Micro-Frame Preamble MAC for Multihop Wireless Sensor Networks

Abdelmalik Bachir and Dominique Barthel
France Telecom R&D
Meylan, France
{Abdelmalik.Bachir, Dominique.Barthel}@francetelecom.com

Martin Heusse and Andrzej Duda
LSR-IMAG Laboratory
Grenoble, France
{Martin.Heusse, Andrzej.Duda}@imag.fr

***Abstract*— MAC protocols based on preamble sampling techniques like WiseMAC and BMAC offer significant energy savings for multihop wireless sensor networks. Although preamble sampling based techniques efficiently reduce idle listening in low-rate data networks, they introduce extra overhead due to possible reception of some irrelevant frames. In this paper, we identify the problem of receiving irrelevant frames and propose to replace the continuous preamble by a series of small frames that we call *micro-frames*. The resulting Micro-Frame Preamble (MFP) scheme makes it possible for a node to switch its radio off to avoid receiving irrelevant frames and improve energy savings. In the paper, we evaluate the theoretical energy savings of MFP compared with traditional preamble sampling techniques. We also discuss some experiences with our implementation of MFP on the Freescale MC13192 SARD.**

I. Introduction

One of the main objectives of MAC (Medium Access Control) protocols for wireless sensor networks is to minimize power consumption while providing reliable low-rate data transmission. Previous studies have identified multiples sources of energy dissipation at the MAC layer, the most important are the following [1]:

- *Idle Listening*: since a node does not know when it will be the receiver of a frame, it must keep its radio in receive mode, which consumes energy.
- *Collisions*: the energy used to send a frame is wasted when the frame collides with another frame.
- *Overhearing*: this happens when a sensor node receives and decodes an irrelevant frame (e.g. a frame not addressed to the node).
- *Protocol Overhead*: control frames do not carry useful information although their transmission consumes energy.

Sensor networks applications usually generate low traffic load so that the communication channel is expected to be idle most of the time. Therefore, idle listening is the most significant source of energy dissipation. Indeed, without any specific energy management, nodes waste considerable amounts of energy as they keep their radios on for large time intervals while listening to an idle channel. To mitigate idle listening, in energy-efficient MAC protocols for sensor networks nodes sleep for long periods of time instead of being active permanently. These MAC protocols define a duty-cycle parameter to control the ratio of the activity period to the sleep period [2], [3].

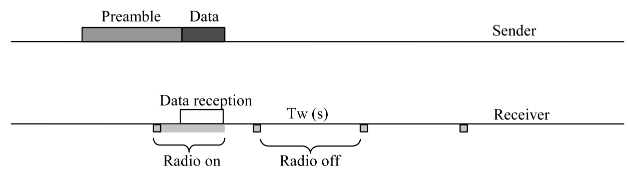

Fig. 1. Preamble sampling technique.

Duty cycles of 1% may substantially reduce energy consumption. However, if a node wakes up only at some chosen instants to mitigate idle listening effects, there is a need for a method to detect transmissions of other nodes. There are two ways for making it possible for nodes to communicate without message loss due to a transmission to a node that is sleeping. The first approach is used in protocols like SMAC [1], TMAC [4] and others: nodes synchronize on a common sleep/wakeup schedule by exchanging synchronization messages to set their sleep/wakeup schedule. The second way used in protocols like WiseMAC [5] and BMAC [6] does not fix a common schedule for sleep and wakeup periods to avoid synchronization overhead and to further reduce idle listening in low-rate data networks. In this second approach, each node chooses its own sleep/wakeup schedule independently of the others and a node transmits a preamble before each data frame which is long enough to make sure that all potential receivers will get their data. For the rest of the paper, we focus on the protocols following the second approach that we call *preamble sampling* MACs (PS-MACs).

Fig. 1 shows an example of a PS-MAC protocol operation. According to the duty-cycle parameter, nodes switch periodically (e.g. each *Tw* seconds) their radios on to sample the channel. If a node finds the channel is idle, it goes back to sleep immediately. However, if it detects a preamble transmission on the channel, then it keeps its radio on until it receives the subsequent data frame. After the reception of the data frame, the node sends an ACK frame, if needed, and goes back to sleep afterward. To be effective, preamble transmission needs to be at least as long as the *check interval*, the period between two consecutive instants of node wakeup. In this way, a node makes sure that all potential receivers are awake during its preamble transmission and they get the subsequent data frame.

