Efficient synchronization technique for non-coherent IR-UWB receiver targeting IEEE 802.15.6 wireless BAN

Houcine Chougrani[†*], Jean Schwoerer[†], Pierre-Henri Horrein[*], Amer Baghdadi[*]
[†]Orange Labs, 28 Chemin du Vieux Chêne, 38240 Meylan, France
[*]Institut Mines-Telecom; Telecom Bretagne; Lab-STICC, Technopôle Brest-Iroise, 29238 Brest, France
{houcine.chougrani,jean.schwoerer}@orange.com, {ph.horrein,amer.baghdadi}@telecom-bretagne.eu

ABSTRACT

The introduction in 2012 of the IEEE 802.15.6 standard for wireless body area networks (BAN) stimulates the research of practical and efficient receiver implementations. One of the main challenges in this context deals with the signal synchronization for the impulse radio ultra wideband (IR-UWB) physical layer. Targeting non-coherent detectors, which are known to offer significant energy-per-bit savings over their coherent counterparts, this paper presents a novel standard-compliant synchronization technique. The proposed technique is based on inter-pulse time interval detection and comparison. The associated performance has been evaluated using relevant channel models specified in the IEEE 802.15.6 standard. Promising results are demonstrated in terms of number of synchronization success rate and time.

1. INTRODUCTION

The last few years have seen significant progress in wireless body area networks (BAN) in terms of both technology and applications. One of the major recent results in this context is the publication in 2012 of the IEEE 802.15.6 standard [1]. It is the first international communication standard for short-range wireless devices for in-body, on-body, and around-the-body communications. The standard does not specify or exclude any application which can be either medical or non-medical belonging to a variety of fields including personal audio or video, gaming, entertainment, wearable computing, ambient intelligence, and many others.

The IEEE 802.15.6 standard defines Physical (PHY) and Medium Access Control (MAC) Layers [1]. Three PHY are proposed: (1) body channel communication for signal propagation on the skin surface, (2) narrowband PHY mainly for healthcare applications using the various available license free band, and (3) ultra wideband (UWB) PHY able to address higher data rates. In this paper we focus on the impulse radio ultra wideband (IR-UWB) PHY, and more particularly on the synchronization issue at the receiver. Achieving an accurate synchronization is a major challenge in IR-UWB systems and a key factor to ensure reliable communications. Even a slight misalignment in the order of nanoseconds can severely degrade the system performance [2, 3, 4]. A recent overview of existing synchronization algorithms for IR-UWB systems is available here [4]. This overview paper analyses in particular the performance of few relevant recent synchronization algorithms: correlation based timing acquisition proposed in [5], orthogonal code matching based method in [6] and energy detection based method presented in [7].

However, in the case of the IEEE 802.15.6 standard, the frame structure for the UWB PHY incorporates a synchronization header (SHR). The specified header presents a particular structure based on the use of 63-bit Kasami sequences. This choice is due to the good cross correlation properties between those sequences(coexistence of BANs), and good autocorrelation properties for accurate synchronization. Therefore, this frame structure offer opportunities for new efficient synchronization schemes, and this constitutes the main motivation behind the work presented in this paper.

We consider in this work non-coherent receivers, which are known to offer significant energy-per-bit savings over their coherent counterparts. In this context, the main contribution of the paper is the proposal of a novel IEEE 802.15.6 standard-compliant synchronization technique for IR-UWB PHY. The proposed technique is based on the detection of the sequence of time intervals between pulses in the SHR.

The rest of the paper is organized as follows. Section 2 gives a brief description of the UWB PHY frame structure and the non-coherent receiver architecture. Section 3 describes in details the proposed synchronization algorithm and Section 4 presents and discusses the obtained results using additive white Gaussian noise (AWGN) and IEEE 802.15.6 channel models. Finally, conclusions are drawn in Section 5.

2. SYSTEM MODEL

This section gives a brief description of the synchronization header and the non-coherent receiver architecture considered in this work.

2.1 Synchronization header

The UWB PHY frame format specified in the IEEE 802.15.6 standard [1] is illustrates in Figure 1. It consists of a Synchronization header (SHR), a Physical-layer Header (PHR), and the Physical-layer Service Data Unit (PSDU) which represents the payload of the frame. The SHR consists of the preamble, which is used for timing synchronization, packet detection, and carrier frequency offset recovery, and the start-of-frame delimiter (SFD), which is used for frame synchronization.

