A BLIND TIMING ACQUISITION ALGORITHM FOR DS-UWB SYSTEMS

Zhila Amini Sheshdeh Abdolreza Nabavi
e-mail: zh.amini@gmail.com *e-mail: abdoln@modares.ac.ir*
Faculty of Engineering, Tarbiat Modares University, Tehran, Iran

Key words: Ds-UWB, Blind, Acquisition

ABSTRACT

This paper develops a blind timing acquisition algorithm for DS-UWB signals, using simple integrate-and-dump operations over the symbol duration. The performance of the algorithm is simulated for single user and multipath channel model. The results illustrate a higher speed and lower complexity in comparison with existing blind methods. The algorithm is designed at hardware level with VHDL, and implemented on an XC2VP7-6FF896-FPGA. The power consumption is 473mW, the speed is 120 MHz, and the BER is about 0.01 in multipath channel.

I. INTRODUCTION

The interest for commercial UWB applications is growing fast especially in the area of short-range indoor wireless communications [1]. One of the most critical challenges in enabling the unique benefits of UWB transmissions is the clock synchronization [2]. The difficulty of this in UWB is because the information bearing waveforms are impulse-like and have low amplitude. Peak-picking the output of a sliding correlator between the received signal and the template is sub-optimum in the presence of dense multipath. Also, it results in unacceptably slow acquisition speed, and has prohibitive complexity when one has to perform exhaustive search over thousands of bins (chips) [1].

A blind timing acquisition algorithm was developed in [1, 3] for frame-level synchronization in TH-UWB system. The method relies on simple integrate-and-dump operations over one symbol duration, and the synchronization time is acquired using the maximum value of the integrations.

In this paper, we present a simple algorithm for rapid timing acquisition in UWB systems. The algorithm exploits the rich multipath diversity present in UWB channels. The synchronization is achieved when cross-correlation exceeds a certain threshold level, determined based on SNR and the channel performance. This reduces the design complexity, and decreases the mean detection time for an acceptable BER. The algorithm is simulated in MATLAB, designed with VHDL and tested on a FPGA.

The reminder of the paper is organized as follows. Section II describes the proposed algorithm. Section III presents the implementation in VHDL and FPGA. Section IV reports the simulation results. Finally, Section V provides the concluding remarks.

II. BLIND TIMING ACQUISITION

In UWB radio, every information symbol is conveyed by N_f data modulated ultra short pulses p(t), each over one frame of duration T_f. Thus, the symbol duration is [3]: $T_s=N_fT_f$ seconds.

With p(t) having duration $T_p(<<T_f)$ at the sub-nanosecond scale, the transmitted UWB signal occupies a bandwidth of B_s~$1/T_p$ [1]. UWB radio generally adopts modulation methods such as pulse position modulation (PPM), pulse amplitude modulation (PAM), and binary phase shift keying (BPSK). In this paper, we employ BPSK [3].

The transmitted UWB signal which consists of a train of short pulses (monocycles) may be dithered by a time hopping (TH) sequence to facilitate multiple-access and to smooth the spectrum. Also, the polarities of the transmitted pulses may be randomized using a direct sequence (DS) spreading code to mitigate multiple access interference (MAI) [3].

The DS-UWB signal transmitted during the acquisition process for single user in duration of T_s can be expressed as [2]:

$$P_T(t) = \sum_{n_f=0}^{N_f-1} \alpha_{[n_f/N_{ds}]} p(t - n_f T_f) \qquad (1)$$

That $\alpha_{[n_f/N_{ds}]}$ is the randomized coefficient that is different (+1 or -1) for each frame.

The multipath channel is modeled as a tapped-delay line, with L+1 taps $\{\alpha_l\}_{l=0}^{L}$ and delays $\{\tau_l\}_{l=0}^{L}$ satisfying $\tau_l < \tau_{l+1}$. Being quasi-static, the channel coefficients and delays remain invariant over one transmission burst, but are allowed to change across bursts.

To isolate the multipath spreading effects from the propagation delay τ_0, all path delays can be uniquely casted into $\tau_{l,0} := \tau_l - \tau_0$.

Focusing on a single user link, and treating multiuser interference (MUI) as noise, the waveform arriving at the receiver is given by:

$$r(t)=\sqrt{\varepsilon}\sum_{l=0}^{L}\alpha_l\sum_{k=0}^{+\infty}s(k)p_T(t-kT_s-\tau_{l,0}-t_2)+w(t),$$

Where the noise term w(t) includes MUI, and the first arrival time t_2 is the transmission starting time t_1 augmented by the propagation delay τ_0 [3].