1-4244-0355-3/06/$20.00 (c) 2006 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

Although PS-MACs efficiently reduce idle listening and offer significant energy savings at the MAC layer, they still cannot avoid receiving some transmissions that are irrelevant for upper layers. Avoiding the reception of such data frames further contributes to energy savings. We have identified two kinds of irrelevant transmissions: the reception of the preamble and the reception of irrelevant data frames.

In PS-MACs, as soon as a node detects a preamble transmission, it keeps its receiver active until it gets the subsequent data frame. Without any particular optimization, this may consume large amounts of energy because a node receives in average half of the preamble each time it receives a data frame. El-Hoiyidi *et al.* [7] have tackled this issue and proposed a way for reducing the preamble length for unicast frames. Although the scheme contributes to saving more energy both at the transmitter and the receiver, the authors have not proposed any solution to avoid keeping receiving the preamble until data reception. Moreover, their proposal only applies to unicast frames and cannot be generalized to broadcast frames.

Similarly, PS-MACs have no means for identifying an irrelevant data frame before it is entirely received. Therefore, PS-MACs cannot avoid receiving irrelevant data frames. Irrelevant data frame, such as for instance data frames transmitted by neighbors during a flooding operation, may consume non-negligible energy because their reception also includes the overhead of their preambles as explained above.

To avoid these two types of irrelevant receptions, we propose a new scheme called *Micro-Frame Preamble* (MFP), in which we replace the continuous preamble by a series of small frames that we call *micro-frames*. Each micro-frame contains an indicator of the data frame contents such as the destination address or a hash on the data field. When a node wakes up, it can learn from a micro-frame whether the data frame is worth receiving, e.g. the node is the destination of the frame or the frame is a broadcast frame not yet received.

Our method efficiently copes with irrelevant receptions by:

- reducing the duration of the listening period to the reception of the preamble. With MFP, a node is able to estimate when the data will be transmitted. Therefore, it can switch off its receiver during preamble transmission and switch it on again only for data reception;
- filtering out irrelevant data frames without entirely receiving them, so that it is possible for a node to switch off its receiver during the reception of an irrelevant frame to save energy;
- reducing the duration of carrier sense that precedes data transmission. With MFP, a node is able to estimate when the whole transmission ends (preamble, data, ack). Therefore, it can switch off its receiver during preamble transmission and switch it on only at the end of the transmission to perform carrier sense;

II. Micro-Frame Preamble Medium Access Control

The key idea is to add to the preamble transmitted before each data frame in PS-MACs more information about the contents of the subsequent data frame. This allows a node to make a decision of receiving or not in a timely manner without having to keep listening to the preamble until the data frame. Hence, we propose to replace the continuous preamble with a series of micro-frames. In this way, it is possible to code some useful information within each micro-frame.

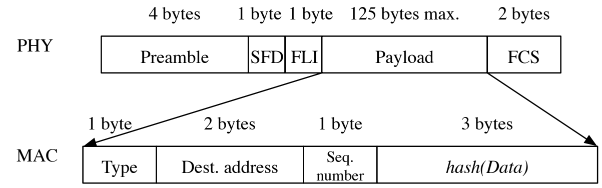

Fig. 2. Micro-Frame structure as implemented on the MC 13192 SARD.

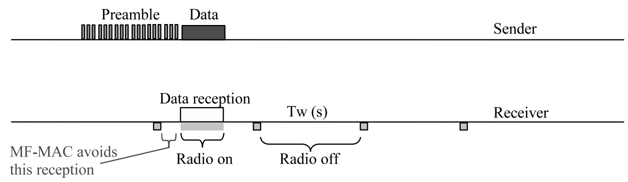

Fig. 3. Avoiding receiving an irrelevant preamble.

A. Reducing the Duration of Listening to the Preamble

Fig. 2 shows the structure of a micro-frame. It contains a type field so that a node is able to distinguish micro-frames from other frames such as data or acknowledgment frames. A micro-frame also contains a sequence number because we number micro-frames from 1 to m, where m is the amount of micro-frames needed so that their transmissions last at most the check interval *Tw*. When all the nodes share the same check interval, i.e. m is the same for all nodes, a node can estimate from the sequence number of the micro-frame when the data will be transmitted. This makes it possible for a node to switch off its radio during micro-frame transmissions and to switch it on again just to receive the data frame as shown in Fig. 3.

