

Low Power Digital Baseband for Impulse Radio Ultra-Wideband Transceiver

Wei Da Toh · Yuanjin Zheng · Chun-Huat Heng

Received: 8 April 2010 / Revised: 8 December 2010 / Published online: 29 December 2010
© Springer Science+Business Media, LLC 2010

Abstract In this paper, a low power digital baseband to be used together with impulse-radio ultra wideband radio frequency front-end has been presented. It can provide received pulse synchronization required for burst mode and low power operation. It also overcomes clock drift issue between different transceivers. The clock and data recovery is implemented fully in digital domain without the need of conventional phase-locked loop, delay locked loop or analog-to-digital converter. The chip is designed using 0.18 μm CMOS technology. It consumes 5 mW and can recover data up to 20 Mbps.

Keywords Clock drift · Digital baseband · Synchronization · Ultra wideband (UWB) · Wireless transceiver

Abbreviations

ADC	Analog-to-digital converter
BER	Bit error rate
CDR	Carrier data recovery
DLL	Delay-locked loop
FSM	Finite state machine
HDL	Hardware description language
IR	Impulse radio
LFSR	Linear feedback shift register

W.D. Toh · C.-H. Heng (✉)
Department of ECE, National University of Singapore, Singapore, Singapore
e-mail: elehch@nus.edu.sg

W.D. Toh
e-mail: g0700301@nus.edu.sg

Y. Zheng
Institute of Microelectronics, Singapore, Singapore
e-mail: yuanjin@ime.a-star.edu.sg

LNA	Low noise amplifier
MAC	Medium access control
MCU	Micro-controller unit
NRZ	Non-return-to-zero
OOK	On-off keying
PLL	Phase-locked loop
PN	Pseudorandom number
RF	Radio frequency
RX	Receiver
RZ	Return-to-zero
SPI	Serial peripheral interface
TX	Transmitter
UWB	Ultra wideband
VCO	Voltage controlled oscillator
WPAN	Wireless personal area network
WSN	Wireless sensor network

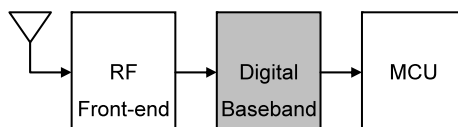
1 Introduction

Ultra wideband (UWB) transceiver had become a hot research topic due to its potential low power applications in wireless personal area networks (WPAN) and wireless sensor networks (WSN). As shown in Fig. 1, impulse radio (IR) UWB receiver architecture consists of RF front-end, digital baseband and micro-controller unit (MCU). In the literature, most researches emphasize on RF front-end implementation [5, 8, 12], which recovers the transmitted pulse into either return-to-zero (RZ) or non-return-to-zero (NRZ) formats. Most of them exclude the clock and data recovery (CDR) function. The CDR ensures proper burst mode operation to minimize receiver power consumption. It also overcomes clock drift issue between different transceivers.

In conventional RF transceiver, phase locked loop (PLL) and delay locked loop (DLL) are very popular analog techniques for CDR [4, 6]. However, their implementations do not integrate well with IR UWB transceiver. One of the key attractiveness of IR UWB transceiver is its amenability to digital processing. This eliminates many power hungry RF or analog blocks such as mixer and frequency synthesizer. Therefore, it is important for us to maintain this trait of IR UWB transceiver while implementing the CDR function.

Another popular all digital timing recovery technique involves the use of analog-to-digital converter (ADC), followed by digital signal processing [1, 10]. This method requires the partition of analog and digital boundary at the very early stage of RF

Fig. 1 General UWB receiver



front-end, usually right after the low noise amplifier (LNA). The required over sampling and narrow pulse width often implies high speed processing which is very power consuming.

In this paper, we look at alternative way of digital timing recovery that integrates well with IR UWB RF front-end. The proposed method eliminates the need for PLL/DLL, and avoids the employment of ADC and high speed processing mentioned earlier. In this paper, the developed digital CDR will be used together with the RF front-end reported in [5] where On-Off Keying (OOK) modulation is employed.

2 Proposed Digital Baseband Architecture

The proposed digital baseband is shown in Fig. 2. It consists of Serial Peripheral Interface Bus (SPI), pulse width calibration, voltage-controlled oscillator (VCO) calibration, pseudorandom number (PN) sequence generator, encoder, decoder/correlator, bit error rate (BER) estimator, pulse searcher, and pulse tracker.