The first step of our blind timing acquisition algorithm takes from the received waveform a segment of duration T_s, starting at time (t_3+ nT_f + mT_s), for integers n $\in$ [0, N_f), and m $\in$[0, M-1], with MT_s being the observation interval [3].

Denoted by $x_{n,m}(t)$, this waveform can be expressed as:

$$x_{n,m}(t)=r(t+mT_s+nT_f+t_3), t\in[0,T_s) \qquad (2)$$

The proposed r(t) is received signal containing several $P_T(t)$ with different coefficient. The blind algorithm based on dirty template can be carried out as follows:

Step 0. Set n=0.

Step 1. For a given n, take M segment $x_{n,m}(t)$ each of duration T_s from the received signal as in (2). Integrate-and-dump the product of adjacent segments $x_{n,m}(t)$ and $x_{n,m+1}(t)$ as below:

$$R_{xx}(n;m):=\int_0^{T_s}x_{n,m}(t)x_{n,m+1}(t)dt \qquad (3)$$

Step 2. Form an estimate of R_{xx} (n) by averaging over all pairs of the absolute values of the integral obtained in Step 1. In this way:

$$\hat{R}_{xx}(n)=\frac{2}{M}\sum_{m=0}^{M/2-1}\left|\int_0^{T_s}x_{n,2m}(t)x_{n,2m+1}(t)dt\right|. \qquad (4)$$

Now we compare $\hat{R}_{xx}(n)$ with a threshold level, determined using SNR and the channel model. If it is more than threshold, the parameter "n" is the estimation for synchronization time. Otherwise, we increase n up to N_f − 1. Note that in [1] and [3], R_{xx} is calculated for all possible n's to obtain the maximum, which may give a better BER and resolution in synchronization. However, by using a proper threshold level, we can decrease the time of detection whereas BER is maintained at an acceptable level.

Thus $\hat{n}_{\in}=\arg_n\{\hat{R}_{xx}(n)>\mathrm{V}th\}$, and $\hat{n}_{\in}T_f$ is an estimate of a symbol starting time. By applying this time shift, receiver will get synchronized.

The threshold level is obtained empirically. One way is to monitor the correlator output for all possible phases, and then set the threshold value so that only some of the strongest correlator output exceeds the threshold. Obviously, a higher threshold results in lower error but higher mean detection time.

Note that instead of having a fixed increment (frame duration) per iteration, our algorithm can be employed with variable non-integer increments using voltage controlled clock (VCC) circuits. The latter then enables both the acquisition and tracking, with the possibility of further reduction of synchronization speed.

In this paper, we develop the algorithm for DS-UWB system with single user and multipath channel at simulation level and also on FPGA. In [3], a similar algorithm has been simulated for a single user TH-UWB system.

III. IMPLEMENTATION

In this section we design the proposed algorithm in VHDL. We have designed two blocks buffer and synchronization and a package as explained below.

The first block is a buffer (buff) sensitive to positive edge of the clock. When the start of preamble bits is indicated by the respective block, e.g., the signal "preamble_start" is high, the buffer starts saving the rest of preamble bits.

```
if(clk='1' and clk'event) then
  if (rst='1')then
        buffer_full                         <='0';
        i                                   :=0;
 else
    if(preamble_start='1' and finish='0')then
        i:=i+1;
        output(i):=sample;
        buffer_full        <='0';
        if (i=preamble_samplenum)then
                buffer_full<='1';
                buffered_sample<=output;
                i:=0;
        end if;
   else
        i:=0;
            buffer_full<='0';
end if;
end if;
end if;
```

This buffer save M bit of preamble that the transmitter send them. When the number of saved sampled in the buffer reaches number of sample per M symbol of preamble, buffer gets full and sets an alarm signal, referred to as "buffer_full" to high and then synchronization block starts operation as below. This block performs this blind algorithm on saved data.

```
if(buffer_full='1') then
xnm:=0;
for m in 0 to(pair-2) loop
index:=(m*2*bit_samplenum)+(symbol_samplenum*n)+
1;
index2:=index+bit_samplenum;

for i in 1 to bit_samplenum loop
        a(i):=buffered_sample(index+i-1);
        b(i):=buffered_sample(index2+i-1);
end loop;
xnm:=abs(correlate(a,b))+xnm;
end loop;
```

```
      rxx(n+1):=xnm;
      if(rxx(n+1)> vth)then
            lock_time <=(n*symbol_samplenum);
            finish     <='1';
            sync_acheived     <='1';
            ok                :='1';
      else
            n          :=n+1;
            sync_acheived     <='0';
            if(n>=nf)then
                  ok:='1';
            else
                  ok:='0';
            end if;
      end if;
else
            lock_time         <=0;
            finish            <='0';
            sync_acheived     <='0';
            n          :=0;
            ok         :='0';
end if;
```

