Demo: mSync in Action

Bastian Bloessl and Falko Dressler
Distributed Embedded Systems Group, University of Paderborn, Germany
{bloessl,dressler}@ccs-labs.org

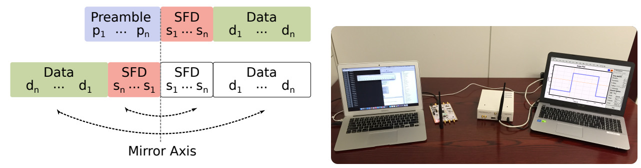
1. INTRODUCTION

In our daily life we are surrounded by a vast amount of low-power wireless systems used for industrial automation, weather stations, and car key fobs, for example. Driven by the idea of an Internet of Things these systems are about to become even more ubiquitous. Most of these networks have two things in common: First, the network structure is heterogeneous with simple energy constrained sensors and more sophisticated sinks. Second, they transmit in bursts using frame-based single-carrier systems. Usually, the frames of those systems have a similar structure. They consists of preamble symbols for synchronization, followed by a Start of Frame Delimiter (SFD) and the actual data payload. Given the small maximum frame sizes, the preamble and SFD can introduce considerable overhead in terms of energy consumption and occupancy of the wireless channel.

A well-known example of such a system is the IEEE 802.15.4 O-QPSK PHY. This PHY is used in many low-power systems and forms the base of the ZigBee communication stack. With the IEEE 802.15.4 PHY, the overhead is 5 bytes per frame, consisting of 4 bytes preamble symbols and 1 byte SFD. This has to be compared to a maximum frame size of 127 bytes or an ACK size of 5 bytes, where the overhead is 100 %.

2. mSync

Recently we proposed mSync [1], a frame format and decoding strategy that does not rely on preamble symbols. A comparison of a normal frame and of mSync is shown in Figure 1a. The colored boxes depict the parts of the frame that are actually sent over the air. With mSync, the frame starts immediately with the reversed data symbols, followed by the reversed SFD. Instead of using the preamble, an mSync receiver uses the data symbols to synchronize on the frame. In a first pass, the receiver cannot decode the data, but uses the signal solely as training data. Since the receiver locks on the signal with the help of the data symbols it is able to detect the SFD following the payload. Once found, receiver traverse the frame again in reverse direction outputting the demodulated symbols. The crucial point is that the receiver stays locked while switching directions. Since the data is sent reversed, the output of the receiver, produced while going backwards through the signal, corresponds to a normal frame (cf. Figure 1a). Therefore, the rest of the receive chain can be left unchanged.



(a) Normal frame format (↑) vs. mSync (↓).
(b) Photo of the demonstration setup.

Figure 1: Comparing normal frames and mSync.

While decreasing the overhead of transmissions, the approach comes with drawbacks. The most obvious is the increased memory demand that results from storing samples corresponding to a maximum sized frame. Furthermore, data symbols do not guaranteed to offer good convergence for the synchronization algorithms. This could lead to longer lock times and maybe even ask for padding if the data is too short. The downsides of the approach are currently investigated and will be part of future work.

3. DEMONSTRATION

To show the feasibility of the approach, in particular that it is possible to stay synchronized while switching directions, we implemented the algorithm on Software Defined Radio (SDR). More precisely, we extended our GNU Radio IEEE 802.15.4 transceiver [2] to support mSync. GNU Radio is a real-time signal processing framework that already provides all means to build a complete transceiver systems. The demonstration setup is depicted in Figure 1b. We use USRPs from Ettus to send dummy data live over the air while switching between normal and mSync frames. On the one hand we output the data, showing that both modes carry the same payload, while, on the other hand, we plot the frames in time domain, where the shorter mSync frames are clearly visible.

Acknowledgements

This work has been supported in part by the German Research Foundation (DFG) under grant no. FOR 1508.

4. REFERENCES

[1] B. Bloessl and F. Dressler. mSync - Frames without Preambles. In *ACM MobiCom 2015, SRIF Workshop, Poster Session*, Paris, France, September 2015. ACM.

[2] B. Bloessl, C. Leitner, F. Dressler, and C. Sommer. A GNU Radio-based IEEE 802.15.4 Testbed. In *GI/ITG FGSN*, pages 37–40, Cottbus, Germany, September 2013.

S3'15, September 11 2015, Paris, France

ACM 978-1-4503-3701-4/15/09.
http://dx.doi.org/10.1145/2801694.2801698 .