The SPI module serves as an interface between the internal reconfigurable registers and the external programming device (MCU). The internal registers will control the settings of sub-blocks within digital baseband and RF front-end. Both the VCO and pulse width calibration are used together with the transmitter block to calibrate centre frequency and pulse width of UWB pulse.

The PN sequence generator is included for testing and debugging purpose. It can generate pseudorandom bit sequence with length of $2^9 - 1$ based on linear feedback shift register (LFSR). The generated bit sequence can either serve as input signal to the transmitter for transmitter testing, or input signal to the receiver path within digital baseband for receiver testing. To enhance the sensitivity of transceiver, additional encoder and decoder/correlator blocks are also included. The encoder encodes the transmitted bit sequence with either Barker code or Gold code before sending it to the transmitter. Similarly, the decoder/correlator block correlates the received bit sequence with the given Barker or Gold code to recover the original transmitted signal.

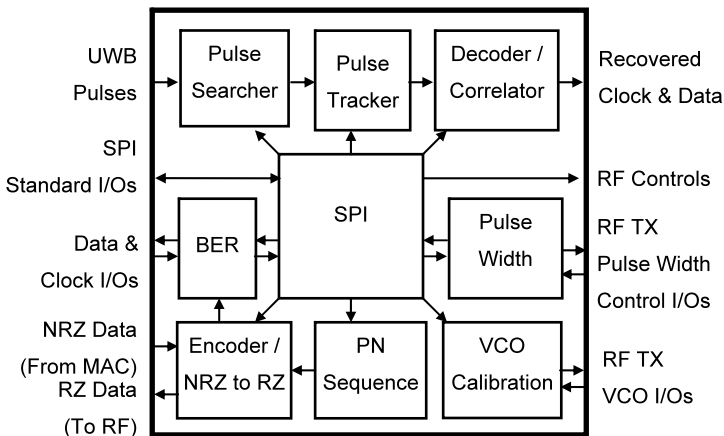


Fig. 2 Digital baseband block diagram

This can improve the sensitivity of transceiver by providing additional coding gain. However, it will lower overall throughput of transceiver due to additional coding bits. The BER estimator is included for the optimization of RF transceiver settings. During the calibration mode, the digital baseband will cycle through different RF transceiver settings and obtain BER for each setting. The setting with the lowest BER will be chosen as the optimum settings for RF transceiver after calibration.

Pulse searcher and pulse tracker are used together to mimic the function of clock and data recovery. Given a sequence of pilot signals, pulse searcher would identify the timing location of received UWB pulse within a specific time interval. However, due to clock drift issue between transmitter and receiver, this identified timing location will also drift with time and become inaccurate. To circumvent this problem, pulse tracker module is invoked once pulse searcher successfully identifies received UWB pulse timing location. It will continuously track the location of UWB pulse even in the presence of clock drift. It also recovers the received data into NRZ format and generates the corresponding data clock. Both the period and duty cycle of data clock are adjusted continuously in the presence of clock drift. In addition, after synchronization, pulse tracker module also generates enable signal for transceiver's burst mode operation. These two blocks will be explained in detail in the following section.

3 Digital Implementation of Clock and Data Recovery

There are three assumptions needed in the implementation of the proposed system. Firstly, digital baseband requires a pilot signal consists of 16 consecutive transmitted 1s at the start of each data packet. This facilitates the identification of received UWB pulse location. Secondly, due to OOK modulation, continuous tracking of pulse location can only occur for received 1s. Therefore, long consecutive received 0s are prohibited in the transmitted sequence. This should not pose any severe problem because a well-designed Media Access Control (MAC) protocol should ensure sufficient randomization of transmitted bit sequence to avoid the above-mentioned scenario. Thirdly, the received UWB pulse has been sufficiently amplified by RF front-end (LNA, energy detector and limiting amplifier) to rail-to-rail level. It should also be pointed out that our proposed solution only targets for OOK modulation. The algorithm would be much more complicated for pulse-position modulation (PPM) due to offset in pulse position and the required timing resolution.

3.1 UWB Pulse Searching Module

As shown in Fig. 3, UWB pulse searching module consists of four D-flip-flop detectors, a sampling controller, ten negative edge-triggered 5-bit registers (NR0–NR9), ten positive edge-triggered 5-bit registers (PR0–PR9) and a decision Finite State Machine (FSM). UWB pulses are detected using the D-flip-flop detector as shown in Fig. 4.