The preamble is constructed based on a Kasami sequence of length 63. The standard defines eight different Kasami sequences which are indexed by C_i for $i = 1, \ldots, 8$. The

BODYNETS 2013, September 30-October 02, Boston, United States

DOI 10.4108/icst.bodynets.2013.253929

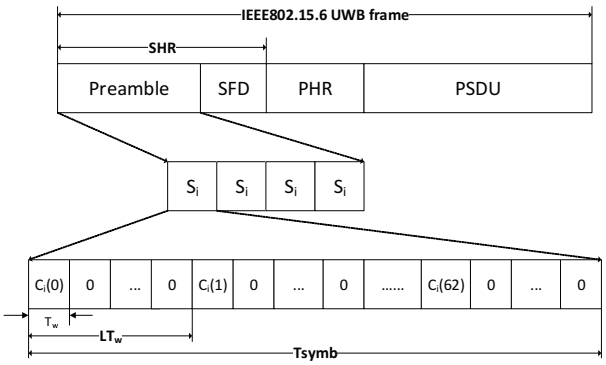


Figure 1: UWB PHY frame format

preamble consists of 4 repetitions of the symbol S_i. Such symbol is obtained by a Kasami sequence zero-padded by $L-1$ zeros. Figure 1 illustrates the construction of the symbol S_i, where the zero-padding period is LT_w and T_w is the pulse waveform duration. The standard specifies a duty cycle of 3% to be employed for the transmission of the synchronization symbol S_i for IR-UWB. Hence, the values of T_w and L depend on the modulation employed.

The SFD consists of the inversion of the Kasami sequence ($0\rightarrow1$ and $1\rightarrow0$) used in the preamble. This choice allows to minimize the correlation between the two fields in order to improve the detection accuracy of the SFD. As this paper focuses on the challenging task of timing synchronization, only the preamble field will be concerned.

2.2 Non-coherent receiver structure

A low complexity non-coherent receiver based on energy detection over a short integration period is considered in this work. Figure 2 illustrates the structure of this receiver which was proposed in [8]. This structure embeds a low pass filter with short integration duration in the order of the pulse duration. The goal is to obtain the envelop of the received signal as a baseband pulse. This envelop is then compared to a predefined threshold in order to determine whether a pulse is being received or not. The result of this comparison is then used as an input to the synchronization process. It should be noted that using a comparator avoids the need for an analog-to-digital converter (ADC), which significantly decreases the complexity and the cost of the receiver. Consequently, such a low complexity structure allows simply to detect the presence/absence of the signal rather than a more accurate amplitude information (energy estimation).

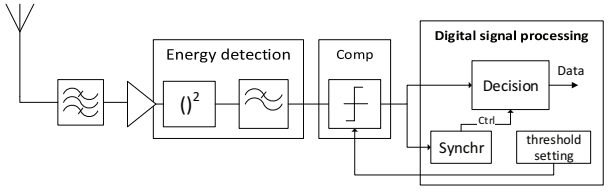


Figure 2: Block diagram of the non-coherent receiver

3. PROPOSED ALGORITHM

In this section, the proposed non-coherent synchronization algorithm is described for 802.15.6 IR-UWB standard compliant systems.

3.1 General principles

The algorithm is based on the time distance between pulses in the synchronization symbol. In order to illustrate the algorithm and its principles, the fourth Kasami sequence (C_4) will be used. This sequence can be represented using its binary values, represented in the upper part of Figure 3. When this sequence is sent, a pulse will be transmitted in the first slot, then 3 slots will stay unused, before having a new slot with a pulse. On the receiver side, this means that there will be a time interval of four between the first two pulses. This sequence can thus be represented using a time interval representation, as presented in the lower part of Figure 3

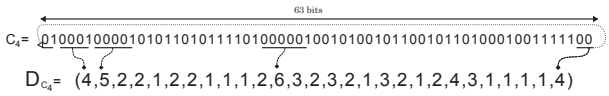


Figure 3: Representations for Kasami sequence C_4

The proposed algorithm is based on these time intervals. The use of time spaces has first been described for a specifically designed synchronization symbol [8]. This symbol had no repeating time intervals. However, in the 802.15.6 case, the Kasami sequence has many distances which are not unique. The new algorithm proposed in this work is designed to process any sequence.