B. Avoiding Listening to Irrelevant Data

We assume simple applications that do not exploit overhearing and that there are data frames the reception of which may be irrelevant. Reception of irrelevant data happens in one of the two following situations: when a node receives a unicast frame not addressed to the node or when a node receives a redundant broadcast frame, i.e. the frame the node has already received.

To avoid receiving an irrelevant unicast frame, protocols such as Adaptive Sleeping [8] usually exploit RTS/CTS exchange to know about the destination of the data frame before its transmission. Thereby, the node can go to sleep in the meantime to save energy, if it is not the destination of the data frame. The idea is similar to the virtual carrier sense with NAV in the IEEE 802.11 DCF. However, this idea may be ineffective in PS-MACs in which the RTS/CTS exchange is

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

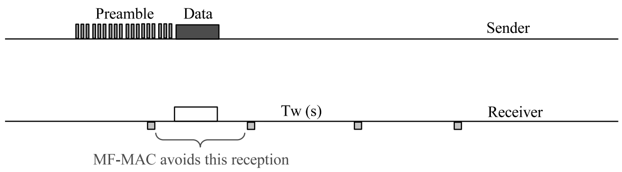

Fig. 4. Avoiding receiving an irrelevant data frame.

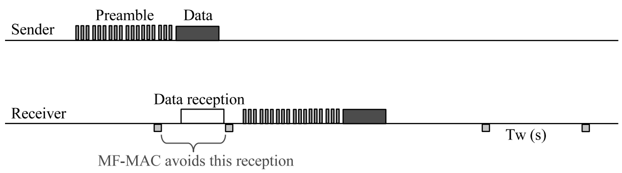

Fig. 5. Avoiding irrelevant reception during contention for the channel.

not encouraged because of its considerable overhead. Indeed, since RTS and CTS are broadcast frames, the optimization of preamble length proposed in [7] cannot be used. Therefore, their transmission consumes a lot of energy, both at the transmitter and at the receivers (as shown in Section II-A). To cope with this problem, we propose to put the destination address of the data frame into micro-frames. In this way, a node that wakes up to sample the channel, receives a micro-frame and knows whether it is the destination of the data frame or not. Fig. 4 shows an example in which a node learns from the reception of a micro-frame that it is not the destination of the data frame.

If the data frame is a broadcast, the destination address used in micro-frames does not help the node to know whether the data frame is redundant or not. That is why we envisage the hash-data field that contains a hash of the data and makes it possible for a node to know about the contents of the data frame without receiving it. Note that an MFP node has a table to keep trace of hash-data of frames it has already seen (sent or received), so it does not need to receive again.

One can argue there will be hash collisions that lead to false positives and a node may ignore a data frame that it has not recently seen because this data frame has the same hash as another data it has received. We think this situation is unlikely to happen for the following reasons. First, hash-field entries in the table of the MFP agent are not permanent, but cleared after a timeout value. Next, we expect there will not be a lot of simultaneous active broadcasts during an interval of a timeout value. We also expect the hash function to have good properties so that collisions are very rare.

C. *Other benefits*

Fig. 5 shows how MFP makes it possible to save energy when a node captures a micro-frame at the instant it performs clear channel assessment before sending a frame. It can read the sequence number and the destination address from the micro-frame to estimate the schedule of data transmission including the acknowledgment.

III. Performance Evaluation

A. *Comparing Performance of MFP with PS-MACs*

Our MFP protocol uses the preamble sampling technique and therefore consumes the same amount of energy in transmission as any PS-MAC protocol we are aware of. Also, the optimization of the preamble length by WiseMAC [5] can be applied to MFP. The main contribution of MFP is to reduce the energy consumed for reception in PS-MACs. To get an estimate of the amount of energy saved with MFP compared to PS-MACs, we propose to compare the mean time a node spends in receiving mode to receive one data frame in MFP with the one it spends to receive a data frame in PS-MAC.

1) Modelling of Reception Duration: Let $M(t,T)$ (resp. $P(t,T)$) be a random variable that corresponds the time interval during which the receiver is in receive mode in MFP (resp. in PS-MAC) to receive one data frame. T is a constant that denotes the transmission time of the data frame. We first analyze the gain obtained from avoiding to keep listening to the preamble. Then, we generalize and include the gain obtained from filtering irrelevant data. To model the avoidance of irrelevant data reception, we set T to 0.