For any given data rate, the system clock is first set to 10 times of the data rate through an internal frequency divider. Each symbol period is then divided into 10

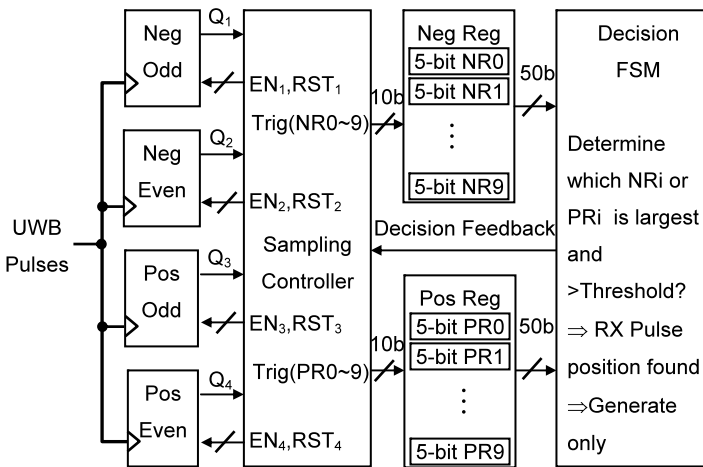
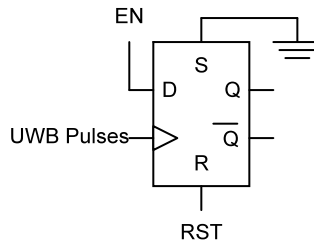


Fig. 3 UWB pulses searching algorithm block diagram

Fig. 4 D-flip-flop detector



smaller duration windows by the sampling controller. Therefore, one duration window (T_{dur}) is equivalent to one system clock interval (T_{clk}). Each duration window within the symbol period is then tracked by either positive counter or negative counter as shown in Fig. 5. In addition, each duration window will have its corresponding 5-bit register to store the number of UWB pulse detected during that duration window. As an example, NR1 would be the 5-bit register corresponds to the duration window when the negative counter equals to 1. The sampling controller will generate the control signals (EN_i and RST_i) for each of the D-flip-flops as shown in Fig. 5.

There are three operation phases for D-flip-flop detector. D-flip-flop is first reset during reset phase ($RST = 1$). After which, D-flip-flop will enter acquisition phase ($EN = 1/RST = 0$) to detect any incoming pulse for one full duration window. Its output (Q) will then be fed to the controller during sampling phase. Any incoming pulse will set the Q to 1 and increase the count of the corresponding 5-bit register (NR_i or PR_i) by 1 through Trig signal. As the three phases of operation span two duration windows ($2 \times T_{clk}$), both odd and even D-flip-flops are needed to cover 10 duration windows alternatively. The sampling controller will generate proper EN_i and RST_i for four flip-flop detectors. It will also produce Trig signals for NR_i and PR_i based on Q outputs from flip-flop detectors. At the end of 16 pilot symbols of transmitted 1s, both the negative and positive edge-triggered registers (NR_i and PR_i) will have statistics which indicate the most likely timing location of incoming UWB

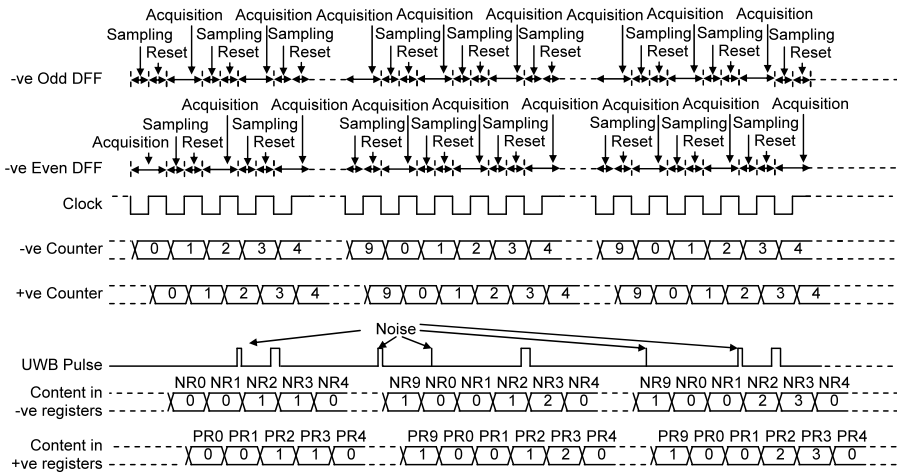
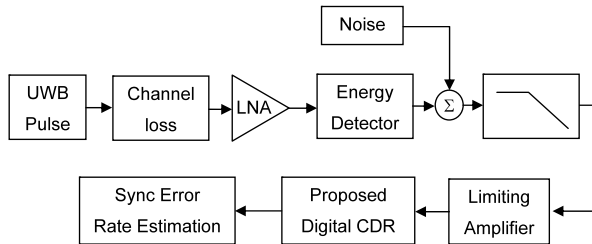