3.2 Detailed description

The algorithm is based on a state machine, which will be used to check the correlation between the received symbol and the expected one. It also uses a time distance computation. When a pulse is received, a time counter is reset, and its value when the next pulse is received gives the time interval between the two pulses. This is possible because transmission is slotted in the 802.15.6 IR-UWB PHY, and a slot duration without enough energy received means no pulse has been received.

Algorithm 1 describes the transition function of the state machine used for the algorithm. Line numbers given as reference from now refers to Algorithm 1. State transitions occur when a pulse is received. The aim of this state machine is to detect a correlation between the sequence being received and the expected sequence (in this case, the fourth Kasami sequence). Three different state types are defined. The idle state, named $State_0$ (line 4) in Alg. 1, represents the case when no correlation has been detected. For example, if two pulses with distance $\{25\}$ are received, they obviously do not belong to the sequence.

The correlated state, named $State_{corr}$ (line 25, represents the case when the received sequence is currently correlated with the expected sequence. The position in the expected sequence, defined as the distance index in the sequence, is stored in the "position" variable. The previously detected positions are also stored, in the "detected_impulse" vector. When the machine is in state $State_{corr}$, with position j and the new computed distance matches distance $j+1$ (line 26), the position is updated to $j+1$, and the "detected_impulse" vector is updated. If it does not match (line 30), the machine goes back to idle state $State_0$, since the sequence being received does not match the expected sequence. When all positions have been detected, the "detected_impulse" vector is filled with 1 in the algorithm, and synchronization is acquired (line 35).

The last state type is the "intermediate state" or $State_{int}$

Algorithm 1 Transition function for the state machine

```
1: procedure TRANSITION
2:     compute distance with previous pulse
3:     switch state do
4:         case State_0
5:             if distance == d_j then
6:                 if d_j ∈ repetitive distances then
7:                     state ← State_Int
8:                 else
9:                     state ← State_corr
10:                    position ← j
11:                    detected_impulse(j) ← 1
12:                end if
13:            end if
14:        case State_Int
15:            if one possible subsequence remaining then
16:                state ← State_corr
17:                position ← j
18:                detected_impulse(j) ← 1
19:            else if multiple subsequences then
20:                state ← State_Int
21:            else
22:                state ← State_0
23:                detected_impulse ← (0, ..., 0)
24:            end if
25:        case State_corr
26:            if distance == d_{j+1} then
27:                position ← j + 1
28:                state ← State_corr
29:                detected_impulse(j + 1)← 1
30:            else
31:                state ← State_0
32:                detected_impulse ← (0, ..., 0)
33:            end if
34:    end switch
35:    if detected_impulse == (1, ..., 1) then
36:        synchronization acquired
37:    end if
38: end procedure
```

(line 14), which represents cases when the correlation with the sequence can not be immediately decided. This occurs when a distance is received which can represent more than one position in the expected sequence. These distance are called repetitive distances in the algorithm. In this case, the machine stays in this temporary state as long as no decision can be taken. For example, if the algorithm finds a distance 3, there are many possible combinations ($\{3,2,3\}, \{3,2,1,3\}, \{3,2,1,2\}, \{3,1\}$), and the position can not be decided. When the next distance is received, if it is $\{2\}$, the case of $\{3,1\}$ is discarded, but the sequence may still match the expected one (line 19). It it is a $\{1\}$, then only the $\{3,1\}$ sequence match, and the state becomes $State_{corr}$ (line 15). Positions are updated accordingly. If it is neither a $\{2\}$ nor a $\{1\}$, no correlation has been found and the algorithm goes back to state $State_0$ (line 21).

In this simple version, the algorithm is very sensitive to missed or spurious pulses. Optimizations have been made to better take into account these errors. For the sake of clarity, these optimizations have not been represeented in algorithm 1. The main idea is that comparisons to decide whether a distance matches the expected on is less strict, and can be made with sum of expected distances (missed pulse) or sum of received distances (spurious pulse)/ For example, if a distance $\{6\}$ is received, expected next distance is a $\{3\}$. However, if the next pulse is missed, the next distance might be a $\{5\}$, which could be the sum of $\{3\}$ and $\{2\}$, the two distances following $\{6\}$. If a $\{5\}$ is received, the algorithm will not stop, and decide according to the next sequence. On the opposite, the two next received distances could $\{1\}$ and $\{2\}$, which should be discarded. However, the sum of received distances is 3, and this error could be caused by noise. This sequence can be kept if the following distances match.