Since each node chooses locally its schedule, the wakeup of a node to check for traffic is random in $[0, Tw]$. For MFP, we have

$$Tw = m(s+f), \tag{1}$$

where m is the number of micro-frames, f is the transmission time of one micro-frame and s is the inter micro-frame interval.

In PS-MAC, when the node wakes up and finds a preamble transmission in the channel, it keeps listening until it receives the data frame. Then, we have

$$P(t,T) = U_{[0,m(s+f)]}(t) + T. \tag{2}$$

In MFP, when the node wakes up and finds the channel busy, it needs to keep listening only until it receives a micro-frame. After receiving a micro-frame, the node switches off its receiver and switches it on again only to receive the data. The time needed to receive a micro-frame depends on when the node wakes up during the transmission of a micro-frame or during an inter micro-frame interval. We define P_f (resp. P_s), the probability that a node wakes up during the transmission of a micro-frame (resp. during an inter micro-frame interval). We have $P_s = s/(s+f)$ and $P_f = f/(s+f)$. Then,

$$M(t,T) = P_f \mu_f(t,T) + P_s \mu_s(t,T), \tag{3}$$

where, $\mu_f(t,T)$ (resp. $\mu_s(t,T)$) is the time a node spends to receive a data frame when it wakes up during a micro-frame transmission (resp. inter micro-frame interval). We have:

$$\begin{cases} \mu_f(t,T) = U_{[0,f]}(t) + s + f + T \\ \mu_s(t,T) = U_{[0,s]}(t) + f + T \end{cases} \tag{4}$$

where $U_{[0,u]}(t)$ denotes a continuous uniform random variable in $[0,u]$.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2006 proceedings.

If the node misses the beginning of a micro-frame, then it cannot decode it. In this case, the node suspects it has just lost a micro-frame and then it keeps listening to the channel until it receives the subsequent micro-frame. Note that during channel sampling, a node must observe the channel idle for a duration of s at least to decide that there are no micro-frames being transmitted.

2) Gain from Avoidance of Irrelevant Preamble Reception: We estimate the rate of energy savings by avoiding to keep listening to the preamble by

$$1 - \frac{E[M(t,T)]}{E[P(t,T)]} = 1 - \frac{P_f E[\mu_f(t,T)] + P_s E[\mu_s(t,T)]}{E[U_{[0,m(s+f)]}(t)] + T}$$
$$= 1 - \frac{(s+f)/2 + f + T}{m(s+f)/2 + T}. \quad (5)$$

We notice that when the preamble is larger, the gain is also larger.

3) Combined Gain: To estimate the combined gain, resulting from avoiding to keep listening to the preamble and avoiding receiving irrelevant data, we introduce p, the probability that the expected data is irrelevant. Therefore, the time $C(t,T)$ a node spends listening to receive a data frame in MFP is

$$C(t,T) = (1-p)E[M(t,T)] + pE[M(t,0)]. \quad (6)$$

Then, the energy savings rate is

$$1 - \frac{C(t,T)}{P(t,T)} = 1 - \frac{(1-p)E[M(t,T)] + pE[M(t,0)]}{E[P(t,T)]} \quad (7)$$
$$= 1 - \frac{(s+f)/2 + f + (1-p)T}{m(s+f)/2 + T}.$$

For broadcast protocols such as flooding, the reception of only one message is enough and all subsequent messages are irrelevant. Therefore, if a node has n neighbors, then all the retransmissions of $n-1$ neighbors are redundant. In this case,

$$p = \frac{n-1}{n}. \quad (8)$$

Fig. 6 plots the percentage of energy savings according to Equation (8). We show that we achieve high energy saving rates: more than 83% of energy saved with check intervals from 50ms to 200ms. We use these values for the check interval, because it has been shown that they are the optimal for networks with less than 20 neighboring nodes [6]. Fig. 6 also shows that most of the energy saving is achieved by avoiding irrelevant reception of the preamble.