It set a signal when the acquisition algorithm get finished and also save the achieved "n".
There is a main block that connects the above two blocks to each other properly.

```
u1: buff port  map(clk,  rst,  preamble_start,    sample,
ending, full,data);
u2: sync port   map   (full   ,rst   ,data,   sync_acheived,
shift,ending);
```

Beside these blocks there is a package that we define the parameters, function in that.

```
use work.blind_method_package.all;
```

For test of the VHDL module and because there is no model for multipath channel in VHDL, we send the transmitted data to multipath channel in simulink and then use the received data in our VHDL code.

IV. SIMULATION RESULTS

The performance of acquisition algorithmic is simulated in MATLAB. The resolution is coarse since the operation is at the frame level. As a result, synchronization is achieved within a reasonable amount of accuracy in a short time. For a better BER, a fine tracking scheme is needed after coarse acquisition.
The simulation is performed for both the multipath and AWGN channel models. In the first, interfaces are present, and the mean detection time is longer.
Fig. 1 shows the BER of this new blind algorithm in proposed systems.

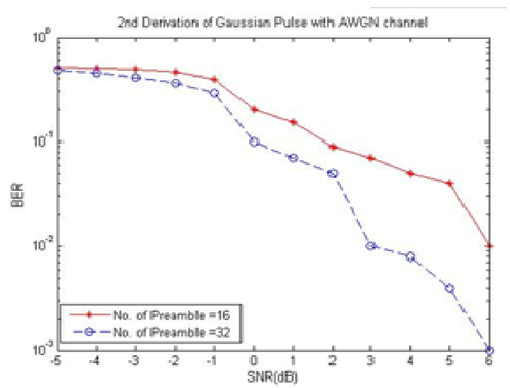

Figure 1. BER of Blind method with two different values of pilot pulses.

To determine a proper value for threshold V_{th}, we calculate R_{xx} of correct synchronous time (or parameter "n"). Since R_{xx} is a function of the channel model and SNR, V_{th} is changed accordingly.
The above algorithm is implemented on a FPGA (XC2VP7-6FF896 from Xilinx) by use of VHDL coding. The performance metrics are reported in Table 1.

Table 1. Performance

Parameter	Value
N_f	8
T_s	12 nsec
T_f	1.5 nsec
T_p	1 nsec
Number of "preamble"	40
Number of "main_data"	900
SNR	5dB
Sampling rate	4GHz
BER multipath channel	0.01
BER AWGN channel	0
Minimum period	8.2982ns
FPGA implement max frequency	120MHz
Total power consumed	473 mW

V. CONCLUSION

In this paper, we developed a simple training scheme for rapid timing acquisition in DS-UWB systems. Our training sequence enables a cross-correlation pattern among received waveforms, which exploits the rich multipath diversity present in UWB channels.

Relying on integrate-and dump operations per symbol, the algorithm reduces the complexity and markedly improves the acquisition speed. The speed improvement is mainly due to using the threshold level instead of computing the maximum value of integration [1, 3].

Implementing this algorithm on FPGA illustrates a power consumption of 473 mW, a maximum speed of 120 MHz, and a BER about 0.01 in multipath channel.

ACKNOWLEDGEMENT

This study has been financed by the Iranian Telecommunication Research Center (ITRC). Authors would like to thank the sponsors for their interest and support.

REFERENCES

1. Liuqing Yang and Georgios B. Giannakis, “Low-Complexity Training for Rapid Timing Acquisition in Ultra Wideband Communications,” *IEEE, GLOBECOM*, 2003.
2. Sandeep R. Aedudodla, Saravanan Vijayakumaran, and Tan F. Wong, “Timing Acquisition in Ultra-wideband Communication Systems,” *IEEE Transaction on Vehicular Technology,* Vol. 54, No. 5, September 2005.
3. Liuqing Yang and Georgios B. Giannakis, “Blind UWB Timing With a Dirty Template,” *IEEE, ICASSP*, 2004.