Fig. 5 UWB pulse searching algorithm timing diagram

Fig. 6 Matlab behavioral simulation for synchronization error



pulses. The decision FSM will then determine this timing location based on the collected data from these registers. Its decision will be feedback to sampling controller for proper selection of D-flip-flop detector. Control signals for subsequent detection of incoming UWB pulses will also be generated. As illustrated by the example shown in Fig. 5, incoming UWB pulses occur at duration window corresponding to NR3 and PR3, which have the highest number of count. During detection, noise and glitches could randomly increase the register counts, such as NR2/PR2 and NR9/PR9. By setting proper threshold in decision FSM (NR_i or $PR_i > 12$), we should be able to mask these detection errors and find the correct timing location for incoming UWB pulses.

The setting of threshold depends on two factors, i.e., receiver sensitivity and channel condition. Poorer channel condition might result in many detection errors and require higher threshold to mask these detection errors. On the other hand, poorer sensitivity with good channel condition might require lower threshold to enable the detection of the desired UWB received pulses. To investigate the selection of threshold, a Matlab behavioral model as shown in Fig. 6 has been built. The RF front-end is modeled after [5]. Noise is injected after the energy detector to vary signal to noise ratio (SNR). It should be pointed out that the RF front-end reported in [5] is not optimum due to the energy detection and limiting amplifier employed. It can result in

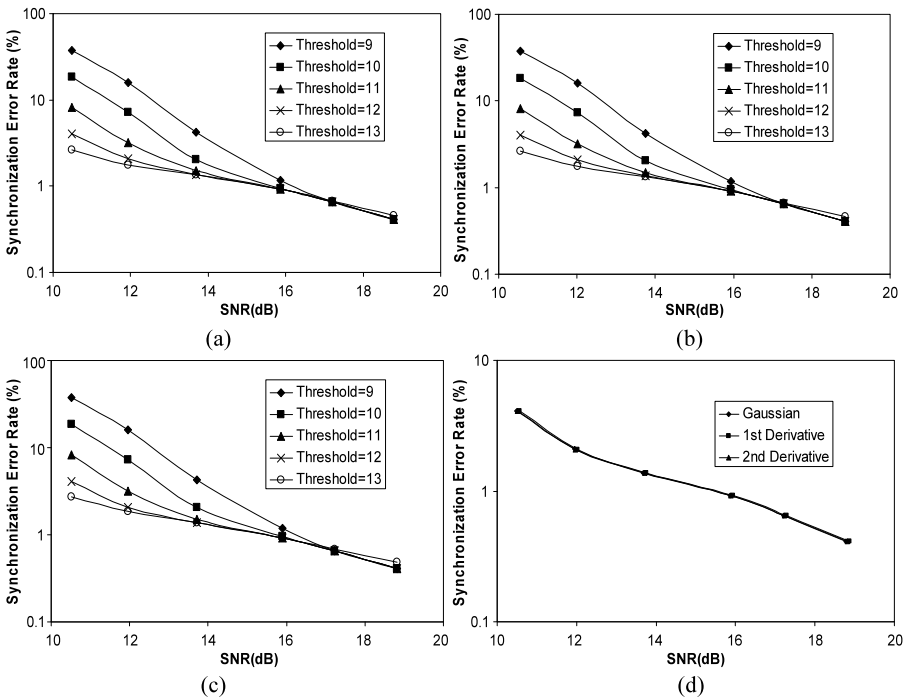


Fig. 7 Synchronization error rate for (a) Gaussian pulse, (b) 1st derivative, (c) 2nd derivative, and (d) different pulse shapes at threshold = 12

a lot of glitches or noise pulses under poor SNR due to high gain of limiting amplifier. The synchronization error rate with different pulse shapes are shown in Fig. 7. As illustrated, high synchronization error rate is observed with lower threshold (<12) for smaller SNR. However, with too high a threshold (>12), the synchronization error rate saturates at higher SNR and performs poorer than lower threshold. This is because high threshold causes failure in detecting right timing location. From this study, the optimum threshold is 12 that tallies with our measurement setting. It is also observed that there is little difference in synchronization error rate for different pulse shapes adopted in the simulation. This is because energy detection instead of matched filter receiver is employed in our RF front-end. Matched filter receiver is not used as it complicates the design of RF front-end and increases power consumption. The transmitter employed in our project exhibits pulse shape similar to the 2nd order derivative of a Gaussian pulse.