4. SIMULATION RESULTS

In this section, the simulation results for the proposed algorithm are presented.

In order to validate its performance, the proposed algorithm was implemented in Matlab, and integrated in a complete UWB BAN simulation environment developed for the RUBY project[1]. This environment implements the different IEEE 802.15.6 channel models, as well as The algorithm has been validated in different environments: an AWGN channel, a IEEE 802.15.6 CM3 channel [9], which represents a body surface to body surface communication, and a IEEE 802.15.6 CM4 channel [9], representing a communication between the body surface and an external device. This last channel model is divided in 4 subcases according to the body inclination (0°, 90°, 180°and 270°)

The following parameters were used for each channel model:

- bandwith: $B = 1.8GHz$,
- pulse duration: $T_w = 600ps$, slot duration: $2ns$,
- synchronization symbol: fourth Kasami sequence C_4 (*cf* Section 3),

The different curves have been obtained for an average of 200 different channel realizations. For each channel model, the studied performance was the synchronization success rate, and the time before synchronization.

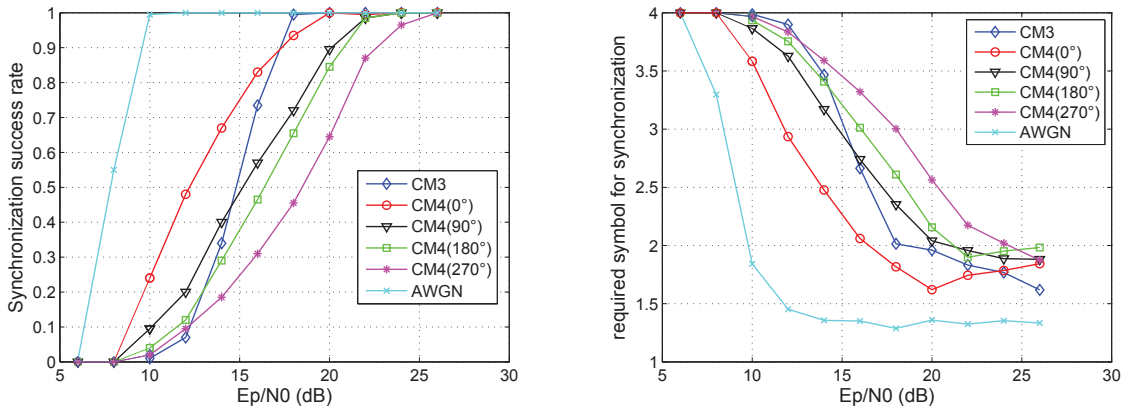
Performance results are presented in Figures 4(a) and 4(b). As can be seen, the difference between all channels is not negligible. Technical requirements for the IEEE 802.15.6 IR-UWB BAN targets a 10% Packet Error Rate (PER). This rate is dependant on the complete receiver, however, in order to avoid problems due to synchronization, success rate must be at least 90%. In the AWGN channel, 90% success rate is obtained when pulse energy to noise ratio (Ep/N0) is around 8 dB, and a 100% success rate is obtained when Ep/N0 reaches 10 dB.

Results for the 802.15.6 channel models are more complex. As can excepted, results are degraded when compared to AWGN, which can be explained by the multipath characteristic of the CM3/CM4 channel models. This may decrease available energy in a single path. The worst cases are CM4 for angles other than 0°. CM4 (0°) is a Line-Of-Sight (LOS) channel, with a strong main path. CM3 may be either LOS or Non-LOS (NLOS), depending on whether a part of the body is located between devices. CM4 for angles other than 0°are either LOS or NLOS, depending on the position of the body. However, the distance between devices is much higher than for CM3, and the spreading of paths over time is more important, which explains why results are lower than CM3.

If we target a 100% synchronization success rate, and compare the results to the proposition in [10], we obtain the results in Table 1. The first column gives the required $Ep/N0$ for the 10% PER in [10], while the second and third columns gives required $Ep/N0$ to obtain respectively a 100% and a 90% synchronization success rate with the proposed algorithm. This comparison does not aim at deciding whether results are better or worse. They aim at deciding whether synchronization results are sufficient for operation with 802.15.6 compliant devices.

