will be routed along the fastest path towards the destination, handling real time any connection problems either due to noise or link blockage.

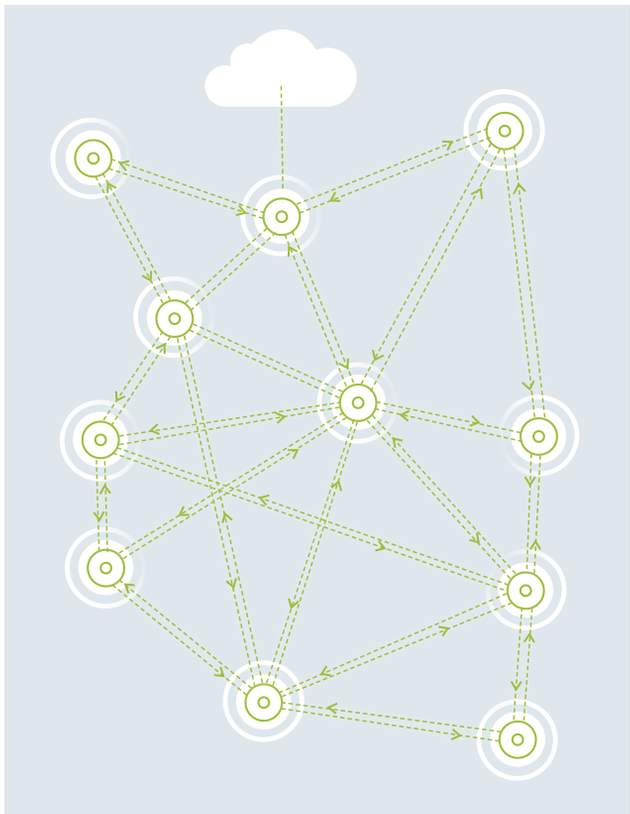
When payload data travels node to node from source to destination in the NeoMesh network, the package exchange is governed by local ACK/NACK between the sending and receiving node.
If the package transfer between two devices fails, it will be retried. The ACK/NACK is based on CRC checking using a 32 bit checksum ensuring a very high degree of probability that a package which is received and acknowledged is free from bit errors.
In addition to the ACK/NACK which happens at each hop along the way from source to destination, NeoMesh also supports end-to-end acknowledge. This implies that once a payload package is delivered at the destination, a ACK message will be routed back to the source node notifying the application layer that the message was delivered successfully. All this is handled automatically by the NeoMesh protocol layer.

FREQUENCY HOPPING

In order to further increase the reliability of the connectivity solution, NeoMesh is using frequency hopping which spreads the communication over the entire frequency band. This is a very effective approach to avoid "dead" channels which are used heavily by other communication systems. The fact that the frequency hopping continuously uses a new channel for each communication burst, ensures that even if new noise sources are popping up during the lifetime of the installation, the NeoMesh based installation will effectively avoid the noise if possible.

STRONG ENCRYPTION

Reliability is not only a matter of ensuring data is delivered at the destination, it also implies that the correct data is delivered. CRC checking as previously discussed is good for making sure there a no bit errors, but it does not guarantee that payload data was not generated by a malicious source.



NeoMesh has two strong features built into the core of the protocol stack which increases the reliability even further by securing the communication link:
First off, all data exchanged between nodes in the network, is encrypted using industry standard AES128. The key for the AES, is the network key which is programmable by the system configurator, and stored securely in each module.
Secondly, when using fully acknowledged communication between the source and the destination, there is an automatic challenge-response handshake implemented which ensures that a destination node will only accept payload data from another node (source) if the payload data includes the correct response to a challenge given by the destination. This challenge-response authentication is unique to NeoMesh, and is handled completely seamless by the protocol stack. It prevents so-called playback attacks where a malicious device records a previously sent payload package which may contain a certain control function like for instance "unlock door" or similar. The perpetrator later re-transmits (playback) the message which would unlock the door. With challenge response authentication this is not possible.

NEOMESH – CABLE-LIKE RELIABILITY

As discussed in this whitepaper, wireless communication is potentially very unreliable. The communication channel is open, and there are many systems competing for the same signaling path. Yet we claim that NeoMesh offers cable-like reliability. Why? Because NeoMesh addresses these issues, as it is designed with reliability in mind, while being optimized for ultra-low power consumption for all the nodes in the network. Not many low power connectivity solutions offer a similar degree of reliability.

A unique combination of NeoMesh features provides for a very robust and reliable system: Redundancy due to the mesh network topology, the frequency hopping scheme to avoid blocked channels and the 32 bit CRC based ACK/NACK, which sets new standards for error detection in this type of connectivity solution. Adding to this the security features result in a wireless connectivity solution which is a very good match for applications requiring high quality of service. Thus NeoMesh can easily replace cables in many solutions where reliability is a requirement.