Most of the time, positive and negative edge-triggered registers will record same number of UWB pulse detection. Therefore, either one can be chosen for subsequent UWB pulse detection. In the proposed algorithm, we choose the duration window corresponding to NR_i if $NR_i = PR_i$. However, if the incoming pulse is occurring right at the transition edge, the corresponding edge-triggered registers might fail to detect pulses due to meta-stability issue. Therefore, both positive and negative edge-triggered flip-flop and registers are included to overcome the problem. If positive (negative) edge-triggered flip-flop encounters meta-stability issue and fails to detect

Fig. 8 UWB pulse detect algorithm flowchart

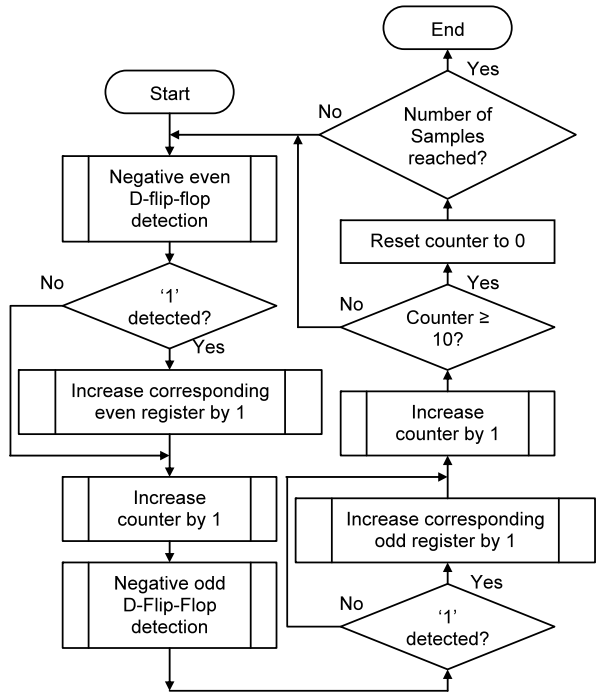
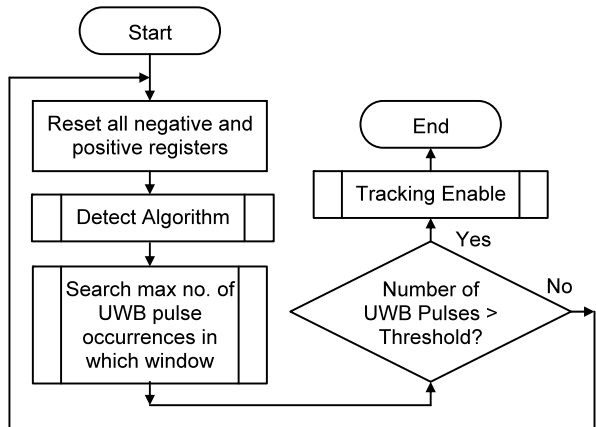


Fig. 9 UWB pulse searching algorithm flowchart



incoming signal, negative (positive) edge-triggered flip-flop should still be able to detect incoming pulse and result in $NR_i > PR_i$ (or $PR_i > NR_i$). The flowchart for pulse detect algorithm is also illustrated in Fig. 8.

Figure 9 shows the complete pulse searching algorithm flowchart. At the beginning detection of each data packet, the count for both the positive and negative edge-triggered registers are reset to zeros. After which, UWB pulse detect algorithm discussed earlier will be invoked. At the end of 16 pilot symbols, FSM will determine the duration window with maximum number of detected UWB pulses. If the num-

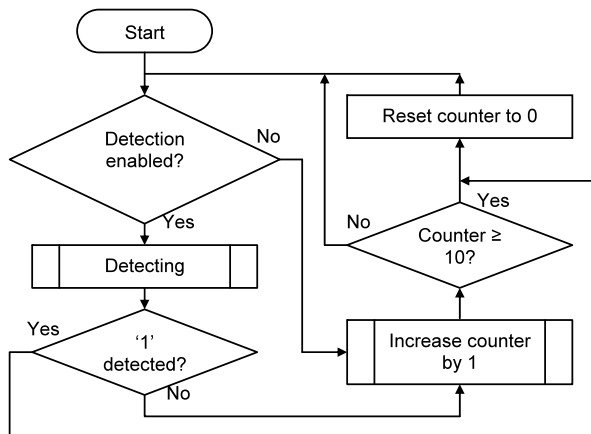
ber also exceeds the preset threshold (12), the pulse searching algorithm completes and the pulse tracker is invoked. Otherwise, the pulse searching algorithm will be repeated.