IV. Experiments

A. MC 13192 SARD

We have implemented MFP on the FreeScale MC 13192 SARD (Sensor Application Reference Design) Module. The module contains the MC13192, a short range low power radio transceiver with a 2.4 GHz 802.15.4 PHY modem. The transceiver is controlled by the HCS(08), a low power 8 bits-16MHz micro controller with 4KB of RAM and 60KB of flash memory. The radio and the micro controller communicate via a Serial Peripheral Interface (SPI). The module also features

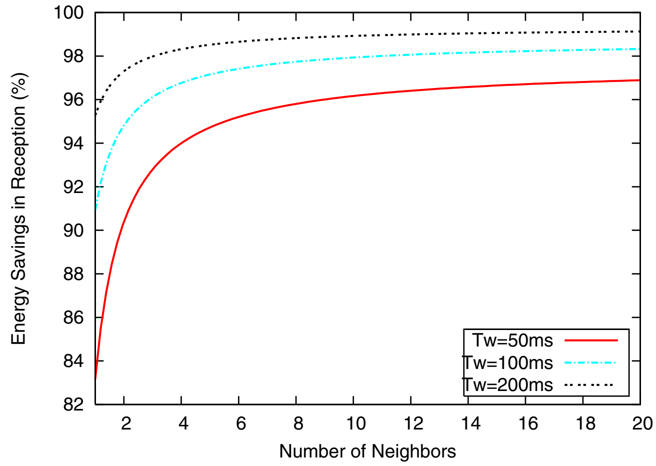

Fig. 6. Energy savings for different check intervals. The inter micro-frame interval s is 52μs. Transmission of a 14 byte micro-frame at 250 kb/s lasts $f = 448$ μs. Transmission of a 132 byte data frame at the same rate lasts $T = 4.224$ s.

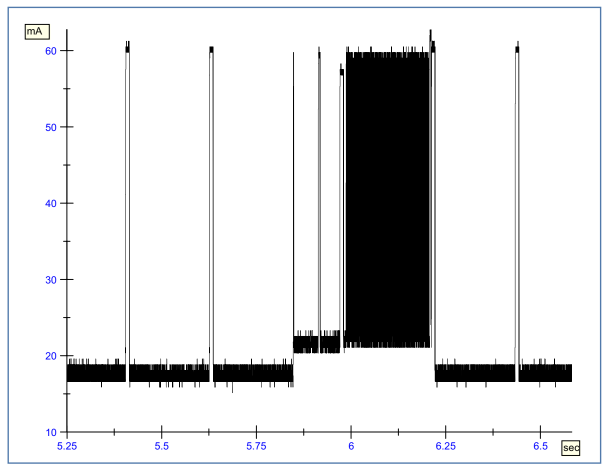

Fig. 7. Observing MFP operation by means of current consumption measurements.

accelerometers, an UART (to communicate over a serial line), 4 LEDs, and 4 buttons for code debugging and application demonstration.

B. Software Architecture

The micro controller of our module handles sequentially two concurrent tasks. The main program runs permanently at the application layer and calls primitives of the underlying layers; the interrupt handler runs every time a hardware interruption happens. In this case, the context of the main program is saved before the execution of the interrupt handler program and restored after the interrupt handler completes.

When a node wants to transmit a data packet it sends the packet to the radio, which in turn emits the packet over the air interface. The main program is blocked until the transmission is complete.

However, the reception of a packet is not blocking and the

TABLE I

CURRENT CONSUMPTION MEASUREMENTS CARRIED ON THE MC 13192 SARD

Radio Idle (not ready to receive)	0.5 mA
Radio Tx (Transmit)	39 mA (at +4 dbm)
Radio Rx (Receive)	39 mA
MCU (Active)	10 mA
MCU (Partially Active)	8 mA
LED	4 mA
Accelerometers sensors	3 mA

main program continues its execution after reception begins. When reception ends, it triggers an interrupt with a successful packet reception or time out (if it is not set to infinity). The interrupt handler program then calls the function that processes the reception.

C. Experiments

We have implemented a simple flooding application in which each node forwards the packet it receives once. The application is permanently waiting for packet reception according to the MFP periodic preamble sampling. We start the flooding sequence by pushing a button.

To make debugging of the code easier and quicker, we have measured the current consumption of the sensor node to identify what it is doing (transmitting, receiving, in a low power mode, etc). Table I shows the current drained for each operation.

Fig. 7 shows a local view of a flooding sequence, reception of frame and its retransmission by the node we are monitoring.