Before discussing the results, it is worth to note that the results for [10] are given for a data rate of 1 Mbps for CM4 with angles diffrent from 0°, while the results of the proposed method are given for 12 Mbps for all channels. Furthermore, the modulation used in [10] may use two pulses per symbol. This increases the energy per symbol and thus decreases the number of missed pulses. However, we can see that the proposed algorithm performs well for AWGN, CM3

[1]Research project started in 2012 and funded in part by the French National Research Agency (ANR)



(a) Synchronization success rate

(b) Time before synchronization

Figure 4: Simulation results of the proposed algorithm for various channel models

and CM4 (0°), with a margin between 0 and 3 dB. This means that the synchronization is fully acquired between 0 and 3 dB before the 10% PER is reached in [10]. We can notice a degradation of 2 dB for CM4 (90°) and CM4 (180°), and 3 dB for CM4 (270°). This can be partly explained by the difference in modulation and in data rate for these channel models. However, it is interesting to notice that the proposed algorithm is based on a simple state machine, and on a comparator for pulse detection, which can be efficiently implemented in hardware. This may lead to significant gains in area, cost, and energy consumption.

Table 1: Results comparison

Channel model	[10] (10% PER)	This work (100% sync.)	This work (90% sync.)
AWGN	13 dB	10 dB	
CM3	20 dB	18 dB	17 dB
CM4 (0°)	18 dB	18 dB	17 dB
CM4 (90°)	18 dB	22 dB	20 dB
CM4 (180°)	19 dB	22 dB	21 dB
CM4 (270°)	20 dB	26 dB	23 dB

To conclude on the results, Figure 4(b) presents the number of symbols required to acquire the synchronization. When comparing with the success rate results, we can see that less than 2 symbols are required to reach 100% synchronization rate. This leaves enough time for further processing to improve the received signal quality.

5. CONCLUSION

In this paper, an efficient synchronization algorithm for the IEEE 802.15.6 UWB PHY has been presented. The proposed algorithm can be integrated in a low complexity non-coherent receiver. It is based on a dedicated state machine, which detects the sequence of time intervals between pulses and compares it to the synchronization symbol based on a Kasami sequence. This algorithm was implemented in a complete communication system environment, and simulations were performed for AWGN and for CM3/CM4 channel models which are defined by the standard. Comparisons with relevant state-of-the-art work illustrate the effectiveness of the proposed technique. As an example, for the CM3 channel model, the algorithm reaches 100% success rate with a 3 dB margin. Furthermore, its low complexity offers good opportunities for integration in low power BAN devices.

6. REFERENCES

[1] Part 15.6: Wireless Body Area Networks, 29 February 2012.

[2] Zhi Tian and G.B. Giannakis. BER sensitivity to mistiming in ultra-wideband impulse Radios-part I: nonrandom channels. *IEEE Transactions on Signal Processing*, 53(4):1550–1560, 2005.

[3] Ning He and C. Tepedelenlioglu. Performance analysis of non-coherent UWB receivers at different synchronization levels. *IEEE Transactions on Wireless Communications*, 5(6):1266–1273, 2006.

[4] R. Akbar and E. Radoi. An overview of synchronization algorithms for IR-UWB systems. In *Proc. of the International Conference on Computing, Networking and Communications (ICNC)*, pages 573–577, 2012.

[5] Liuqing Yang and G.B. Giannakis. Timing ultra-wideband signals with dirty templates. *IEEE Transactions on Communications*, 53(11):1952–1963, 2005.

[6] Yeqiu Ying, M. Ghogho, and A. Swami. Code-Assisted Synchronization for UWB-IR Systems: Algorithms and Analysis. *IEEE Transactions on Signal Processing*, 56(10):5169–5180, 2008.

[7] Xiliang Luo and G.B. Giannakis. Low-complexity blind synchronization and demodulation for (ultra-)wideband multi-user ad hoc access. *IEEE Transactions on Wireless Communications*, 5(7):1930–1941, 2006.

[8] Benoît Miscopein and Jean Schwoerer. Low complexity synchronization algorithm for non-coherent uwb-ir receivers. In *Vehicular Technology Conference*, pages 2344 – 2348, 22-25 April 2007.

[9] Kamya Yekeh Yazdandoost and Kamran Sayrafian-Pour. Channel model for body area network (ban). *IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs)*, Document IEEE802.15-08-0780-05-0006, 2009.

[10] Kiran Bynam et al. Etri & samsung phy proposal to 802.15.6. Technical report, IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs), May 2009.