3.2 UWB Pulse Tracking Algorithm

Once UWB pulse searching algorithm completes, the exact duration window where UWB pulses occur will be known. UWB pulse tracking algorithm as shown in Fig. 10 will begin. Pulse detection will span three consecutive duration windows. The center duration window is determined from pulse searching algorithm. A tracking counter will be used to track UWB pulse. As tracking algorithm only corrects drift upon detection of 1s, slow clock drift coupled with short consecutive transmission of 0s are needed to ensure the proper operation of the algorithm. This is a reasonable assumption given that typical crystal oscillator exhibits frequency stability of ± 50 ppm. Assuming a worst case scenario of 100 ppm between RX and TX system clocks, this will give rise to frequency difference of 10 kHz for system clock of 100 MHz. For the algorithm to fail, RX and TX system clocks have to be out of synchronization by more than one system clock interval (10 ns) between the detection of 1s. This corresponds to transmission of 1000 consecutive 0s at 10 Mbps (100 μ s), which is unlikely to happen for a well-designed MAC.

When tracking counter value is zero, it indicates the position of duration window where UWB pulse should occur. Under normal condition with no pulse detected, tracking counter would repetitively count to 10 cycles and reset, which matches symbol period exactly. The algorithm will reset tracking counter to zero once UWB pulse is detected. If the position of UWB pulse does not change over time, it will be detected when tracking counter value is zero. Therefore, it would not incur any change on tracking counter as well as duration window. However, if clock drift causes the UWB pulse position to change to the adjacent duration window, received UWB pulse will reset the tracking counter to zero instantaneously and the position of the center duration window will now be updated. The detailed timing diagram of pulse tracking algorithm is shown in Fig. 11. As illustrated, initial UWB pulse occurs when negative

Fig. 10 UWB pulses tracking algorithm flowchart



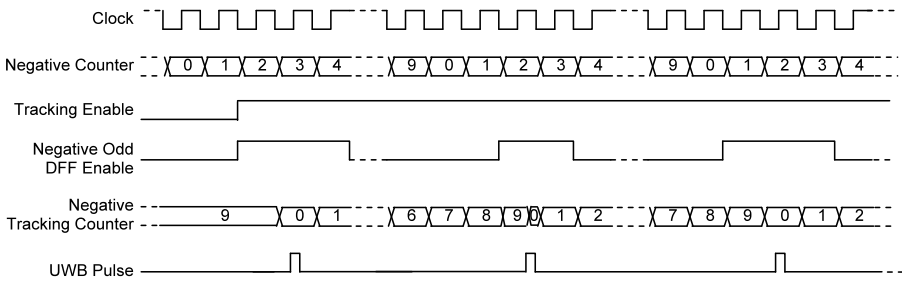
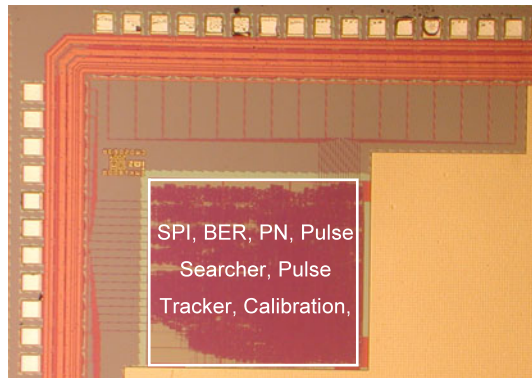


Fig. 11 UWB pulse tracking algorithm timing diagram

Fig. 12 Die photo



counter equals to three, where the tracking counter is set to zero. Once UWB pulse drift to duration window where negative counter equals to two, the tracking counter will be reset right away to reflect the changes. The corresponding NRZ data and data clock can then be easily generated based on tracking counter and detecting D-flip-flop output.