The first period, from 5.25s to 5.85s, shows the periodic preamble sampling operation of our MFP. Every check interval (200ms), the sensor goes to receive mode to sample the channel. During this first period, the channel was free. When it was in receive mode, the node drained 60mA corresponding to the current drained by the micro controller (10mA), embedded accelerometers (3mA), two LEDs (8mA, we used two LEDs for debugging), and the radio chip in receive mode (39mA). When the node leaves the receive mode, it goes to another mode that we call idle mode, as the radio is in idle[1] mode. In the idle mode, the node drains 17.5mA that corresponds to the current drained by the radio in the idle mode (0.5mA), the current drained by the micro controller (10mA), one LED (4mA) and accelerometers (3mA). Note that we lit 2 LEDs during the channel sampling mode and 1 LED during the idle mode for visual observation. The second period (zoomed in Fig. 8) starts with the reception of a micro-frame that the node has detected during its periodic sampling. There are 225 micro-frames in our preamble. When it receives a micro-frame, the MFP protocol of the node we are monitoring checks if the subsequent data frame is relevant. If so, based on the micro-frame sequence number provided by each micro-frame, the node estimates the time when the data frame will be transmitted and goes back to the idle mode to save energy.

[1]We chose to use the same terminology as in the MC13192 datasheet [9] where in the idle mode the radio in not ready to receive

Before sending a broadcast frame, the MAC layer picks a random backoff time to avoid collisions due simultaneous access to the channel. Note that, contrary to the IEEE 802.11 DCF, but like the IEEE 802.15.4 CAP [10], the channel is not continuously sensed during the backoff to save energy. When the backoff timer expires, the node performs a CCA (Clear Channel Assessment) and transmits directly, if the channel is sensed free for a duration equal to the inter micro-frames interval. Otherwise, if the channel is busy, the node re-executes the backoff procedure. Fig. 9 zooms in on the backoff procedure that occurred from 5.92s to 5.97s. The channel was sensed free from $5,97$s to $5,98$s and the node began the transmission by sending its series of micro-frames right afterward. Note that the current consumption during the carrier sense (5.97s - 5.98s) is slightly less than the one during data reception (5.9s - 5.925s). This is because, the micro-controller in our implementation goes into the power save mode during the carrier sense to save energy (2 mA are saved).

We carried out several experiments in order to obtain the right number of micro-frames needed to take over the channel for the check interval (200ms). We needed to do this by experiments, because we have found that the duration of the inter micro-frame interval is variable and depends on the micro-frame length. In general, the inter micro-frames interval increases with the size of a micro-frame, because we are transmitting micro-frames by software. The inter micro-frame duration includes the time needed by the micro-controller to initialize each micro-frame (e.g. setting the sequence number) and the time needed to transfer the micro-frames from the RAM to Radio Module via SPI (Serial Peripheral Interface). For the sensor node module we are using, the MC13192 SARD, this time is about 2ms[2]. As a consequence of this, the time needed by a receiver to make a decision whether there is a preamble transmission on the channel or not must be at least as large as 2ms. This very large time increases the amounts of idle listening, decreasing hence the energy savings of MFP.

We argue that this drawback is hardware and radio architecture dependent and is not an inherent weakness of the MFP. For an efficient implementation of MFP, we imagine next generation radio transceivers transmit micro-frames using a dedicated hardware to reduce the inter micro-frames interval. With such equipment, the MFP MAC agent just sends the frame to the PHY agent without worrying about micro-frames, since it is up to the PHY agent to send them.

V. CONCLUSION

This paper introduces a Micro-Frame Preamble MAC that aims at improving energy savings of traditional preamble-sampling techniques by reducing the amount of irrelevant receptions. In the paper, we have shown that switching the radio off during an irrelevant reception gives considerable

[2]This is the largest inter micro-frame interval. It is placed between the last micro-frame transmission and the data transmission. This interval is greater than the others because it depends on the data frame size, which is greater than a micro-frame size; see Fig. 10.

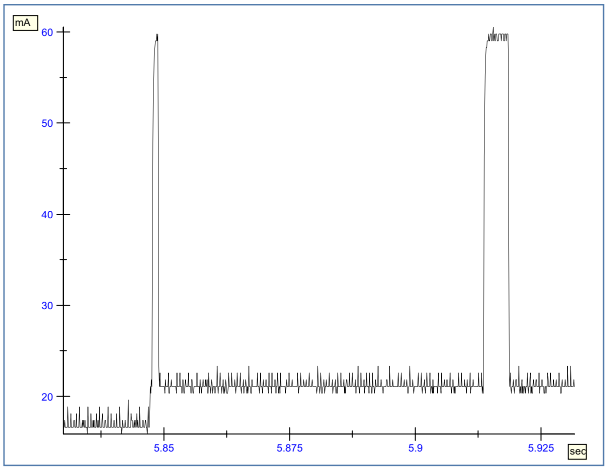

Fig. 8. Reception of a broadcast frame.