The full system simulation is done using Matlab Simulink. Once the critical system parameters are determined, Cadence NC-Verilog is employed for digital simulation. The whole chip is then synthesized with Synopsys DC Compiler before sending to Cadence SOC Encounter for place and route. Post-layout simulation is then performed again with NC-Verilog after incorporating the post-layout verilog netlist and timing data.

4 Measurement Results

The digital baseband is implemented using Verilog hardware description language (HDL) and fabricated using 0.18 μm CMOS technology. The chip occupies a die area of 0.8 mm \times 0.7 mm and is shown in Fig. 12. It should be pointed out that the reported chip includes all the functions described in earlier section, including pulse searching and tracking module. To verify the functionality of clock and data recovery, the following setup is used. 100 MHz crystal oscillator is employed as system clock and the

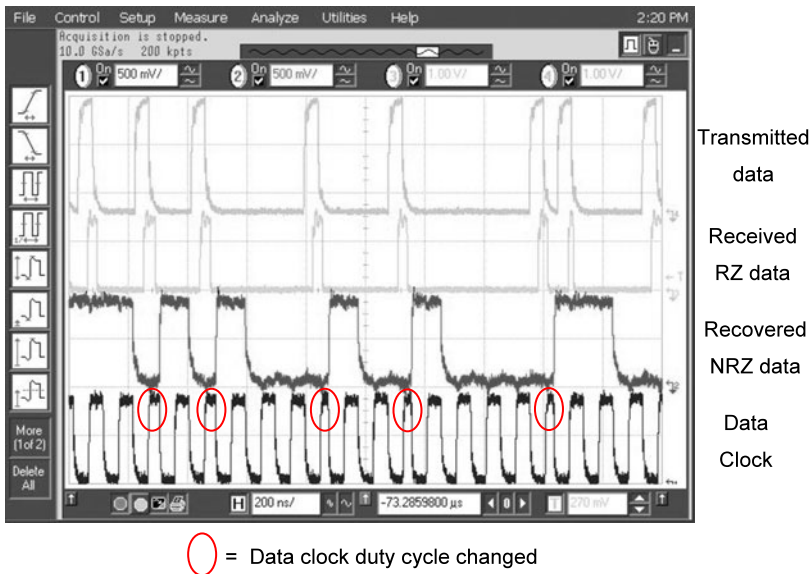


Fig. 13 Measured waveforms

targeted data rate is 10 Mbps. A PN data sequence with bit rate of 10.5 Mbps is transmitted through a UWB transmitter and received by a corresponding UWB receiver front-end. The receiver and transmitter employed are similar to those reported in [5] and [3] earlier. The recovered RZ data from receiver front-end is then sent to the proposed digital baseband for proper clock and data recovery. The PN data is purposely set to 10.5 Mbps to introduce clock drift phenomenon. The recovered NRZ data and corresponding data clock are shown in Fig. 13. As illustrated, digital baseband correctly recovers transmitted data in NRZ format and eliminates clock drift issue. As pulse tracking algorithm can only update pulse position when 1s are detected, we see that duty cycle of data clock is only updated when 1s are received. In addition, when longer consecutive 0s are observed in the transmitted sequence, more drift error is accumulated, which result in larger change in duty cycle of data clock. The experiment setup achieves sensitivity of -70 dBm @ 20 Mbps with BER of 10^{-3} with cable testing [5]. The performance is compared with other UWB receiver in Table 1. The higher bit energy consumed is due to larger LNA current [5] which improves the sensitivity. The setup can also achieve wireless transmission distance of 2.5 m with BER of 10^{-3} at same data rate. For wireless transmission, multi-path fading might potentially result in receiving pulses at multiple duration windows and cause error at pulse searching algorithm. However, this phenomenon is not observed during the testing. One possible explanation is due to the poor sensitivity of non-coherent receiver employing energy detector. As received pulses due to multi-path fading have much smaller energy, they do not result in rail-to-rail UWB RZ pulse at the input of digital baseband. The performance of digital baseband is summarized in Table 2. It operates under 1.8 V and consumes only 5 mW. From testing, it can handle data rate up to 20 Mbps.

Table 1 Performance comparison

Paper	Modulation	Sensitivity (dBm)	Data rate	nJ/bit	Tech./Area
[8]*	PPM	−99	100 kbps	2.5	90 nm/2 mm ²
[2]*	OOK	−65	1 Mbps	2.6	180 nm/1.8 mm ²
[11]	PPM	N/A	20 Mbps	1.44	180 nm/NA
[5]	OOK	−70	20 Mbps	3.1	180 nm/
+		−83	1 Mbps	+	2.8 mm ² +
This work				0.25	0.56 mm ²

*Without clock and data recovery

Table 2 Digital baseband specification

System clock	<200 MHz
Data rate	<20 Mbps
Area	800 μm × 700 μm
Operating voltage	1.8 V
Power consumption	5 mW @ 10 Mbps
Technology	0.18 μm CMOS

It is difficult to obtain a fair comparison for the proposed solution and other reported OOK receivers. Most OOK receivers, such as [8] and [2], do not consider CDR and clock drift issue, assuming it will be taken care of by baseband. On the other hand, [1] and [11] circumvent the problems through the introduction of power hungry ADC. In [11], signal band of 500 MHz is first down converted to baseband before sending to ADC. This avoids high sampling rate and results in lower power ADC of around 10 mW. However, it has added cost of mixer and LO. On the other hand, [1] employs direct sampling of the RF signal which results in high power ADC of around 80 mW due to narrow UWB pulse width. Although the ADC also provides higher flexibility and sensitivity for receiver, the larger power penalty might not be desirable. The PLL and DLL solutions introduced in [7, 9], covering similar data rate, consume around 17 mW.

5 Conclusion

A reconfigurable digital baseband for UWB transceiver is designed using 0.18 μm CMOS. The clock and data recovery function is fully implemented in digital domain without any need for PLL/DLL or ADC, and is thus very amenable to the integration with UWB transceiver. The successful synchronization of received data also allows burst mode operation to minimize the power of RF front-end.

Acknowledgement This research is supported by Singapore Agency for Science, Technology and Research SERC grant R-263-000-459-592.

References

1. R. Blazquez, P. Newaskar, F. Lee, A. Chandrakasan, A baseband processor for pulsed ultra-wideband signals, in *Proc. CICC* (2004), pp. 587–590
2. D.C. Daly, A.P. Chandrakasan, An energy-efficient OOK transceiver for wireless sensor networks. *IEEE J. Solid-State Circuits* **42**, 1003–1011 (2007)
3. S. Diao, Y. Zheng, C.H. Heng, A CMOS ultra low-power and highly efficient UWB-IR transmitter for WPAN applications. *IEEE Trans. Circuits Syst. II* **56**, 200–204 (2009)
4. R. Farjad-Rad, A. Nguyen, J. Tran, T. Greer, J. Poulton, W.J. Dally, J.H. Edmondson, R. Senthinathan, R. Rathi, M.J.E. Lee, H.T. Ng, A 33 mW 8 Gb/s CMOS clock multiplier and CDR for highly integrated I/Os. *IEEE J. Solid-State Circuits* **39**, 1553–1561 (2004)
5. Y. Gao, Y. Zheng, C.H. Heng, Low-Power CMOS RF front-end for non-coherent IR-UWB receiver, in *Proc. ESSCIRC* (2008), pp. 386–389
6. P.K. Hanumolu, G.Y. Wei, U.K. Moon, A wide-tracking range clock and data recovery circuit. *IEEE J. Solid-State Circuits* **43**, 425–439 (2008)
7. P. Larsson, A 2-1600-MHz CMOS clock recovery PLL with low-V_{dd} capability. *IEEE J. Solid-State Circuits* **34**, 1951–1960 (1999)
8. F.S. Lee, A.P. Chandrakasan, A 2.5 nJ/bit 0.65 V pulsed UWB receiver in 90 nm CMOS. *IEEE J. Solid-State Circuits* **42**, 2851–2859 (2007)
9. C.K. Liang, R.J. Yang, S.I. Liu, An all-digital fast-locking programmable DLL-based clock generator. *IEEE Trans. Circuits Syst. I* **55**, 261–269 (2007)
10. I. O'Donnell, S. Chen, S. Wang, R. Brodersen, An integrated, low power, ultra-wideband transceiver architecture for low-rate, indoor wireless systems, in *Proc. IEEE CAS Workshop on Wireless Communications and Networking* (2002), pp. 1623–1631
11. J. Ryckaert et al., A CMOS ultra-wideband receiver for low data-rate communications. *IEEE J. Solid-State Circuits* **42**, 2515–2527 (2007)
12. T. Terada, S. Yoshizumi, M. Muqsith, Y. Sanada, T. Kuroda, A CMOS ultra-wideband impulse radio transceiver for 1-Mb/s data communications and ± 2.5 -cm range finding. *IEEE J. Solid-State Circuits* **41**, 891–898 (2006)