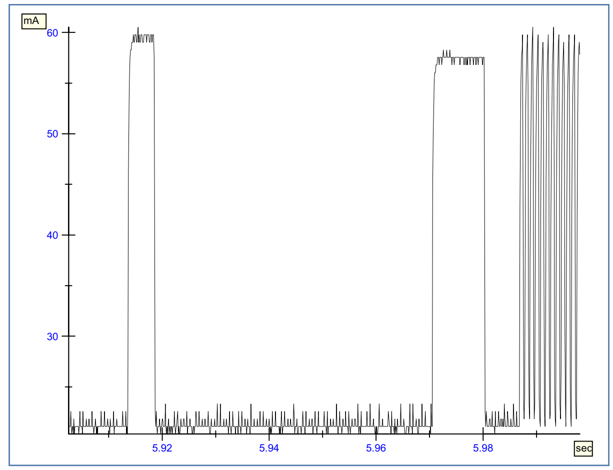

Fig. 9. Random backoff before transmission.

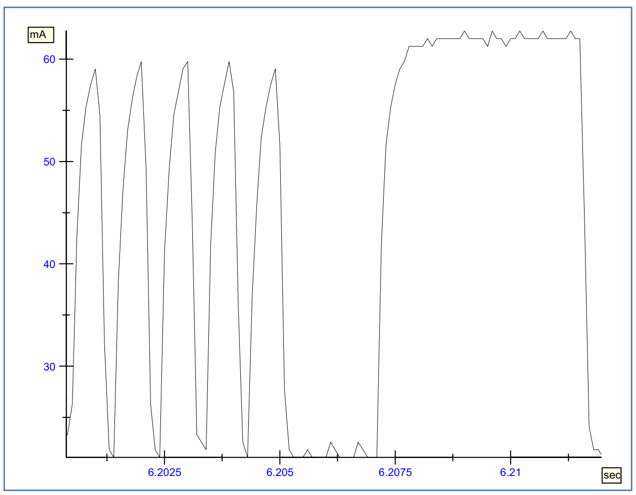

Fig. 10. Transmission of micro-frames and a data frame.

energy savings. We have carried out experiments with the MC 13192 SARD equipment essentially designed to implement IEEE 802.15.4 for Zigbee applications. The experiments show the need for radio modules that take into account MFP characteristics to obtain even better energy savings.

VI. Acknowledgments

Special thanks to Laurent Granger for the implementation work on MC13192 and to Philippe Jayet for the measurements of energy consumption.

References

[1] W. Ye, J. Heidemann, and D. Estrin. An energy-efficient MAC protocol for wireless sensor networks. *Proceedings of the IEEE Infocom*, pages 1567–76, New York, NY, July 2002.

[2] K. Langendoen and G. Halkes. Energy-Efficient Medium Access Control. *Book chapter in the Embedded Systems Handbook, R. Zurawski (editor), CRC press*, 2005.

[3] I. Demirkol, C. Ersoy, and F. Alagoz. MAC Protocols for Wireless Sensor Networks: a Survey. *IEEE Communications Magazine, to appear*, 2005.

[4] T. van Dam and K. Langendoen. An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. *Proceedings of the ACM Sensys*, pages 171–80, Los Angeles, CA, November 2003.

[5] C. Enz, A. El-Hoiydi, J. Decotignie, V. Peiris. WiseNET: An Ultralow-Power Wireless Sensor Network Solution. *IEEE Computer*, 37(8):62–70, August 2004.

[6] J. Polastre, J. Hill and D. Culler. Versatile Low Power Media Access for Wireless Sensor Networks. *Proceedings of the ACM SenSys*, 2004.

[7] A. El-Hoiydi and J. Decotignie and J. Hernandez. Low Power MAC Protocols for Infrastructure Wireless Sensor Networks . *Proceedings of European Wireless*, Feb. 2004.

[8] W. Ye, J. Heidemann, D. Estrin. Medium access control with coordinated, adaptive sleeping for wireless sensor networks. *IEEE Trans. on Networking*, 12(3):493–506, June 2004.

[9] Freescale Semiconductor. MC13192/MC13193 2.4 GHz Low Power Transceiver for the IEEE 802.15.4 Standard. *Rev. 2.8*, 04/2005.

[10] IEEE 802.15.4. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). 2003.